

В. Л. БЫКОВ, Ю. П. АШАЕВ

# Основы информатики Практикум

*Рекомендовано советом Брестского государственного  
технического университета в качестве пособия для студентов  
технических специальностей*

БРЕСТ 2006

УДК 004(75)  
ББК 22.183.492 : 73Я73  
Б 95

**Рецензенты:**

профессор кафедры "Вычислительные методы и программирование" Белорусского государственного университета информатики и радиоэлектроники,  
доктор физико-математических наук, профессор **Колосов С. В.**;

заведующий кафедрой "Информационные технологии" Брестского государственного педагогического университета, кандидат физико-математических наук доцент **Котов И. В.**

**В. Л. Быков, Ашаев Ю. П.**

Б95 Основы информатики. Практикум. Пособие для студентов технических специальностей. Издательство БрГТУ. – Брест. 2006. - 316 с.: ил.

ISBN 985-493-048-3

Содержит описание лабораторных работ по всему курсу дисциплины "Информатика", предназначено для активного изучения предмета. Пособие является второй частью пособия "Информатика" тех же авторов, поэтому содержит лишь краткие теоретические сведения. Ориентировано на студентов технических специальностей, но может быть полезно всем студентам, изучающим самостоятельно основы информатики и вычислительной техники.

Имеет обширный справочный материал и ряд дополнительных глав по языку программирования Visual Basic 6.0, имеющих прикладную направленность.

УДК 681:3.06:800.92  
ББК 22.183.492: 73Я73

ISBN 985-493-048-3

© В. Л. Быков, 2006  
© Ю. П. Ашаев, 2006  
© Издательство БрГТУ, 2006

## Оглавление

Введение	5
Требования к выполнению лабораторных работ	6
1. Общие сведения о компьютере	6
1.1. Системы счисления	6
1.2. Знакомство с компьютером	10
1.3. Работа на печатающем устройстве	16
1.4. Сервисная оболочка Far Manager. Знакомство	21
1.5. Работа в среде сервисной оболочки Far Manager	28
1.6. Знакомство с Windows	32
1.7. Настройка параметров рабочего стола	37
1.8. Знакомство с Windows Commander.	39
2. Математическая система MATHCAD	41
2.1. Организация вычислений и типы данных	41
2.2. Символьные вычисления	53
2.3. Построение графиков	64
2.4. Численные вычисления	73
2.5. Матричные вычисления	87
2.6. Решение дифференциальных уравнений	94
2.7. Решение прикладных задач	100
3. Алгоритмизация и программирование	107
3.1. Базовые структуры схем алгоритмов	107
3.2. Изучение среды разработки проекта	109
3.3. Разработка простой формы	114
3.4. Разработка линейных программ	117
3.5. Программы с ветвлениями	123
3.6. Циклы с заданным числом повторений	129
3.7. Операции с массивами	134
3.8. Функции пользователя	138
3.9. Циклы с параметром	140
3.10. Численное интегрирование	142
3.11. Решение алгебраических и трансцендентных уравнений	149
3.12. Интерполирование функций	150
3.13. Обработка символьных переменных	154
3.14. Разработка меню пользователя	157
3.15. Работа с массивами элементов управления	161
3.16. Использование сетки MSFlexGrid	164
3.17. Графические объекты	168
3.18. Исследование методов Line и Circle	170
3.19. Построение графиков и диаграмм	174
3.20. Анимация	176

3.21. Дополнительные элементы интерфейса	181
3.22. Работа с файлами	185
4. Редактор документов Microsoft Word	191
4.1. Ввод и редактирование текста	191
4.2. Оформление документа	194
4.3. Создание и редактирование рисунков, объектов WordArt	197
4.4. Таблицы и диаграммы	201
4.5. Работа с формами и макросами	203
4.6. Слияние документов	207
5. Электронная таблица Excel	209
5.1. Разработка таблиц	209
5.2. Табулирование функций	215
5.3. Графические возможности электронной таблицы	220
5.4. Расчет деформации твердого тела	223
5.5. Работа с матрицами. Решение систем линейных уравнений	226
5.6. Списки (базы данных) в электронной таблице	230
5.7. Разработка функций пользователя	236
5.8. Решение алгебраических и трансцендентных уравнений	239
5.9. Определение коэффициентов эмпирических формул методом наименьших квадратов	243
5.10. Автоматизация операций в Excel	248
5.11. Решение дифференциальных уравнений первого порядка с начальными условиями	251
6. ИНТЕРНЕТ	255
6.1 Поиск информации в ИНТЕРНЕТ	255
6.2. Создание Web - странички	258
Приложение 1. Основные приемы работы в среде Visual Basic	264
Приложение 2. Префиксы имен объектов	266
Приложение 3. Основные функции и типы данных	267
Приложение 4. Классы и операторы Visual Basic	274
Приложение 5. Моделирование движения механизма	276
Приложение 6. Проверка и обработка пользовательского ввода. Обработка ошибок	298
Приложение 7. Отладка программ на языке Visual Basic	309
Литература	314

## Введение

Настоящее издание является второй книгой пособия "Основы информатики" тех же авторов. В нем приведены задания для практических занятий по приобретению и закреплению навыков практического использования информационных компьютерных технологий.

Данное пособие состоит из шести частей. Состав заданий подобран таким образом, чтобы обеспечить последовательное изучение материала, в том числе и при самостоятельной работе без помощи преподавателя. Каждое задание содержит краткие теоретические сведения, методические указания по выполнению задания, контрольные вопросы. В необходимых случаях приведены примеры выполнения заданий, упражнения для закрепления навыков.

В первой части приведены задания для изучения клавиатуры, основных типов печатающих устройств, сервисных оболочек Windows Commander, Far Manager, операционной системы Windows 98, программы Проводник.

Во второй части приведены задания для изучения приемов работы и приобретения практических навыков работы в среде MathCad.

Третья часть посвящена алгоритмизации и программированию. Базовые структуры языка программирования и алгоритмы решения типовых задач рассматриваются на базе языка программирования Visual Basic 6.0. Язык программирования VB 6.0 выбран в качестве базового языка программирования студентов технических специальностей непрограммистов по ряду причин: простота синтаксиса; совпадение базовых структур языка VB 6.0 со структурами языка алгоритмического программирования, изучаемого в средней школе; использование версий данного языка в приложениях Windows для написания прикладных программ и макросов. Здесь же достаточно подробно рассмотрены вопросы по разработке интерфейса пользовательских программ.

Четвертый раздел содержит задания для изучения приемов работы в среде текстового процессора Microsoft Word. Основное внимание уделено оформлению документов средствами текстового процессора.

В пятом разделе приведены задания для изучения приемов работы в среде табличного процессора - электронной таблицы Microsoft Excel и использования ее для решения инженерных задач.

Шестой раздел содержит задания для обучения практике поиска информации в ИНТЕРНЕТ и использования языка HTML для создания домашней странички.

Пособие подготовлено к изданию Быковым Вячеславом Леонидовичем. Разделы второй и седьмой подготовлены Ашаевым Юрием Павловичем.

Пособие предназначено для студентов технических специальностей, может быть полезно и для студентов других специальностей, а также для всех желающих изучить основы дисциплины "Информатика" и основы программирования.

Авторы выражают искреннюю благодарность рецензентам доктору ф.-м. н., профессору кафедры "Вычислительные методы и программирование" Белорусского государственного университета информатики и радиоэлектроники Колосову С. В. и заведующему кафедрой "Информационные технологии" Брестского государственного педагогического университета, кандидату ф.-м. н., доценту Котову И. В. за ценные замечания, способствовавшие улучшению структуры и содержания пособия.

# Требования к выполнению лабораторных работ

## Общие требования

Подготовка к каждой лабораторной работе осуществляется студентом во внеаудиторное время. Студенты знакомятся с общими сведениями и порядком выполнения работы, пишут необходимые программы в соответствии с полученным вариантом задания.

Перед занятием преподаватель проверяет готовность студентов к выполнению лабораторной работы и принимает решение о допуске студента к выполнению работы.

Лабораторная работа выполняется студентом самостоятельно.

По результатам выполнения лабораторной работы каждый студент оформляет отчет. Отчет должен быть составлен и представлен преподавателю на текущем или следующем занятии. Если отчет не представлен, работа считается не выполненной.

После представления отчета он должен быть **ЗАЩИЩЕН**. Защита лабораторной работы служит для проверки качества усвоения материала студентом и заключается в объяснении цели занятия и ответе на контрольные вопросы.

Защита лабораторной работы проводится на текущем или следующем занятии.

## Требования к содержанию и оформлению отчета

1. Отчет по лабораторной работе оформляется в двенадцатилистовой тетради или на отдельных листах. На обложке тетради (на титульном листе) указывается фамилия, имя и отчество студента, номер группы и предмет обучения.

2. В отчете указывается:

- \* название лабораторной работы, тема и цель занятия;
- \* задание на лабораторную работу;
- \* протокол выполнения работы (порядок выполнения работы, используемые команды и их форматы с необходимыми пояснениями);
- \* распечатки или рисунки результатов работы;
- \* ответы на контрольные вопросы;
- \* выводы.

## 1. ОБЩИЕ СВЕДЕНИЯ О КОМПЬЮТЕРЕ

### 1.1. Системы счисления

**Цель работы:** ознакомиться с системами счисления, применяемыми в вычислительной технике, приобрести навыки преобразования данных из одной системы счисления в другую.

**Время:** 2 часа.

**Литература:** Л1, с. 18 – 21, Л2. с. 18-23.

#### 1.1.1. Общие сведения

В позиционной системе счисления произвольное число  $A$  может быть представлено в виде полинома:

$$A(q) = a_n \cdot q^{n-1} + a_{n-1} \cdot q^{n-2} + \dots + a_1 q^1 + a_0 q^0 + a_{-1} q^{-1} + \dots + a_{-m} q^{-m}, \quad (1.1)$$

где  $a_i$  - символы алфавита системы счисления;

$q$  - основание системы счисления;

$n, m$  - число разрядов в целой и дробной частях числа соответственно.

Например, определить значение числа, представленного в двоичной системе счисления  $1101101.1011_2$ .

**Решение.** Число содержит целую и дробную части:  $n=7$ ,  $m=4$ .

$$A_{(2)} = 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} + 1 \cdot 2^{-4} = 1 \cdot 64 + 1 \cdot 32 + 0 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 + 1 \cdot 1/2 + 0 \cdot 1/4 + 1 \cdot 1/8 + 1 \cdot 1/16 = 109,6875$$

Любая позиционная система счисления характеризуется *основанием* (базисом). Основание позиционной системы счисления - это число знаков или символов для изображения цифр в данной системе.

В вычислительной технике применяются преимущественно следующие позиционные системы счисления: двоичная, восьмеричная, шестнадцатеричная, десятичная. Основаниями в этих системах счисления являются числа 2, 8, 16 и 10, соответственно. Коды этих систем счисления приведены в табл. 1.1. Цветом выделены ячейки, содержащие коды алфавита в соответствующих системах счисления. В двоичной системе счисления таких кодов два: ноль и единица; в восьмеричной системе счисления 8 - коды от 0 до 7, в шестнадцатеричной системе счисления 16 - коды от 0 до F. Для представления чисел больших, чем основание системы счисления, необходимо увеличивать число разрядов.

Для систем счисления, приведенных в таблице, справедливы следующие формулы. Требуемое число разрядов  $n$  для представления максимального числа  $A_{\max}(q)$ , где  $q$  - основание системы счисления, равно:

$$n = \log_q(A_{\max}(q) + 1). \quad (1.2)$$

И наоборот, если известна длина разрядной сетки  $n$ , то можно определить максимальное число  $A_{\max}(q)$ , которое можно представить с использованием данной разрядной сетки:

$$A_{\max}(q) = q^n - 1. \quad (1.3)$$

Например, с помощью одного байта (восьми разрядов) можно представить число 255, с помощью шестнадцати разрядов - 65535 и т. д.

Таблица 1.1 Позиционные системы счисления

Десятичная	Двоичная	Восьмеричная	Шестнадцатеричная
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11

Для перевода чисел из одной системы счисления в другую разработаны правила (Л1, с 18 - 20).

**Примеры:**

1. Перевести число  $101\ 100\ 111\ 110_2$  из двоичной системы счисления в восьмеричную.

Разобьем двоичный код числа на триады и представим каждую триаду символом в восьмеричной системе счисления:

$$101\ 100\ 111\ 110_2 = 5476_8$$

2. Перевести число  $1011\ 1100\ 1001\ 0110_2$  из двоичной системы счисления в шестнадцатеричную систему счисления.

Разобьем двоичный код числа на тэтрады и представим каждую тэтраду символом из шестнадцатеричной системы счисления:

$$1011\ 1100\ 1001\ 0110_2 = BC96_{16}$$

3. Перевести число  $1011\ 1100\ 1001\ 0110_2$  из двоичной системы счисления в десятичную систему счисления:

$$1011\ 1100\ 1001\ 0110_2 = 2^{15} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^7 + 2^4 + 2^2 + 2^1 = 48278$$

4. Перевести число 55 из десятичной системы счисления в двоичную систему счисления и обратно.

Разделим число 55 на 2. Остаток от деления запишем в виде кода числа, а полученное число снова разделим на 2 и т. д.

Перевод целого числа из десятичной системы счисления в двоичную	Перевод целого числа из двоичной системы счисления в десятичную
$  \begin{array}{r}  55:2 = 27 \uparrow 1 \leftarrow \text{младший разряд} \\  27:2 = 13 \quad 1 \quad \text{кода двоичного} \\  13:2 = 6 \quad 1 \quad \text{числа} \\  6:2 = 3 \quad 0 \\  3:2 = 1 \quad 1 \\  1:2 = 0 \quad 1 \\  \text{Код числа: } 110111  \end{array}  $	$  \begin{array}{r}  \text{Код числа: } 110111 \\  \text{ст. разряд} \rightarrow 1 \quad   \quad = 1 \\  1 \times 2 + 1 = 3 \\  3 \times 2 + 0 = 6 \\  6 \times 2 + 1 = 13 \\  13 \times 2 + 1 = 27 \\  27 \times 2 + 1 \downarrow = 55  \end{array}  $

Для перевода двоичного числа в десятичный запишем код числа столбиком так, чтобы старший разряд кода двоичного числа был вверху. Умножим старший разряд кода двоичного числа на 2, прибавим к полученному результату код следующего разряда и полученный результат снова умножим на 2, и т. д.

Для перевода чисел из десятичной системы счисления в восьмеричную или шестнадцатеричную системы счисления надо делить число на 8 или на 16, соответственно.

5. Перевести число 0,37 из десятичной системы счисления в двоичную систему счисления и обратно.

Дробные части десятичного числа переводятся в двоичный код несколько иначе, чем целые числа: умножим число на основание системы счисления, целая часть полученного числа образует код двоичного числа, а дробная часть полученного числа используется для последующего умножения.

ПЕРЕВОД ДРОБНОЙ ЧАСТИ ЧИСЛА ИЗ ДЕСЯТИЧНОЙ СИСТЕМЫ СЧИСЛЕНИЯ В ДВОИЧНУЮ	Перевод дробной части числа из двоичной системы счисления в десятичную
$0,37 \times 2 = 0,74$ 0 ← старший разряд $0,74 \times 2 = 1,48$ 1    кода двоичного $0,48 \times 2 = 0,96$ 0    числа $0,96 \times 2 = 1,92$ 1 $0,92 \times 2 = 1,84$ 1 $0,84 \times 2 = 1,68$ 1 $0,68 \times 2 = 1,36$ 1 $0,36 \times 2 = 0,72$ 1 <span style="float: right;">↓</span> 0	мл. разряд    → 1    : 2 = 0,5 (0,5+1)    ↑ : 2 = 0,75 (0,75+1)    : 2 = 0,875 (0,875 + 1) : 2 = 0,9375 (0,9375 + 0) : 2 = 0,46875 (0,46875 + 1) : 2 = 0,734375 (0,734375 + 0) : 2 = 0,3671875

Для перевода дробной части двоичного числа в десятичную систему счисления запишите код двоичного числа таким образом, чтобы вверху был младший разряд. Разделите код двоичного числа на основание системы счисления, прибавьте к полученному числу код следующего разряда и снова разделите полученное число на основание системы счисления.

### 1.1.2. Задание

1. Переведите числа согласно варианту задания

Варианты задания к п. 1			
№	Код числа	№	Код числа
1	0110 1101 0011 1010	2	1000 1100 0101 1001
3	1000 1101 0010 1111	4	1010 0011 1011 0110
5	1010 0010 0110 1011	6	0110 0101 0110 0111
7	1100 1010 0100 1100	8	1001 0110 1101 1000
9	1011 0110 1100 0111	10	1011 0100 1001 0011
11	0001 1001 0111 1011	12	0111 1001 0110 1011
13	0110 1101 1001 1111	14	0011 1010 0111 0001
15	0010 1001 1110 1010	16	0110 1001 1110 0010

- из двоичной системы счисления в восьмеричную систему счисления;
- из двоичной системы счисления в шестнадцатеричную систему счисления.

2. Переведите число из восьмеричной системы счисления в шестнадцатеричную систему счисления согласно варианту задания.

Варианты задания к п. 2							
№	Код числа	№	Код числа	№	Код числа	№	Код числа
1	735	2	632	3	714	4	136
5	561	6	541	7	647	8	253
9	253	10	742	11	652	12	371
13	752	14	265	15	174	16	456

3. Переведите число из шестнадцатеричной системы счисления в двоичную систему счисления согласно варианту задания.

Варианты задания к п.3							
№	Код числа	№	Код числа	№	Код числа	№	Код числа
1	F7A35	2	E6A32	3	71F4C	4	D13A6
5	C56D1	6	5F4C1	7	B6D47	8	2E53B
9	2B5E3	10	C74A2	11	C65E2	12	D3B71
13	1AD65	14	EF2D4	15	38CA1	16	E63A5

4. Переведите целое двухзначное число из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления. Выполните проверку, используя формулу (1.1).

5. Переведите дробную часть числа 0,725 из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления. Выполните проверку, используя формулу (1.1).

6. Выполните перевод числа, полученного в п. 4 задания, из двоичной системы счисления в десятичную систему счисления.

7. Выполните перевод числа, полученного в п. 5 задания, из двоичной системы счисления в десятичную систему счисления.

### Контрольные вопросы

1. Опишите порядок перевода чисел из двоичной системы счисления в восьмеричную (шестнадцатеричную) систему счисления.

2. Опишите порядок перевода чисел из восьмеричной (шестнадцатеричной) системы счисления в двоичную систему счисления.

3. Опишите порядок перевода чисел из десятичной системы счисления в двоичную (восьмеричную, шестнадцатеричную) систему счисления.

4. Опишите порядок перевода дробной части десятичного числа в двоичную (восьмеричную, шестнадцатеричную) систему счисления.

5. Опишите порядок перевода числа из двоичной (восьмеричной, шестнадцатеричной) системы счисления в десятичную систему счисления.

6. Опишите порядок перевода дробной части числа из двоичной (восьмеричной, шестнадцатеричной) системы счисления в десятичную систему счисления.

## 1.2. Знакомство с компьютером

**Цель работы:** ознакомиться с составом компьютера, назначением его блоков. Получить практические навыки в работе с клавиатурой и печатающим устройством.

**Время:** 1 час.

**Литература:** Л1, 36 – 68 , Л2. 39-56.

### 1.2.1. Общие сведения

#### Структура (конфигурация) персонального компьютера

Персональный компьютер (ПК) включает следующие основные блоки: системный блок, который содержит материнскую плату, платы адаптеров, дисководы, блок питания,

шины адреса, данных и управления; видеомонитор (дисплей), клавиатуру, манипулятор типа "мышь". Внешнее оформление и компоновка ПК разных моделей могут отличаться друг от друга. В настоящее время преобладает моноблочная конструкция, в которой системный блок включает и блок дисководов.

К компьютеру могут дополнительно подключаться:

печатающие устройства, предназначенные для выдачи информации из компьютера на бумажный носитель;

графопостроитель (плоттер), позволяет выдавать на бумажный лист проектно-конструкторскую документацию, технические чертежи или любую другую графическую информацию; стример – накопитель информации на магнитной ленте;

сканер, позволяет считывать с помощью оптики информацию с листа бумаги в компьютер; световое перо, обеспечивает ввод графической информации с экрана видеомонитора;

графический планшет, обеспечивает ввод в ЭВМ (оцифровку) контурных графических изображений с плоских немагнитных носителей. Используется в системах автоматизированного проектирования для ввода чертежей, а также дизайнерами и художниками для ввода рисунков;

модем (сетевой адаптер), обеспечивает обмен информацией между компьютерами через телефонную сеть;

факс-модем - устройство, сочетающее возможности модема и средства для обмена факсимильными изображениями с другими факс-модемами и с обычными телефаксными аппаратами.

### **Основные технические данные персонального компьютера**

Можно выделить следующие основные технические данные ПК фирмы IBM и совместимых с ними компьютеров:

1) тип процессора: 80386, 80486, 6x86MX, Pentium (P), Celeron фирмы Intel; Athlon, Duron фирмы AMD и др.;

2) тип математического сопроцессора, на старых моделях компьютеров, (Intel 8087, 80287, 80387). Процессоры 80486 DX, 80486 DX2 и старших моделей сами поддерживают операции с плавающей точкой, и поэтому при их использовании математический сопроцессор не требуется;

3) тактовая частота (определяет быстродействие) находится в интервале от 8 МГц, у младших моделей, до 3,8 ГГц, у последних моделей;

4) объем ПЗУ: от 8 Кбайт и выше;

5) объем ОЗУ: от 64 Кбайт до 4 Гбайт;

6) объем внешней памяти: для гибких дисков - 1,44 Мбайт, для жесткого диска - от 20 Мбайт до 400 Гбайт; для оптического диска - до 700 Мбайт, DVD-диска – от 4,7 Гбайт до 17 Гбайт;

7) тип монитора: CGA, EGA, VGA, sVGA, Hercules, жидкокристаллические мониторы.

### **Описание клавиатуры**

В настоящее время в эксплуатации находится значительное число типов клавиатуры, но все они построены по одному принципу. На клавиатуре можно выделить четыре поля:

Таблица 1.2.1 Назначение клавиш на клавиатуре IBM PC/AT

Клавиатура IBM PC/AT	Назначение клавиш
F1...F12	Функциональные клавиши
Caps Lock	Переключение регистра на ввод прописных букв
⇧ Shift	Смена регистра. Переключение на ввод прописных букв без фиксации или на ввод спецсимволов
Tab	Табуляция, перемещение курсора на фиксированное число позиций
Ctrl	Ввод дополнительного кода. Используется в комбинации с другими клавишами
Alt	Альтернативный набор. Используется в комбинации с другими клавишами, а также для ввода символов с кодами 128-255
← Backspace	Возврат на символ. Стирает последний введенный символ
Spacebar	Ввод пробела
Enter	Конец ввода команды, набора текста
Esc	Выход в вызвавшую программу, отказ от операции
Prt Scr (Print Screen)	Печать экрана
Scroll Lock	Блокировка прокрутки
Pause/Break	Пауза, приостанавливает вывод файла на экран. Для продолжения просмотра нажать любую клавишу
Num Lock	Переключение дополнительного поля на ввод цифр или управляющих символов
Ins (Insert)	Включение/выключение режима вставки символов
Del (Delete)	Удаление символа в позиции курсора
End	Переместить курсор в конец файла
Home	Переместить курсор в начало файла
PgUp	Переместить курсор на страницу вверх
PgDn	Переместить курсор на страницу вниз
← ↑ → ↓	Направление перемещения курсора

1) основное (центральное) поле, где расположены клавиши с буквами русского и латинского алфавита, цифрами, специальными символами и некоторые управляющие клавиши;

2) дополнительное поле для ввода цифровой или символьной информации, а также для управления курсором;

3) поле с клавишами для управления курсором и редактирования текста;

4) поле с функциональными и управляющими клавишами.

Назначение клавиш приведено в табл. 1.2.1.

Расположение клавиш на клавиатуре приведено на рис. 1.2.1

Для переключения клавиатуры на ввод русских символов чаще всего используются комбинации клавиш Shift+Shift<sup>1</sup>, Ctrl+Alt, Ctrl+Shift правый, Ctrl+Shift левый, а также клавиша Caps Lock.

<sup>1</sup> Запись Shift+Shift или Ctrl+Shift и т. д. означает: нажать первую клавишу и, не отпуская ее, нажать вторую клавишу.

													1 0 A 0 ↓0								
sc	1	2	F3	F4		F5	F6	F7	F8	F9	F10	F11	F12	PrtScr SysRq	Scrol Lock	Pause Break					
~ ` É	 1	@ 2	#№ 3	\$; 4	% 5	^: 6	&? 7	* 8	( 9	) 0	- _	= +	 \ /	←	Insert	Home	Page Up	Num Lock	/	*	-
Tab	Q Й	W Ц	E У	R К	T Е	Y Н	U Г	I Ш	O Щ	P Ц	{ З	}	]	↵Enter	Delete	End	Page Down	7 Home	8 ↑	9 PgUp	+
Caps Lock	A Ф	S Ы	D В	F А	G П	H Р	J О	K Л	L Д	:	" Ж	' Э	↵Enter	Power	Sleep	Wake	4 ←	5	6 →		
⇧Shift	Z Я	X Ч	C С	V М	B И	N Т	M Ь	< Б	> Ю	?, /.	⇧Shift				↑		1 End	2 ↓	3 PgDn		Enter
Ctrl◆	⌨	Alt							Alt	⌨	⌨	Ctrl◆	←	↓	→	0 Ins		.	Del		

Рис. 1.2.1. Раскладка клавиатуры PC CE

Другие клавишные комбинации:

Alt+PrScr или PrScr - включение/выключение режима копирования на принтер информации, выводимой на экран монитора. В операционной системе Windows эта комбинация клавиш помещает в буфер памяти копию содержимого экрана;

Ctrl+NumLock - приостанавливает выполнение программы, для продолжения выполнения нажмите любую клавишу;

Ctrl+S - приостанавливает выполнение программы;

Ctrl+Break - завершение работы выполняемой программы или команды.

### Включение и выключение компьютера

Для включения компьютера нажмите клавишу Power на системном блоке и кнопку включения питания на видеомониторе. На некоторых моделях компьютеров предварительно необходимо включить выключатель на задней панели компьютера.

В случае зависания компьютера (программа не реагирует на команды, вводимые с клавиатуры или с помощью мыши) выполните "горячий перезапуск" компьютера, нажав комбинацию клавиш Ctrl+Alt+Del. Если это не помогает, нажмите клавишу Reset на системном блоке.

Для выключения компьютера выключите питание видеомонитора, а затем выключите питание компьютера. Современные графические операционные системы обеспечивают автоматическое выключение питания компьютера при выходе из операционной системы. Например, для выключения компьютера в операционной системе Windows 98 необходимо выбрать команду *Пуск, Завершение работы*, активизировать в окне диалога переключатель "Выключить компьютер" и щелкнуть по кнопке ОК. После появления сообщения на экране монитора "Теперь можно выключить питание компьютера" нажмите клавишу Power на системном блоке, а затем выключите питание монитора. В некоторых случаях происходит автоматическое выключение питания при выходе из операционной системы.

### 1.2.2. Задание

1. Изучите описание лабораторной работы. Ответьте на контрольные вопросы.
2. Изучите устройство клавиатуры и назначение клавиш.
3. Создайте текстовый файл, сохраните его на диске.

### 1.2.3. Порядок выполнения работы

1. Включите компьютер.
2. Если компьютер работает в локальной сети, то войдите в сеть:
  - введите логическое имя;
  - введите первичный пароль;
  - ЗАМЕНИТЕ первичный пароль на ВАШ пароль.
3. Загрузите редактор WordPad командой: *Пуск, Программы, Стандартные, WordPad<sup>2</sup>*.

---

<sup>2</sup> Указанная запись означает: щелкнуть мышью по кнопке Пуск, выбрать в главном меню команду Программы, перемещая указатель мыши вправо по строке команды Программы, открыть меню второго уровня, выбрать в нем команду Стандартные, перемещая указатель мыши вправо по строке команды стандартные, перейти в меню следующего уровня, найти в списке команду WordPad и щелкнуть по ней мышью.

4. Изучите клавиатуру.

Переключите клавиатуру на ввод русских символов. Введите прописные и строчные символы русского алфавита. Переключите клавиатуру на ввод латинских символов. Введите прописные и строчные символы латинского алфавита.

Введите цифры и специальные символы на нижнем регистре.

Введите цифры и специальные символы на верхнем регистре.

Наберите произвольный текст (стихотворение, прозу) 10-15 строк.

5. Сохраните набранный текст в файле на диске. Введите команду **Файл, Сохранить как**, введите в строке Имя файла имя файла, например, **Лабораторная работа 1** и щелкните по кнопке **ОК**.

6. Выйдите из редактора: введите команду **Файл, Выход**.

7. Выключите компьютер (выйдите из сети командой **Пуск, Завершение сеанса**).

### 1.2.4. Упражнения

Работы выполняются в редакторе текста WordPad.

1. Введите строку текста. Сотрите ее клавишей BackSpace.

2. Введите строку текста. Сотрите ее клавишей Delete.

3. Введите строку текста. Переместите курсор ввода в начало строки клавишей Home.

Переместите курсор ввода в конец строки клавишей End.

4. Переместите курсор ввода на шаг табуляции вправо (клавиша Tab), влево (клавишами Shift+Tab).

5. Вставьте пробелы в середину строки. Уберите пробелы.

6. Наберите по одной строке каждым спецсимволом.

7. Начертите девять колонок разных спецсимволов на расстоянии шага табуляции.

8. Изобразите произвольный рисунок с использованием цифр и спецсимволов.

### Контрольные вопросы

1. Назовите состав персонального компьютера.

2. Какие поля имеются на клавиатуре и каково их назначение?

3. Как загрузить программу WordPad?

4. Как сохранить информацию на диске?

5. Поясните назначение клавиши Caps Lock.

6. Поясните назначение клавиши Shift.

7. Найдите на клавиатуре клавиши Tab и Esc. Поясните их назначение.

8. Перечислите клавиши управления перемещением курсора.

9. Как осуществляется "горячий" перезапуск компьютера?

10. Как осуществляется перекодировка клавиатуры на ввод русского шрифта?

11. Как вывести на печать содержание экрана?

12. Какими командами можно вывести на печать текстовые файлы?

13. Как ввести режим вставки символов?

14. Какими клавишами можно удалить: а) последний введенный символ; б) символ в позиции курсора?

### 1.3. Работа на печатающем устройстве

**Цель работы:** ознакомиться с назначением и принципом работы печатающего устройства (ПУ), приобрести практические навыки работы с печатающим устройством. Закрепить знание клавиатуры.

**Время:** 1 час.

**Литература:** П2 с. 57 – 58.

#### 1.3.1. Общие сведения

##### Устройство принтера

Печатающие устройства (принтеры) предназначены для получения "твердой копии" документа на бумажном носителе. Они могут выводить текстовую и графическую информацию, а также рисунки, в том числе и цветные.

Наибольшее распространение в настоящее время получили матричные, струйные и лазерные принтеры.

В настоящем пособии приведены сведения о матричных, струйных и лазерных принтерах.

##### Матричные принтеры

Основными узлами матричного принтера являются печатающая головка, бумагоопорный вал, планка прижима бумаги, электронный блок, панель управления, кассета с красящей лентой.

Принтеры имеют два режима печати: символьный и графический. Для настройки режимов печати они содержат переключатели (S1...Sn), число которых зависит от типа принтера.

Управление работой принтера осуществляется с панели управления с помощью сенсорных переключателей и клавиш. Панели управления некоторых типов матричных принтеров приведены на рис. 1.3.1, 1.3.2, 1.3.3.

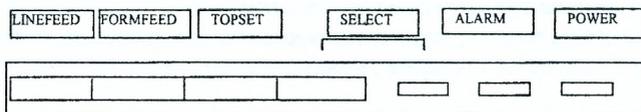


Рис. 1.3.1. Панель управления принтера STAR LC-15

На панелях управления принтеров **STAR** и **MICROLINE** установлены клавишные переключатели, а на панели управления **EPSON** - клавишные и сенсорные переключатели. Клавиши и индикаторы имеют следующее значение:

а) индикаторы:

POWER - питание;

P.OUT (PAPER OUT, ALARM) - отсутствие бумаги;

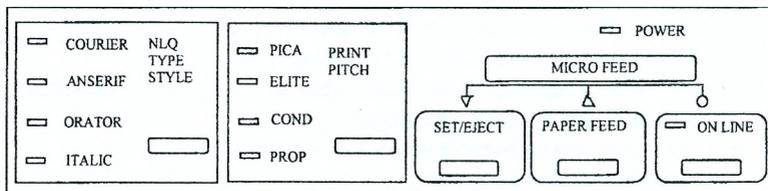


Рис.1.3.2. Панель управления принтера MICROLINE 183

ON/LINE - сигнализирует об установлении связи с ПК (оперативный режим);

READY - готовность принтера;  
 NLQ - качественная печать;  
 COURIER, SANSERIF, ORATOR, ITALIC, ROMAN - тип (рисунок) шрифта, PICA, ELITE, PROP -  
 высота и плотность шрифта, COND(ENSED) - сжатый шрифт, DRAFT - черновой шрифт.

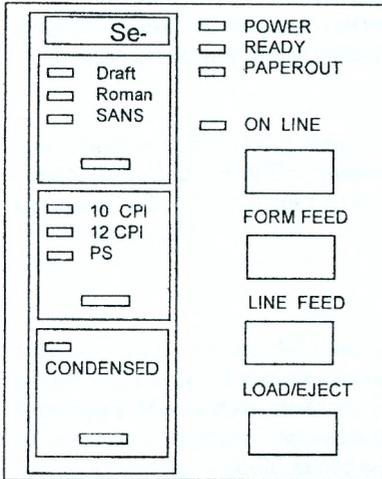


Рис. 1.3.3. Панель управления принтера EPSON FX-1050

SHIFT - задание режимов работы принтера (PITCH - шаг печати, FONT - шрифт (Roman, Sans Serif и др.), MARGIN - границы печати).  
 CONDENSED - плотная печать.

## Работа с матричными принтерами EPSON FX-1050

### Включение принтера

Включите клавишный переключатель на левой боковой стенке корпуса. На панели управления загораются индикаторы POWER, PAPER OUT, ROMAN, 12 CPI, печатающая головка устанавливается в исходное положение.

### Загрузка бумаги

Загрузка осуществляется в полуавтоматическом режиме:

- установите лист на подставку до упора в бумагоопорный вал;
- нажмите клавишу LOAD/EJECT: производится позиционирование печатающей головки, планка прижима бумаги отводится от бумагоопорного вала и производится загрузка листа бумаги. По окончании загрузки бумаги индикатор "PAPER OUT" гаснет.

### Автономная проверка

Автономная проверка включает проверку функционирования клавиш управления перемещением листа на строку и страницу, тест "косой печати" и распечатку установки ключей. Тест косоугольной печати обеспечивает вывод последовательности символов с циклическим сдвигом.

б) клавиши:

FF (Form Feed) - перевод страницы в режиме OFF/LINE ( в режиме отсутствия связи с ПК, автономный режим);

NLQ - качественная печать в режиме ON/LINE. Если клавиша не нажата, осуществляется черновая (DRAFT) печать;

LF (Line Feed)- перевод строки в автономном режиме;

TEST - запуск принтера на внутренний тест при включении питания;

ON/LINE - перевод в режим связи с ПК (LINE) или в автономный режим;

LF/FF - переводит бумагу в автономном режиме на одну строку (LF). Если удерживать клавишу больше одной секунды, то перевод страницы (FF);

LOAD/EJECT - загрузка/выдача листа.

P.PARK/MARGIN - комбинированная подача бумаги/движение печатающей головки влево в режиме FONT SELECTION (выбор шрифта);

Проверка функционирования клавиш управления перемещением листа на строку и страницу выполняется следующим образом:

нажмите клавишу LF (Line Feed) - бумага перемещается на одну строку вверх;

нажмите клавишу FF (Form Feed) - бумага перемещается на одну страницу, при листовом формате осуществляется выдача загруженного листа.

Тесты запускаются клавишами пульта при включении питания (во время начальной установки каретки). После завершения цикла теста печатающее устройство переходит к начальной установке:

Включите питание, загрузите бумагу.

Выключите питание. Нажмите и удерживайте клавишу FORM FEED, включите питание принтера и через несколько секунд отпустите клавишу FORM FEED. Сначала распечатывается состояние ключей, а затем выполняется тест косой печати. Для окончания теста косой печати нажмите клавишу ON/LINE.

## Цветной пузырьково-струйный принтер BJC-210

### Устройство

Цветной пузырьково-струйный принтер BJC-210 (рис.1.3.4) состоит из корпуса (1) с размещенной на нем панелью управления (2), передней крышки (7), устройства подачи бумаги (5) с направляющей бумаги (3), упором для продвижения бумаги (4) и рукояткой выбора типа бумаги (6), печатающей головки, размещенной в корпусе. В состав принтера входят также сетевой адаптер AD-300, черно-белый BC-02 и цветной BC-05 пузырьково-струйные картриджи.

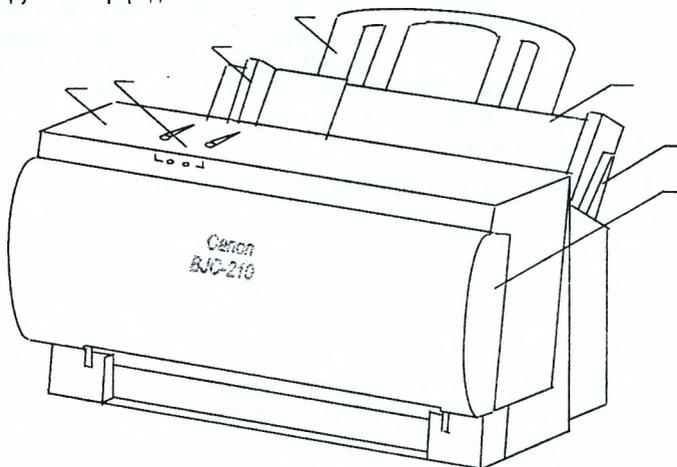


Рис.1.3.4. Струйный принтер

Панель управления позволяет управлять работой принтера вручную: включать его, производить чистку печатающей головки или запускать тестовые распечатки. На панели управления расположены клавиши POWER (сеть) и RESUME (продолжить), а также индикаторы POWER и RESUME/ERROR.

Клавиша POWER предназначена для включения и выключения принтера. При включении питания загорается индикатор POWER.

При выключении принтера печатающая головка автоматически возвращается в исходное положение и закрывается колпачком. Клавиша RESUME выполняет следующие функции:

- повторный запуск принтера при сбое: однократно нажмите и отпустите клавишу для загрузки листа в принтер или перезапуска печати;
- прогон одного уже загруженного в принтер листа: однократно нажмите и отпустите клавишу;
- запуск процедуры чистки печатающей головки: нажмите и удерживайте клавишу не менее двух секунд.

Индикатор RESUME/ERROR сигнализирует об отсутствии бумаги или возникновении замятия бумаги.

Устройство подачи вмещает до 100 листов бумаги стандартного формата и обеспечивает автоматическую подачу или полистную загрузку бумаги. Рукоятка выбора типа бумаги устанавливается в заднее положение в случае использования обычной бумаги и в переднее положение для толстой бумаги.

Передняя крышка обеспечивает доступ к печатающей головке, картриджу, снимается для их обслуживания, замены, а также для извлечения бумаги при ее замятии.

#### **Загрузка пачки бумаги:**

- откройте переднюю крышку: возьмитесь за верхнюю часть крышки по бокам обеими руками и потяните ее слегка на себя;
- переведите рукоятку на узле печатающей головки в требуемое положение: слева - обычная бумага, справа - толстая бумага;
- закройте крышку;
- убедитесь, что рукоятка выбора типа бумаги (6) установлена в требуемое положение, откиньте упор для бумаги (4);
- установите направляющую бумаги (3) по формату бумаги;
- возьмите пачку бумаги и пролистайте ее по загружаемому краю, чтобы не было слипания листов;
- выровняйте бумагу по правому краю устройства подачи и вставьте пачку бумаги в принтер до упора.

#### **Запуск тестовой распечатки:**

- загрузите бумагу формата А4;
  - включите принтер и подождите 15 секунд;
  - нажмите и удерживайте клавишу POWER до тех пор, пока принтер не выдаст один звуковой сигнал, отпустите клавишу;
  - после запуска печати демонстрационной распечатки нажмите клавишу RESUME.
- Демонстрационная распечатка печатается в непрерывном режиме. Для прерывания непрерывной печати следует нажать клавишу RESUME.

#### **Лазерный принтер**

Лазерный принтер еще более прост в управлении, чем струйный принтер.

Например, лазерный принтер Laser Jet – 1100 на задней стенке имеет выключатель, а на передней панели клавишу-индикатор и два индикатора (Рис. 1.3.5.).

Клавиша индикатор сигнализирует о наличии информации для печати при его остановке, а также для включения режима контрольной печати. Если индикатор горит, то для продолжения печати нажмите эту клавишу. Для контрольной печати загрузите в принтер



Рис. 1.3.5. Элементы управления лазерного принтера Laser Jet 1100

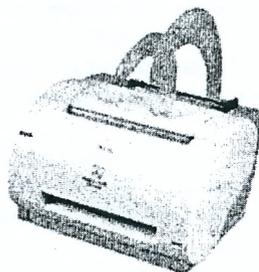


Рис. 1.3.6. Лазерный принтер Canon LBP - 1120

чистый лист бумаги, нажмите и удерживайте клавишу-индикатор 10 – 20 секунд. Левый индикатор служит для контроля наличия бумаги и готовности к печати. При загрузке бумаги индикатор загорается. Правый индикатор сигнализирует о наличии ошибки или отсутствии бумаги. Принтер имеет два "лотка" для закладки листов: первый лоток служит для подачи одиночных листов, задний – для закладки пачки листов.

На рисунке 1.3.6. приведен внешний вид лазерного принтера LBP – 1120. Скорость печати 10 стр./мин.

### Основные технические характеристики принтеров

1. Тип принтера.
2. Характеристика печатающей головки (число игл, сопел).
3. Скорость печати у матричных принтеров достигает от 300 до 1120 зн/с. Скорость печати струйных принтеров от 10 до 20 страниц за минуту. Скорость печати лазерных принтеров находится в интервале от 10 до 50 стр./мин.
4. Объем буфера печати (от 2 Кбайт до нескольких десятков Мбайт).
5. Формат печати: узкий, широкий или указание размера листа).
6. Вид интерфейса: последовательный, параллельный или по заказу и его тип.
7. Возможность использования русского шрифта и количество используемых шрифтов.
8. Совместимость с имеющимися программами по управляющим кодам.
9. Для матричных принтеров имеет значение число печатаемых копий – от 3 до 7.

### 1.3.2. Задание

1. Проверьте принтер в автономном режиме.
2. Создайте текстовый файл, сохраните его на диске.
3. Выведите файл на экран. Выведите на печать содержание экрана.
4. Выведите на печать текстовый файл средствами редактора WordPad.

### 1.3.3. Порядок выполнения работы

1. Изучите описание лабораторной работы и ответьте на контрольные вопросы.
2. Включите компьютер.

3. Подготовьте матричный принтер к работе: включите принтер; загрузите бумагу; установите связь принтера с компьютером.

4. Переведите принтер в автономный режим. Проверьте принтер в автономном режиме.

5. Выведите на печать экран монитора:

– создайте текстовый файл. Наберите 4-5 строк произвольного текста (отрывок из повести, рассказа, песни и т. д.) и сохраните его на рабочем диске (например R:) командой Файл, Сохранить как;

– выведите файл на печать командой Файл, Печать.

6. Выключите компьютер.

7. Ознакомьтесь с работой на струйном принтере: загрузите бумагу; получите демонстрационную распечатку.

8. Ознакомьтесь с работой на лазерном принтере: загрузите бумагу, получите демонстрационную распечатку.

### 1.3.4. Упражнения

1. Включите принтер, загрузите бумагу. Переместите лист на 10 строк вперед. Выполните прогон бумаги.

2. Проверьте принтер в автономном режиме с помощью теста косой печати.

3. Получите распечатку установки ключей.

4. Установите связь ПК с компьютером.

5. Распечатайте документ, подготовленный на первом занятии.

### Контрольные вопросы

1. Какие типы принтеров Вам известны?

2. Как подготовить матричный принтер к работе?

3. Как проверить возможности принтера по выводу шрифтов?

4. Как вывести на печать содержание экрана?

5. Какими командами можно вывести на печать текстовые файлы?

6. Как загрузить бумагу в струйный принтер?

7. Как получить демонстрационную распечатку на струйном принтере?

## 1.4. Сервисная оболочка Far Manager. Знакомство

**Цель работы:** ознакомиться с возможностями программы Far Manager (Far) и приобрести практические навыки в работе с ней: управление панелями, работа с файлами, папками (каталогами), дисками.

**Время:** 2 часа.

**Литература:** Л2 с. 105 – 119.

### 1.4.1. Общие сведения

#### Назначение и запуск программы

Программа Far Manager – это работающая в текстовом режиме программа управления файлами для операционных систем Windows 9x и последующих версий. Она обеспечивает обработку файлов с длинными именами и имеет обширный набор дополнительных функций.

Программа Far Manager может работать с программами, написанными для дисковой операционной системы MS-DOS, и командами операционной системы MS-DOS, также как известные сервисные оболочки Norton Commander, Volkov Commander, ДОО-навигатор и др. Отличительной особенностью Far от названных программ является возможность работы с длинными именами файлов. Кроме того, имеется простой способ изменения языка представления меню и помощи. Для перевода программы на русский язык необходимо нажать клавишу F9, выбрать команду **Options, Languages** и выбрать в меню русский язык для меню и для помощи. Однако не все программы MS-DOS могут быть запущены из ее среды.

Программа имеет большие возможности по работе с файлами. В настоящем пособии рассмотрены только основные приемы работы.

Запуск Far осуществляется из среды Windows как обычное приложение.

Far, как и многие другие сервисные оболочки, является многооконной системой. Используются окна трех видов: панели, окна диалога и окна сообщений. Панели используются для вывода списка файлов и каталогов, текстовых файлов и другой информации. Окна диалога содержат, как правило, строку ввода информации и набор команд для выбора варианта действий. Окна сообщений выводят информацию о состоянии системы и имеют только одну кнопку для закрытия окна. Кроме того, имеется встроенное меню, которое имеет набор команд для настройки среды пользователя и управления компьютером. Меню построено по иерархическому принципу. Первое меню горизонтальное. При выборе требуемого пункта меню открывается вертикальное меню второго уровня, которое может содержать меню следующего уровня.

Для выхода из Far следует нажать клавишу F10. В центре экрана появляется запрос на подтверждение. Вернуться в Windows можно нажатием комбинации клавиш Alt+Tab.

Far имеет систему помощи по работе с программой. Вывод помощи на экран осуществляется нажатием клавиши F1.

### Общие принципы управления

Информация в панелях и меню представляется в виде списка. Для перемещения по списку используются клавиши управления перемещением курсора ←, →, ↑, ↓, а также клавиши Home - установка курсора в начало списка и End - установка курсора в конец списка. Для быстрого перемещения по списку можно использовать также клавиши прокрутки PgUp и PgDn.

Выделить требуемый пункт меню можно также с помощью мыши.

Для ввода команды ее необходимо выделить курсором и нажать клавишу Enter. Нажатие клавиши Enter завершает ввод команды. В дальнейшем изложении материала, когда речь будет идти о вводе команд, клавиша Enter может не упоминаться.

Для завершения ввода команды нажмите клавишу Enter.

Ввод команд можно производить также с помощью "горячих клавиш" - символов, выделенных в команде цветом, жирным шрифтом или символом подчеркивания: нажмите клавишу Alt и, не отпуская ее, нажмите "горячую клавишу".

### Работа в среде Far Manager

#### Описание экрана

После запуска Far на экране появляются два прямоугольных окна, ограниченных двойной рамкой (рис.1.4.1.) Эти окна называют *панелями*. Ниже панелей размещается строка ввода команд с приглашением операционной системы, а в последней строке экрана - справка о назначении функциональных клавиш.

В каждой панели Fag может изображаться:

- оглавление каталога на диске;
- дерево каталогов на диске;
- сводная информация о диске и каталоге на другой панели;
- содержимое файла, выделенного на другой панели.

**Оглавление каталога.** Если в панели Fag выводится оглавление каталога, то в верхней части панели выводится имя этого каталога. Имя каталога текущей панели подсвечено. *Текущей панелью* называется панель, в которой находится курсор панели - инверсная полоса шириной в одну строку экрана. Для перевода курсора в другую панель - *нелекующую* следует нажать клавишу Tab.

В первых строках панели выводятся имена каталогов на диске. Имена каталогов изображаются прописными буквами, справа от имени каталога написано <Папка>. Самую верхнюю строку занимает ссылка на родительский каталог. Для корневого каталога ссылки на родительский каталог нет. В поле имени для родительского каталога изображается двоеточие – символ "..", а справа от него <Вверх>.

Имена файлов выводятся строчными буквами. Для файлов с атрибутами "скрытый" или "системный" имя файла начинается с прописной буквы, а между именем файла и его расширением выводится символ "..." или "A", в зависимости от используемой кодовой таблицы.

C:\		C:\WINDOWS\COMMAND 19:50			
Имя	Имя	Имя	Размер	Дата	Время
Config.Msi	config.sys	..	<Вверх>	25.03.03	23:43
Program Files	detlog.old	EBD	<Папка>	25.03.03	23:43
QBasic	detlog.txt	ansi	sys	9735	05.05.99 22:22
RECYCLED	io.sys	attrib	exe	15428	05.05.99 22:22
System Volume	Info}msdos.--	bootdisk	bat	2080	05.05.99 22:22
Temp	msdos.sys	chkdsk	exe	28544	05.05.99 22:22
VC	netlog.txt	choice	com	5447	05.05.99 22:22
wincmd	ntdetect.com	country	sys	30742	05.05.99 22:22
WINDOWS	ntldr	cscript	exe	90112	26.03.03 18:08
Мои документы	pagefile.sys	cvt	exe	122799	05.05.99 22:22
Пользовательские	scandisk.log	debug	exe	20874	05.05.99 22:22
autoexec.bat	setuplog.txt	deltree	exe	19099	05.05.99 22:22
boot.ini	setupxlq.txt	diskcopy	com	22231	05.05.99 22:22
Temp	<Папка> 02.04.03 09:06	CHKDSK.EXE			
122,247,947 байтов в 26 файлах		1,812,888 байтов в 49 файлах			
C:\>					
1Левая 2Правая 3Смотр 4Редак. 5Печать 6Переи 7Искать 8Истор 9Видео 10Дерево					

Рис. 1.4.1. Рабочий экран программы Fag

Каталог может выводиться в *полной*, *краткой* форме и других формах. При *полной* форме выводятся имя файла, его размер в байтах, дата и время создания. При *краткой* форме выводятся только имя файла. Имеется несколько вариантов вывода информации в панели.

В нижней части панели выводится строка мини-статуса. Она отделена одной горизонтальной чертой. В строке мини-статуса отображается характеристика *выделенного файла* или *каталога*. Файл или каталог считается выделенным, если на него установлен курсор панели.

Для быстрого выделения файла в текущем каталоге, необходимо нажать клавишу *Alt* и, не отпуская ее, нажать любую клавишу. Открывается окно "Поиск". Отпустите клавиши и наберите в командной строке первые буквы имени файла. *Fag* выделит нужный файл, как только будет введено достаточно символов имени файла для его распознавания.

В выделенный каталог можно "*войти*", нажав *Enter*. При этом в текущей панели появится оглавление выделенного каталога. Для *выхода* из текущего каталога в родительский каталог необходимо переместить курсор панели в начало каталога и нажать *Enter*.

**Быстрый поиск файла на диске.** Для быстрого поиска файла во всех каталогах на диске необходимо нажать [*Alt - F7*]. Открывается окно поиска файлов. В строке ввода необходимо набрать имя файла, который требуется найти, и нажать клавишу *Enter* или ввести команду *Искать*. При указании имени файла можно использовать маску. После нажатия клавиши *Enter* состояние окна меняется. В рабочую часть окна выводятся имена найденных файлов, а в нижней части окна поиска выводится меню команд: *Новый поиск* – для повторного поиска, *Перейти* - переход на выделенный файл, *Смотреть* – просмотр выделенного файла, *Панель* – переход в панель где находится выделенный файл, *Отмена* - для отказа от поиска. При поиске по маске на экран выводятся имена всех файлов, удовлетворяющих условию поиска. После окончания поиска выделите нужный файл и введите команду *Перейти* – для перехода к найденному файлу.

### Управление панелями

Панели *Fag* находятся на экране постоянно. При выполнении команд они временно убираются, а затем автоматически восстанавливаются, так что результаты выполнения команды, выведенные на экран, оказываются закрытыми панелями. Для просмотра результатов необходимо убрать одну или обе панели. Управление панелями осуществляется с помощью комбинаций клавиш:

*Ctrl - O* — убрать обе панели с экрана или вывести их на экран;

*Ctrl - P* — убрать одну из панелей (не текущую) с экрана или вывести ее на экран;

*Ctrl - U* — поменять панели местами;

*Ctrl - F1* — убрать левую панель с экрана или вывести ее на экран;

*Ctrl - F2* — убрать правую панель с экрана или вывести ее на экран.

### Переход на другой диск

Для вывода в панель оглавления другого диска необходимо нажать *Alt+F1* для левой панели и *Alt+F2* для правой панели. После нажатия клавиш в панели появляется список доступных дисков. Затем нужно выбрать имя нужного диска клавишами *↑*, *↓* и нажать *Enter*. Для отказа от перехода на другой диск нажмите *Esc*.

В случае, если диск отсутствует, не отформатирован или дефектный, появляется соответствующее сообщение (рис. 1.4.2).

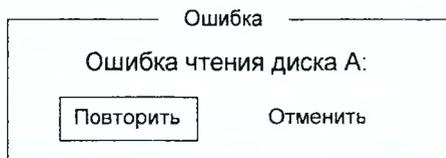


Рис. 1.4.2. Сообщение об ошибке

В случае возникновения таких ситуаций можно повторить операцию, выбрав пункт меню "Повторить" и нажав *Enter*, или отказаться. Команда *Отменить* позволяет окончить ту функцию *Far*, при которой она возникла. Если ошибка возникла при чтении диска, будет выдано сообщение о невозможности выполнить операцию.

При отсутствии неисправностей после нажатия клавиши *Enter* в соответствующую панель выводится каталог выбранного диска.

### **Запуск программ и команд DOS**

Если требуется выполнить программу или команду дисковой операционной системы, необходимо, как обычно, набрать эту команду в командной строке с помощью клавиатуры и нажать *Enter*. Во время выполнения команды панели убираются с экрана, а по окончании выполнения программы восстанавливаются.

Для перемещения по командной строке используются клавиши: *→*, *←*, *Home*, *End*. Названные команды действуют только тогда, когда оглавление текущего диска выведено в полной форме.

Для ускорения набора имени файла в командной строке можно использовать средства *Far*: установите курсор строки ввода команд на свободное место, выберите в текущей панели нужный файл и нажмите *Ctrl+Enter*. Имя выбранного файла помещается при этом в командную строку.

*Far* позволяет выводить в командную строку ранее введенные команды:

*Ctrl+E* — выводит предыдущую введенную команду;

*Ctrl+X* — выводит команду, которая была выполнена после той, что находится в командной строке;

*Alt+F8* — выводит список всех команд (хранится информация о 10...15 последних командах). В этом списке можно выделить и выполнить нужную команду.

Команду, выведенную в командную строку, можно отредактировать и затем исполнить.

Для запуска программ из среды *Far Manager* необходимо выделить нужный файл и нажать *Enter*.

Если при запуске исполняемых файлов (файлы с расширением *COM*, *EXE*, *BAT*) требуется указать другие опции, то следует поступить следующим образом:

- снести выделенный файл в командную строку командой *Ctrl+Enter*,
- дополнить команду необходимой опцией и нажать *Enter*.

### **Использование мыши**

При работе с манипулятором «мышь» *Far* выводит на экран красный прямоугольник - указатель мыши. Его можно перемещать по экрану, двигая мышью по гладкой поверхности, например, по столу. Работа с мышью аналогична работе с клавиатурой. Например, чтобы выделить файл, пункт меню или вариант ответа на запрос, щелкните по нему мышью, то есть кратковременно нажмите и отпустите левую клавишу мыши.

### **Выделение файлов**

Все операции в *Far* выполняются над выделенными файлами или каталогами. Для выделения одного файла или каталога достаточно установить на него курсор. *Far* позволяет выбрать группу файлов, над которой можно выполнить некоторые действия: копирование, перемещение в другой каталог, удаление и т.д. Выбранные файлы выделяются желтым цветом на цветном мониторе и повышенной яркостью на монохромном мониторе.

Выбор файлов можно осуществлять двумя способами: с помощью клавиши *Insert* (Ins)

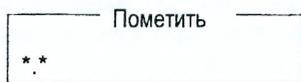


Рис. 1.4.3. Маска ввода

или по маске. В первом случае курсор панели устанавливается на имя требуемого файла и нажимается клавиша *Insert*. Повторное нажатие этой клавиши меняет выделение. Во втором случае поступают следующим образом: нажимают клавишу "+" на дополнительном поле клавиатуры. На экране появляется окно

для ввода маски "Пометить". По умолчанию предлагается маска \*.\* (рис. 1.4.3). Можно указать свою маску, вводя соответствующие символы с клавиатуры. В маске можно использовать символы "\*" и "?".

Чтобы отменить выбор по маске, нажмите "-" на дополнительном поле и задайте маску файлов, выбор которых хотите отменить.

### Встроенное меню

Встроенное меню *Far* позволяет установить наиболее удобный вид представления информации на экране, изменить режим работы *Far*, а также выполнить некоторые другие действия. Для входа в меню следует нажать клавишу F9. Вверху экрана появится строка, содержащая пункты меню *Left, Files, Commands, Option u Right*.

Один из пунктов меню является выделенным. Для выбора нужного пункта меню используются клавиши →, ← или горячие клавиши.

Для выхода из меню или подменю следует нажать клавишу *Esc*.

### Назначение пунктов меню

**Left и Right (Левая, Правая).** Пункты меню "Left" и "Right" задают режим вывода информации соответственно в левой и правой панелях, сортировки, управления просмотром файлов. Действующие режимы отмечены галочкой "√" или "+". Чтобы установить или отменить режим, надо выделить его и нажать *Enter*.

**Files (Файлы).** Пункт меню *Files* дает возможность производить различные операции с файлами: копирование, сортировку, архивацию, установку атрибутов, выделение файлов по маске.

**Commands (Команды).** Пункт меню "Commands" позволяет выполнять различные команды по управлению поиском файлов и каталогов, управления режимами работы.

**Options (Параметры).** Пункт меню *Options* позволяет задавать параметры выполнения команд, панели, конфигурацию программы, режим работы.

### Установка атрибутов файла

Для изменения атрибутов файла необходимо:

- выделить файл или группу файлов;
- ввести команды *F9, Files, file Attributes* (Файлы, Атрибуты файлов);
- установить с помощью клавиши «Пробел» или мыши необходимые флажки;
- ввести команду *Set* (установить).

При изменении атрибутов файлов на экран будет выведено сообщение о текущих атрибутах файла. Знаком "x" будут помечены установленные атрибуты: "Read only" — только для чтения, "Archive" — файл не архивирован, "Hidden" - скрытый файл, "System" - системный файл.

Для отказа от изменения атрибутов файла введите команду *Cancel*.

## Информационная панель

В панели Far можно вывести сводную информацию о диске и каталоге на другой панели. В верхней части информационной панели выводится строка "Information".

В панели отображаются следующие сведения:

- емкость оперативной памяти, компьютера, в байтах (Total memory);
- количество свободной оперативной памяти, в байтах (Free memory);
- емкость текущего диска, в байтах (Total bytes);
- количество свободного места на текущем диске (Free bytes);
- и другие сведения.

Вывод на экран информационной панели осуществляется командой F9, Left, Info panel или нажатием Ctrl - L. Для удаления информационной панели следует повторно нажать Ctrl - L.

### 1.4.2. Задание

1. Изучите панели Far Manager и порядок управления ими.
2. Изучите порядок вывода оглавления дисков в панели, поиска каталогов, входа в каталог и выхода из него.
3. Изучите порядок выделения и поиска файлов на дисках.
4. Изучите порядок установки атрибутов файлов и управления выводом их в панели.

### 1.4.3. Порядок выполнения работы

1. Изучите описание лабораторной работы и ответьте на контрольные вопросы.
2. Изучите панели Far. Найдите левую панель, правую панель, строку мини-статуса, командную строку.
3. Опробуйте команды управления панелями. Уберите правую (левую) панель, обе панели. Поменяйте панели местами. Уберите одну, не текущую панель.
4. Выведите в левую (правую) панель оглавление каталогов дисков: в левую панель оглавление диска C:; в правую панель оглавление диска D:; в правую панель оглавление диска C:; в обе панели оглавление диска D:.
5. Выведите в левую панель оглавление текущего диска в полной форме. Выведите в правую панель оглавление текущего диска в краткой форме в алфавитном порядке расширения имен файлов.
6. Войдите в каталог Windows диска C:. Выйдите из каталога.
7. Перейдите в подкаталог Colors каталога Far\Addons, используя команду быстрого поиска. Выйдите в корневой каталог.
8. Найдите и выделите файлы command.com, graphics.com. на диске C:; скопируйте их на рабочий диск (R:). Отмените выделение.
9. Войдите в каталог Far. Выделите все файлы подкаталога. Запишите в отчет объём, занимаемый этими файлами на диске. Отмените выделение.
10. Выделите все файлы с расширением .TXT в текущем каталоге по маске. Скопируйте эти файлы на рабочий диск.
11. Войдите в подкаталог Office каталога Program Files\Microsoft Office, используя способ быстрого перехода в каталог, и выделите все файлы с расширением .exe.
12. Установите атрибут "скрытый" для файлов с расширением .TXT на рабочем диске.
13. Уберите "скрытые" файлы с экрана. Покажите "скрытые" файлы. Выведите на экран только исполняемые файлы.

#### 1.4.4. Упражнения

1. Уберите левую панель, уберите правую панель. Восстановите панели. Уберите обе панели, восстановите панели. Поменяйте панели местами. Уберите не текущую панель, восстановите панель.

2. Выведите в левую панель оглавление диска C:. Выведите в правую панель оглавление диска D:. Выведите в обе панели оглавление диска C:.

3. Выведите в левую панель оглавление каталога Windows диска C: в полной форме и в алфавитном порядке по времени создания (по объему, по расширению имени файла), используя команды меню Файл, Левая (Правая), Режимы сортировки.

4. Выведите в правую панель оглавление каталога Far диска C: в краткой форме и в алфавитном порядке по расширению имени файла (по времени создания, по объему, в не сортированном виде).

5. Найдите на диске C: (D:, E:) все текстовые файлы, все файлы с расширением COM. Найдите файлы graphics.com, graftabl.com. Выпишите спецификацию этих файлов.

6. Выделите на диске C: все файлы клавишей Ins, отмените выделение. Выделите на диске C: в каталоге VC все файлы с расширением .EXE по маске, отмените выделение по маске.

7. Установите файлам рабочего диска атрибут "спрятанные". Спрячьте эти файлы. Покажите спрятанные файлы. Отмените атрибуты "спрятанные" у файлов рабочего диска.

8. Выведите на панель только исполняемые файлы.

9. Найдите на компьютере все файлы с расширением .TXT, .DOC.

10. Перейдите в каталог VC, выведите на экран файлы с расширением .EXE.

#### Контрольные вопросы

1. Какая панель называется текущей?

2. Как сменить текущую панель?

3. Какой командой убирается правая (левая) панель?

4. Как вывести оглавление диска в правую (левую) панель?

5. Что значит войти в каталог? Как выйти из каталога?

6. Как выделить файлы по маске?

7. Какие атрибуты могут иметь файлы? Как их установить?

8. Как показать спрятанные файлы?

9. Как показать только текстовые файлы?

### 1.5. Работа в среде сервисной оболочки Far Manager

**Цель работы:** приобрести навыки пользования сервисной оболочкой Far Manager.

**Время:** 2 часа.

**Литература:** Л2 с. 105-119.

#### 1.5.1. Использование функциональных клавиш

В Far Manager каждой функциональной клавише поставлена в соответствии определенная команда. Справка о назначении функциональных клавиш приведена в нижней строке экрана.

*F1 - Помощь* — получение справки о программе.

*F2 – Пользм* - меню пользователя — запуск команд, указанных в списке, заданном пользователем.

*F3 - Просм* — просмотр файла.

*F4 - Редакт* — редактирование файла.

*F5 - Копир* — копирование файлов.

*F6 - Перен* — переименование файлов (каталогов) или пересылка файлов в другой каталог.

*F7 - Папка* — создание каталогов (папок).

*F8 - Удален* — удаление файлов и каталогов.

*F9 - КонфМн* — вызов встроенного меню Far.

*F10 – Выход*— выход из Far.

### **F1 - получение справки о программе**

Нажатие клавиши F1 позволяет получить подробную справку о работе с Far Manager. Весь файл помощи разбит на темы. Каждая тема имеет один или несколько экранов помощи.

При нажатии F1. На экране появляется меню помощи, содержащее наименование тем. Выделите нужный пункт меню клавишами и нажмите Enter. На экране появляется текст описания выбранного пункта меню. В тексте справки имеются гипертекстовые ссылки. Они выделены на экране желтым цветом. Если выделить данный текст и нажать клавишу Enter или щелкнуть по нему мышью, то осуществляется переход к описанию соответствующей темы.

Имеется возможность получения контекстной подсказки, то есть подсказки, соответствующей текущему состоянию программы, например, если нажать клавишу F5 копирование файлов, то при нажатии клавиши F1 на экран будет выведена подсказка о команде копирования файлов.

### **F2 - вызов меню пользователя**

При нажатии клавиши F2 на экран выводится меню команд, в которое включают наиболее часто используемые программы, чтобы они были всегда "под рукой". Для запуска той или иной программы выберите нужную команду и нажмите ENTER. Меню помощи может редактироваться пользователем.

### **F3 - просмотр файла**

Клавиша F3 выводит на экран файл, выделенный курсором в текущей панели. Для выхода из режима просмотра нажмите клавишу Esc. Просматривать можно только текстовые файлы и файлы, сохраненные в символьном виде, например, файлы подготовленные программой qbasic и сохраненные с ключом "a"; файлы autoexec.bat, config.sys.

### **F4 - редактирование файлов**

После ввода команды появляется экран встроенного редактора текста. В конце каждой строки набираемого текста необходимо нажимать клавишу Enter. Для сохранения набранного документа на диске нажмите F2. Для сохранения документа под другим именем нажмите Shift+F2. В открывшемся окне наберите имя диска, маршрут, имя файла и нажмите Enter.

Для создания нового файла нажмите комбинацию клавиш Shift+F4. В открывшемся окне запишите имя файла, нажмите Enter.

## **F5 - копирование файлов и каталогов**

Позволяет скопировать файл или группу файлов или каталогов. После нажатия клавиши на экране появляется окно, в котором указано, что копируется и куда. По умолчанию выделенный файл или группа выделенных файлов копируется в каталог, указанный на другой панели. Можно указать свой маршрут и имя файла, набирая его с помощью клавиатуры или выбрав требуемый каталог с помощью дерева каталогов (пункт F10-Tree).

Пример копирования файл `command.com` на диск R:

- выведите в не текущую панель каталог диска R;
- найдите на текущем диске S: файл `command.com` и выделите его;
- нажмите клавишу F5. На экране появляется окно с указанием выполняемой операции. Для копирования нажмите клавишу ENTER. Для отказа от копирования выберите в меню окна команду "Cancel" и нажмите ENTER или нажмите клавишу ESC. При копировании с изменением имени файла наберите в строке ввода после маршрута новое имя файла и нажмите ENTER.

При возникновении каких-либо ошибок на экран выводится окно диалога с соответствующим сообщением, например:

- а) копируемый файл уже имеется на диске:

```
"File already exists C:\aidstest.exe  
Owrite over the old file ?
```

```
Owerwrite All Skip Skip all Append Cancel"
```

( Файл C:\aidstest.exe уже существует.

Хотите изменить старое состояние файла?)

и предлагается набор команд для принятия решения:

Записать, Все, Пропустить, Пропустить все, Добавить, Отмена;

- б) на диске, куда помещается файл, недостаточно места для его размещения:

```
"Insufficient disk space A:\1SBW.DD "  
(Недостаточно места на диске A:\1SBW.DD )
```

- в) диск не готов:

```
" Error on drive A. The disk may not be formatted.  
Retry Cancel"
```

( Сбой на диске A. Диск кажется не форматирован).

## **F6 – пересылка и переименование файлов и каталогов**

Для переименования файла или каталога (папки) необходимо выделить его курсором панели и нажать F6. После нажатия F6 открывается окно, где следует стереть маршрут, набрать новое имя файла или каталога и нажать Enter.

Клавиша F6 используется также для пересылки файлов. В этом случае в окне необходимо указать маршрут, куда пересылаются файлы. Процедура пересылки файлов и каталогов аналогична процедуре копирования, но пересылаемые файлы удаляются из текущего каталога.

## **F7 - создание каталогов**

Перейдите на диск и в каталог (папку), где собираетесь создавать новый каталог, и нажмите F7. Появляется окно, в котором надо набрать имя каталога и нажать Enter. Для отмены создания каталога нажмите Esc. Если программа не может создать указанный каталог, то выдаётся сообщение "Can't create directory" ("не могу создать каталог").

## **F8 - удаление файлов и каталогов**

Для удаления файла или каталога необходимо выделить его и нажать F8. На экран выводится предупреждение и запрос на подтверждение удаления, например:

" The file is read only Pc(60).bak

Do you wist to delete it?

Delete All Skip Skip all Cancel"

(Файл только для чтения Pc(60).bak

Вы желаете удалить его

Удалить Все Пропустить Пропустить все Отменить)

После выбора пункта "Удалить" и нажатия клавиши Enter выбранный файл удаляется. Если удаляется группа файлов, скрытые или системные файлы, то запрос на подтверждение удаления повторяется дважды. Можно нажать Enter для удаления или Esc для отказа.

После нажатия клавиши Enter или Esc программа возвращается к состоянию, которое было до нажатия клавиши F8.

## **F9 - Вызов собственного меню Far Manager**

Собственное меню Far позволяет управлять выводом информации в панели, выполнять команды и настраивать режимы работы самого Far.

## **F10 - Выход из программы**

После ввода команды появляется запрос на подтверждение выхода. Для выхода выберите команду "Y", для отказа - "N".

### **1.5.2. Задание**

**Внимание.** Все операции по пересылке, копированию и удалению файлов выполнять на рабочем диске, например, R:.

1. Получите помощь о назначении команды Copy. Выпишите в отчет порядок копирования файлов.

2. Найдите на диске C: текстовые файлы и просмотрите их. Просмотрите файлы autoexec.bat и config.sys.

3. Создайте текстовый файл на текущем диске.

4. Отредактируйте текстовый файл.

5. Создайте на диске R: дерево каталогов: CAT-1, CAT-2, CAT-3.

6. Скопируйте в каталог CAT-2 файлы print.com и tree.com из каталога DOS (DOS7, DOS330).

7. Переименуйте файлы print.com и tree.com в каталоге CAT-2.

8. Скопируйте в каталог CAT-3 часть файлов каталога Far.

9. Перешлите файлы с расширением .EXE из каталога CAT-3 в каталог CAT-1.

10. Удалите с диска R: каталог CAT-3.

### **1.5.3. Упражнения**

1. Скопируйте на диск R: все файлы с расширением .SYS из каталога DOS7 диска S:.

2. Переименуйте файл ansi.sys на диске R:

3. Создайте в корневом каталоге диска R: каталоги FAIL-1, FAIL-2, в каталоге FAIL-2 создайте каталог FAIL-3.

4. Перешлите в каталог FAIL-1 файлы из корневого каталога диска R:.
5. Скопируйте в каталог FAIL-2 исполняемые файлы из каталога DOS330 диска S:.
6. Перешлите файлы graphics.com и graftabl.com из каталога FAIL-2 в каталог FAIL-3.
7. Удалите файлы из каталога FAIL-1 и FAIL-3.
8. Удалите каталоги FAIL-1 и FAIL-3.
9. Установите атрибуты "скрытый" для файлов в каталоге FAIL-2. Выведите в левую панель оглавление диска R:. Покажите скрытые файлы.
10. Найдите в компьютере текстовые файлы и скопируйте часть их в каталог FAIL-2.
11. Отмените атрибуты "скрытый" у файлов в каталоге FAIL-2. Выведите в правую панель оглавление диска R:. Установите режим вывода в панель только исполняемых файлов.
12. Выведите в правую панель оглавление диска D: в краткой (полной) форме.
13. Удалите с диска R: все файлы и каталоги.
14. Создайте текстовый файл на диске R:.

### Контрольные вопросы

1. Перечислите назначение функциональных клавиш.
2. Как произвести копирование файла, группы файлов?
3. Как осуществляется пересылка файлов?
4. В чем отличие процедуры пересылки от процедуры копирования файлов?
5. Какие файлы можно просматривать на экране монитора?
6. В чем состоит особенность удаления группы файлов?

## 1.6. Знакомство с Windows

**Цель работы:** ознакомиться с назначением и устройством "Рабочего стола" операционной системы Windows. Изучить приемы навигации в системе.

**Время:** 2 часа.

**Литература:** Л1 с. 83 – 104, Л2. с. 119 – 122.

### 1.6.1. Краткие теоретические сведения

Размещение элементов управления на рабочем столе показано на рис. 1.6.1.

Панель задач предназначена для размещения значков запущенных задач, *Панели быстрого запуска*, *Панели адресов*, *Панели каналов (ссылки)*, *Панелей пользователя*, а также кнопки *Пуск* и *Панели Индикации*. Она размещается обычно внизу рабочего стола, но может быть установлена пользователем в любое положение: зацепите *Панель Задач* мышью и переместите в нужное положение.

*Кнопка ПУСК* открывает доступ к командам *Главного меню*. Чтобы изучить возможности Windows, достаточно просмотреть внимательно пункты меню.

*Программы* – позволяет найти и запустить любую программу. Запуск программ осуществляется двойным щелчком мыши по соответствующему пункту меню;

*Документы* - хранит список 15 документов, обрабатывавшихся на компьютере в последнее время;

*Настройка* - открывает доступ к программам настройки компьютера: *Панель управления*, *Принтеры*, *Панель задач* и *меню ПУСК*, *свойства папки* и *рабочий стол*;

*Найти* – позволяет осуществлять поиск на компьютере файлов, папок, людей;

*Справка* - открывает доступ к справочной системе Windows.

*Выполнить* – запуск программ через строку ввода.

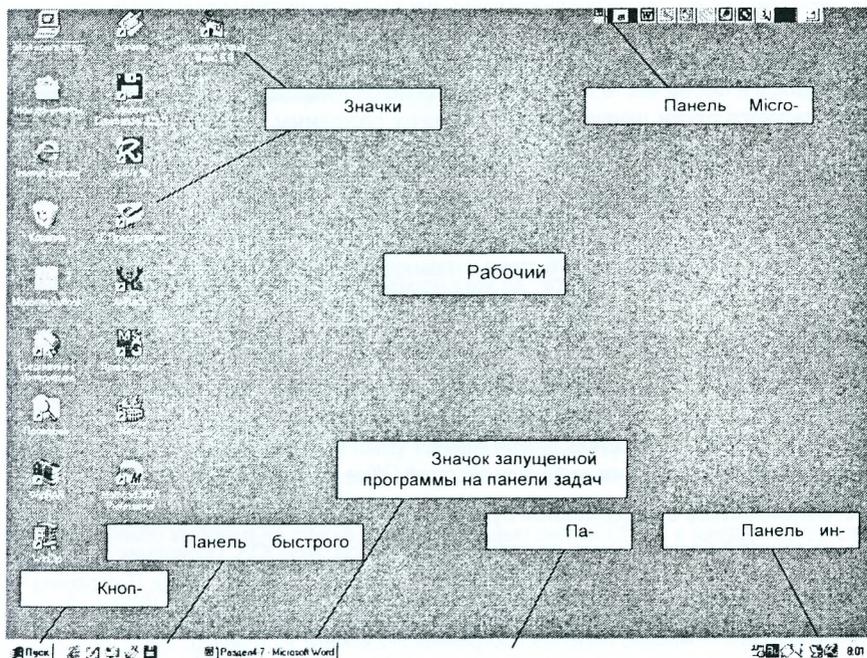


Рис.1.6.1. Рабочий стол

Панель индикации - служит для отображения текущего времени, раскладки клавиатуры, регулятора громкости, настройки разрешающей способности экрана монитора и др. Если подвести указатель мыши к значку в панели индикации и задержать его на некоторое время (*зависнуть*), то высвечивается либо дополнительная информация, либо открывается меню настройки свойств выделенного объекта.

*Чтобы узнать назначение любого объекта в Windows зависните на нем мышью. Через некоторое время появится всплывающая подсказка.*

Панель быстрого запуска – используется для размещения наиболее часто используемых программ.

Панель каналов – служит для размещения узлов Интернета, на которые можно осуществить "подписку" на услуги канала. Тогда в установленное время на компьютер будет сбрасываться новая информация.

Панель адресов - сюда можно помещать значки узлов Интернета, к которым приходится обращаться наиболее часто, чтобы они были всегда под рукой.

На панель задач выводятся значки всех запущенных программ. Поэтому легко перейти от одной программы к другой щелчком мыши по соответствующему значку.

На рабочем столе размещаются значки программ и ярлычки – "заместители программ". Ярлычки содержат ссылки на адреса программ и позволяют запускать программы двойным щелчком мыши.

Панель Microsoft Office – предназначена для облегчения запуска офисных приложений Windows: редактора текста Word, электронной таблицы Excel, системы управления базой данных Access, системы подготовки презентаций Power Point и др. Панель может размещаться в любом месте по периметру рабочего стола. Запуск приложений с панели осуществляется одним щелчком мыши по значку программы. На панель можно добавлять кнопки или удалять их. В принципе на панель Microsoft Office можно поместить значок любой программы.

### **1.6.2. Задание**

1. Изучите элементы рабочего стола и главного меню программы.
2. Изучите программу Проводник и приемы работы в ее среде.

### **1.6.3 Порядок работы**

#### **1. Изучите элементы рабочего стола (рис. 1.6.1):**

- какие значки и ярлычки расположены на рабочем столе;
- расположение и назначение панели задач;
- расположение и назначение панели Microsoft Office.

#### **2. Ознакомьтесь с содержанием пунктов меню Программы**

- Найдите в Главном меню программы Word, Excel, MS-DOS, Проводник: выберите команды Пуск, Программы, переместите указатель мыши вправо по строке Программы – откроется меню второго уровня. Найдите в этом меню значки указанных программ.
- Перетащите мышью ярлычок программы *Проводник* из главного меню на Панель Microsoft Office.

#### **3. Изучите порядок управления окнами**

- Запустите программу Проводник: щелкните мышью по значку программы Проводник на панели Microsoft Office или в Главном меню, или в контекстном меню кнопки Пуск.
- Измените размеры окна и его местоположение. Для перемещения окна зацепите его мышью за заголовок и перемещайте в нужном направлении. Для изменения размеров окна зацепите границу окна мышью (подведите указатель мыши к границе окна так, чтобы он превратился в двунаправленную стрелку) и переместите в нужном направлении.
- Разверните окно программы во весь экран. Восстановите первоначальные размеры окна. Сверните окно программы. Разверните окно программы.
- Откройте второе окно программы Проводник, используя контекстное меню кнопки Пуск.
- Разместите оба окна программы Проводник на рабочем столе один возле другого путем перемещения и изменения их размеров.

#### **4. Изучите программу Проводник**

- Изучите содержание пунктов главного меню Файл, Правка, Вид, Сервис программы Проводник.
- Изучите назначение кнопок на стандартной панели инструментов.
- Изучите структуру папок в левой панели. Откройте папки щелчком мыши по кнопкам "+". Закройте папки щелчком мыши по кнопке "-".

- Выведите в правую панель содержание диска C: (щелкните мышью по значку диска C: в левой панели).
- Узнайте размер диска C: . Вызовите контекстное меню, щелкнув правой кнопкой мыши по имени диска, и выберите команду Свойства.

### Управление представлением информации в окне программы

- Измените метод представления файлов в окне программы (Крупные значки, Мелкие значки, Список, Таблица) используя команды меню и кнопки панели инструментов.
- Отсортируйте файлы в окне программы по имени, типу, размеру и дате создания файла, с помощью команд меню.

Выберите способ представления информации на экране, предпочтительный с Вашей точки зрения.

### Поиск информации и управление папками:

- Просмотрите дерево папок и выпишите имена дисков на Вашем компьютере.
- Найдите на диске C: и разверните папку Windows. Узнайте, имеет ли данная папка вложенные папки. Сверните папку Windows, сверните диск C:.
- Найдите папку Windows, используя команду Поиск. Введите команду Сервис, Поиск, Файлы и папки. Введите в строке ввода "Имя" имя папки, а в строке "Где искать" имя диска либо выбором из списка, либо с помощью кнопки Обзор и щелкните по кнопке Найти. По окончании поиска выпишите полный путь к папке.
- Найдите на диске C: любой файл и выпишите его имя. Перейдите на другой диск. Найдите файл, имя которого Вы выписали, используя команду Поиск. Выпишите полный путь к файлу.
- Найдите на компьютере файл excel.exe. Выпишите полный путь к файлу. Просмотрите свойства файла: выделите файл, вызовите контекстное меню, выберите команду Свойства.

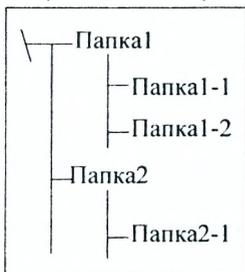


Рис. 1.6.2. Дерево папок

### Создание папок

- Выведите в панель оглавления рабочего диска (будем полагать, что имя рабочего диска R:). Создайте на диске дерево папок следующей структуры (рис. 1.6.2) : Папка1, Папка1-1, Папка1-2, Папка2, Папка2-1. Где Папка1 и Папка2 – папки первого уровня, находятся в корневом каталоге диска, Папка1-1 и Папка1-2 – папки второго уровня, размещены в Папке1, Папка2-1 – папка второго уровня, расположена в папке Папка2.

Порядок работы:

- перейдите на диск R:
- введите команду **Файл, Создать, Папку** или откройте

контекстное меню щелчком по правой панели программы Проводник и выберите команду **Создать, Папку**,

- введите вместо имени по умолчанию "Новая папка" имя Папка1;
- откройте Папку1 и создайте в ней Папку1-1;
- создайте аналогичным образом остальные папки.

- Разверните дерево папок диска R: и проверьте правильность созданной структуры папок.

## Копирование файлов и папок

1. Скопируйте произвольные файлы с диска С: на рабочий диск в Папку1, используя команды меню Правка или кнопки панели инструментов:

- а) введите в правую панель содержание диска С:;
- б) выделите несколько файлов (при выделении группы файлов нажмите и удерживайте клавишу Ctrl) ;
- в) введите команду **Правка, Копировать** или щелкните по кнопке **Копировать** на панели инструментов. Файлы помещаются в буфер обмена программы Windows;
- г) выведите в правую панель содержание Папки1 диска R:;
- д) введите команду **Правка, Вставка**. Файлы копируются из буфера обмена в папку диска R:.

2. Скопируйте произвольные файлы с диска С: в папку Папка2-1, используя контекстное меню:

- выполните пункты 1.а и 1.б;
- щелкните правой кнопкой мыши по выделенным файлам, выберите в контекстном меню команду **Копировать**. Файлы помещаются в буфер обмена программы Windows;
- выведите в правую панель содержание Папки2-1 диска R:;
- щелкните правой кнопкой мыши по правой панели программы Проводник, выберите в контекстном меню команду **Вставить**. Файлы копируются из буфера обмена в папку диска R:.

3. Скопируйте произвольные файлы с диска С: в папку Папка1-1, используя способ перетаскивания мышью:

- выполните пункт 1.а;
- откройте в левой панели программы Проводник структуру диска R:, щелкая мышью по кнопкам "+";
- выполните пункты 1.б;
- зацепите выделенные файлы мышью, перетащите их на Папку1-1 (при попадании на папку указателя мыши она выделяется синим цветом) и отпустите клавишу мыши.

## Пересылка файлов

- Перешлите файлы из папки Папка1 в папку Папка1-2.
- Пересылка файлов выполняется так же как и операция копирования путем перетаскивания мышью, но с использованием правой клавиши. после отпускания мыши открывается окно диалога, которое предлагает выбрать операцию: копировать, переместить, создать ярлыки или отказаться.

## Переименование файлов и папок

- Переименуйте один из файлов в папке Папка2-1:
  - перейдите в Папку2-1 и выделите файл, имя которого следует изменить;
  - введите команду Правка, Переименовать или команду Переименовать контекстного меню;
  - введите новое имя файла.

## Удаление файлов и папок

- Удалите файлы из одной папки и одну папку диска R:  
Для удаления папки (файла) выделите ее и нажмите клавишу Delete или воспользуйтесь командой Удалить из меню Правка или контекстного меню.

### Поиск файлов и папок

- Найдите файлы `autoexec.bat` и `Config.sys` на диске С: и в компьютере. Время поиска может занять много времени:
  - введите команду **Поиск, Файлы и папки** меню **Сервис** программы Проводник или команду **Найти, Файлы и папки** Главного меню;
  - введите в строке "Имя" имя файла – `autoexec.bat`, укажите в строке "Где искать" С:\ и щелкните по кнопке **Найти**.
  - повторите предыдущую операцию, но местом поиска укажите "Мой компьютер".
- Выполните аналогично поиск файла `Config.sys`.
- Закройте программу Проводник
- Завершите работу

### Контрольные вопросы

1. Какие объекты расположены на рабочем столе?
2. Для чего предназначена Панель задач? Где она расположена?
3. Каково назначение панели Microsoft Office?
4. Как запустить программу Проводник?
5. Для чего предназначена программа Проводник?
6. Как создать новую папку?
7. Что такое дерево папок и как его просмотреть?
8. Как узнать свойства папок или файлов?
9. Расскажите алгоритм копирования файлов и папок.
10. Как найти нужный файл или папку?

## 1.7. Настройка параметров рабочего стола

**Цель работы:** приобрести первичные навыки по настройке параметров рабочей среды по интересам пользователя.

**Время:** 2 часа.

**Литература:** Л1 с. 91-104, Л2. с. 103 – 105.

### 1.7.1. Задание

1. Упорядочите размещение значков на рабочем столе.
2. Создайте на рабочем столе ярлыки файлов и папок.
3. Настройте свойства Панели задач.
4. Настройте свойства клавиатуры и мыши.
5. Настройте параметры рабочего стола: фон, заставку, разрешение экрана.

### 1.7.2. Порядок работы

#### 1. Управление размещением значков на рабочем столе

- переместите значки мышью в произвольное положение на рабочем столе;
- проведите упорядочивание значков: вызовите контекстное меню рабочего стола;
- выберите в меню команду **Упорядочить значки**, а затем выберите способ упорядочивания: по имени, размеру, типу или по дате. Если выбрать команду автоматически, то перемещение значков станет невозможным.

#### 2. Создание ярлыков файлов и папок

- создайте на рабочем диске папку "Учебная";
- создайте с помощью программы WordPad файл "Упражнение1" и сохраните его на диске:

- введите команду Пуск, Программы, Стандартные, Word Pad;
  - наберите произвольный текст (две – три строки);
  - сохраните файл на рабочем диске в папке Учебная командой Файл, Сохранить как;
  - найдите файл Упражнение1; ;
  - вызовите контекстное меню для файла и выберите команду **Создать ярлык**. Ярлык появится в той же папке, где находится и файл;
  - выделите ярлык и перетащите его на рабочий стол.
- Другой способ создания ярлыка на рабочем столе:
- вызовите контекстное меню рабочего стола;
  - выберите команду **Создать, Ярлык** – открывается диалоговое окно мастера создания ярлыков;
  - выберите с помощью кнопки Обзор файл Упражнение1, введите имя для ярлыка, выберите пиктограмму и завершите работу щелчком по кнопке Готово.

### 3. Настройка свойств Панели задач

- Вызовите контекстное меню панели задач.
- Выберите команду Свойства. В окне диалога Свойства панели задач имеется четыре флажка, два из которых установлены по умолчанию: “Расположить поверх всех окон” и “Отображать часы”. Флажок “Мелкие значки в главном меню” позволяет изменить значки программ в главном меню Windows. Если установлен Флажок “Автоматически убирать с экрана” то при запуске любого приложения панель задач убирается с экрана, освобождая место для приложения. Чтобы теперь вызвать панель задач, подведете указатель мыши к нижнему краю экрана.
- Проверьте действие флажков.

### 4. Настройка параметров клавиатуры

- Введите команду **Пуск, Настройка, Панель управления, Клавиатура**. Открывается окно диалога Свойства. Клавиатура. Окно имеет две закладки: “Скорость” и “Язык”. Закладка “Скорость” позволяет настроить интервал времени между появлением повторных символов при удержании клавиши на клавиатуре. Закладка “Язык” позволяет выбрать язык, используемый по умолчанию, а также добавить новые языки. Добавление нового языка возможно при наличии дистрибутивной дискеты. Обратите внимание на флажок “Отображать индикатор языка на панели задач”.
- Проверьте действие команд. Измените язык, используемый по умолчанию.

### 5. Настройка параметров мыши

Для настройки параметров мыши введите команду **Пуск, Настройка, Панель управления, Мышь**.

Отрегулируйте скорость двойного щелчка мыши.

### 6. Настройка параметров рабочего стола

Вызовите контекстное меню рабочего стола.

- Выберите команду Видеорежимы, выберите схему настройки цвета и измените разрешающую способность экрана (предварительно запишите установленные режимы, они отмечены галочками). Восстановите настройки по умолчанию.
- Выберите команду **Рабочий стол Active Desktop, Настройка рабочего стола**:
  - ИЗМЕНИТЕ ФОНОВЫЙ РИСУНОК;
  - измените экранную заставку;
  - измените раскладку клавиатуры.

## Контрольные вопросы

1. Как убрать с экрана панель задач?
2. Как изменить раскладку клавиатуры по умолчанию?
3. Как добавить другой язык, например, белорусский?
4. Как изменить разрешение экрана?
5. Замените фоновый рисунок.
6. Измените экранную заставку.
7. Настройте скорость двойного щелчка мыши.

## 1.8. Знакомство с Windows Commander. Архивация данных

**Цель работы:** приобрести начальные навыки работы в среде Windows Commander и архивации данных.

**Время:** 4 часа.

**Литература:** Л1 с. 104 – 110, 110 – 114, Л2 с. 119 – 121, 122-129.

### 1.8.1. Задание

1. Изучите рабочую среду Windows Commander.
2. Изучите назначение функциональных клавиш.
3. Изучите архивацию файлов.

### 1.8.2. Порядок работы

#### 1. Изучение рабочей среды Windows Commander

Для работы с файлами в Windows Commander используются практически те же команды, что и в Far Manager.

1. Загрузите программу командой **Пуск, Программы, Windows Commander**.
2. Переведите программу на русский язык:
  - выберите в главном меню команду **Configuration, Options, Languages**, выберите в списке язык – **Russian** и щелкните по кнопке ОК.
3. Изучите содержание пунктов главного меню.
4. Выведите в левую панель оглавление диска C:. Откройте список дисков над левой панелью и выберите в списке диск “-C-“. Выведите в правую панель оглавление диска D:. Выведите в обе панели оглавление диска C:.
5. Выведите в левую панель оглавление каталога Windows диска C: в полной форме и в алфавитном порядке по времени создания (по объему, по расширению имени файла), используя команды меню Вид.
6. Выведите в правую панель оглавление каталога VC диска C: в краткой форме и в алфавитном порядке по расширению имени файла (по времени создания, по объему, в не сортированном виде).
7. Найдите на диске C: (D:, E:) все текстовые файлы (расширение \*.txt). Просмотрите текстовые файлы: выделите файл и нажмите клавишу F3.
8. Найдите на диске C: (D:, E:) все файлы с расширением COM. Выпишите спецификацию двух любых файлов. Найдите эти файлы с помощью команды поиска (Alt+F7).
9. Перейдите в каталог VC, выведите на экран файлы с расширением .EXE.
10. Выделите на диске C: все файлы клавишей Ins, инвертируйте выделение с помощью команды **Инвертировать выделение** меню **Выделение**. Отмените выделение командой **Снять всё выделение**. Выделите на диске C: в каталоге VC все файлы с расширением .EXE по маске, отмените выделение по маске командой **Снять выделение**.

**Внимание:** все команды относятся только к текущему диску.

## 2. Изучение назначения функциональных клавиш

11. Создайте на рабочем диске папку "Практика" командой **Создать каталог** (клавиша F7).
12. Скопируйте в эту папку несколько файлов с диска С:. Операция копирования выполняется так же, как в Far Manager, с помощью клавиши F5.. Можно копировать файлы также перетаскиванием мышью, как в Windows.
13. Установите файлам диска R: атрибут "скрытый". Выделите файлы и введите команду **Файл, Изменить атрибуты**, установите флажок "Скрытый".
14. Спрячьте файлы с атрибутом *скрытый*: введите команду **Конфигурация, Настройка, Экран**; снимите флажок **"показывать скрытые/ системные файлы"**. Покажите скрытые файлы. Отмените атрибуты "скрытый" у файлов диска R:.
15. Выведите на панель только исполняемые файлы. Введите команду **Вид, Фильтр** и установите маску \*.com; \*.exe. Отмените фильтр, установив маску \*.\*.

## 3. Архивация и восстановление файлов

16. Заархивируйте все файлы из папки "Практика", архивный файл поместите в эту же папку:
  - выделите все файлы папки "Практика" по маске;
  - введите команду **Файл, Упаковать**, выберите программу-архиватор, например ZIP, укажите в строке ввода путь R:\Практика\практика.zip и щелкните по кнопке ОК.
17. Удалите из папки "Практика" все файлы, кроме архивного. Извлеките файлы из архива в папку "Практика": выделите архивный файл, введите команду **Файл, Распаковать**, укажите в строке ввода путь R:\Практика\ и щелкните по кнопке ОК.
18. Выполните работы по п. 16 – 17 с помощью программы WinRar.

## Контрольные вопросы

1. Как переключить программу на русский язык?
2. Как вывести оглавление диска в правую (левую) панель?
3. Как выделить отдельные файлы? Как выделить файлы по маске?
4. Как отсортировать файлы по имени (по типу, по дате, по размеру)?
5. Как создать каталог, дерево каталогов?
6. Что такое фильтр, для какой цели он используется?
7. Как установить/изменить атрибуты файла?
8. Как найти текстовые файлы на диске?
9. Как выполняется архивация файлов средствами Windows Commander?
10. Как восстановить архивные файлы средствами Windows Commfinder?
11. Как скрыть/показать скрытые/системные файлы?
12. Как выполнить архивацию файлов с помощью программы WinRar?
13. Как восстановить архивные файлы с помощью программы WinRar?

## 2. МАТЕМАТИЧЕСКАЯ СИСТЕМА MATHCAD

### 2.1. Организация вычислений и типы данных

**Цель работы:** познакомиться с интерфейсом Mathcad. Овладеть навыками ввода и корректировки математических выражений. Изучить встроенные математические функции Mathcad и их применение при расчетах значений математических выражений для действительных и комплексных чисел. Освоить формы представления результатов вычислений.

**Время:** 2 часа.

**Литература:** Л1 с.121-179.

#### 2.1.1. Задание 2.1

1. Вычислить заданное арифметическое выражение и результат представить в виде смешанной дроби.

2. Вычислить заданное арифметическое выражение и результат представить в виде десятичной дроби.

3. Вычислить заданное арифметическое выражение и выбрать формат для более точного представления результата, путем сравнения результатов в виде смешанной дроби и десятичной дроби. Если для обоих случаев получаем одинаковые значения, то итоговый результат представить в виде десятичной дроби.

4. Вычислить значение функции при заданных значениях аргумента и результат представить с точностью до 8 знаков после десятичной точки.

5. Вычислить значение двух заданных функций. Задать для обеих функций произвольное, но одинаковое значение аргумента (значения функций, при одинаковых значениях аргумента, должны быть равными).

6. Вычислить значение тригонометрической функции при задании значения аргумента в градусах.

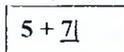
7. Упростить выражение с комплексными числами. Выделить действительную и мнимую часть результата, определить значение угла между радиусом вектором комплексного числа и осью  $x$ . Найти значение комплексного числа, сопряженное результату вычисления, и модуль результата.

#### 2.1.2. Указания к выполнению задания

Задачи 1, 2 и 3 преследуют цель выработать навыки ввода, корректировки и отображения результатов вычислений с требуемой точностью и в нужных форматах.

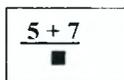
При вводе математических выражений следует учитывать особенности Mathcad, которые заключаются в том, что вычисления выполняются в строго определенном порядке: слева направо и сверху вниз. Ввод выражений осуществляется в блоки. Каждый блок занимает на экране некоторое пространство, ограниченное прямоугольной областью. Указанный выше порядок вычисления относится и к блокам. Блоки могут быть трех типов: текстовые, вычислительные и графические. Текстовые блоки играют роль неисполняемых комментариев. Они служат лишь для повышения наглядности документа. Вычисляемые блоки состоят из исполняемых математических выражений (формул, уравнений, равенств, неравенств и т.д.). Блоки не должны налагаться друг на друга.

Ввод формул в Mathcad имеет некоторую специфику. Например, необходимо сложить два числа 5 и 7 и разделить их на сумму чисел 1 и 3. Сначала введем выражение  $5 + 7$ . На рабочем поле Mathcad в блоке ввода это выглядит следующим образом.

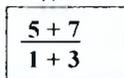

$$5 + 7|$$

При вводе курсор ввода в виде вертикальной черты охватывает последний операнд. Теперь надо ввести знак деления « / ». Однако если это сделать сразу же и добавить знаменатель  $1 + 3$ , то получим выражение  $5 + \frac{7}{1 + 3}$ . Для правильного ввода необходимо первоначально выделить часть выражения, которое будет представлять числитель. Выделение производится нажатием клавиши **SPACE BAR (Пробел)**, до тех пор пока все выражение, представляющее числитель, не будет выделено.

Затем набирается знак деления « / »


$$\frac{5 + 7}{\blacksquare}$$

И в область ввода, представленную в виде  $\blacksquare$  вводится знаменатель. В результате получаем требуемое выражение


$$\frac{5 + 7}{1 + 3}$$

Можно воспользоваться скобками, введя выражение числителя и заключив его в скобки ( $5 + 7$ ). Тогда Mathcad автоматически сформирует выражение в числителе. Однако не рекомендуется вводить лишние скобки там, где в этом нет необходимости. Простейшие вычисления можно выполнить, используя знак равенства «=», что соответствует схеме: **выражение** = . В левой части равенства могут стоять любые математические уравнения, содержащие встроенные в систему функции. В нашем случае итоговое выражение будет выглядеть следующим образом.  $\frac{5 + 7}{1 + 3} = 3$ .

Для отображения результата в нужном формате необходимо установить курсор ввода на результат, который нужно форматировать, активизировать диалоговое окно **«Формат результата»**, путем меню **Формат, Результат**, выбрать нужный формат (десятичный, дробный и т. д.) и ввести его параметры: количество десятичных точек, смешанный формат для дроби и т. д.

В задачах 4,5,6 используются переменные и функции. Чтобы в Mathcad присвоить переменной какое-либо значение, используется оператор присваивания «:=». При вводе условия задачи на это необходимо обратить внимание. В процессе вычислений используются встроенные функции и функции, задаваемые пользователем. Для облегчения ввода встроенных математических функций можно воспользоваться кнопкой **f(x)**, которая выводит диалоговое окно с полным набором функций, разбитых на группы. В таблице 2.1. приведены наиболее часто используемые функции Mathcad. Несмотря на то, что форма записи математических выражений в Mathcad максимально приближена к стандартной математической, есть ряд особенностей, которые необходимо учитывать. В таблице 2.2. приведены некоторые фрагменты математических выражений и форма их записи в Mathcad. Для вычисления значения функции пользователя необходимо сначала

ла задать функцию, а затем вычислить ее значение при заданных значениях переменных. В Mathcad функция задается в следующем виде:

**NAME (список параметров):= выражение**, где NAME – имя функции пользователя; список параметров – перечень переменных, являющихся аргументами данной функции; выражение – математическое выражение, задающее нужную функциональную зависимость. Mathcad имеет множество встроенных функций. При их вводе необходимо соблюдать их точное синтаксическое написание. Например,  $tg\ x \rightarrow \tan(x)$ ,  $arctg\ x \rightarrow atan(x)$ . В случае неправильного задания имени функции, выводится сообщение об ошибке или получается неверный результат.

Для отображения результатов вычислений с требуемой точностью (задача 4), требуется активизировать диалоговое окно «**Формат результата**», указать десятичный формат и задать 8 знаков после десятичной точки.

Таблица 2.1. Встроенные функции для работы с числовыми данными

Математическое написание	Обозначение в Mathcad	Примечания
$\sin x$	$\sin(x)$	аргумент $x$ в радианах
$\cos x$	$\cos(x)$	аргумент $x$ в радианах
$tg\ x$	$\tan(x)$	аргумент $x$ в радианах
$ctg\ x$	$\cot(x)$	аргумент $x$ в радианах
$arcsin\ x$	$asin(x)$	результат в радианах
$arccos\ x$	$acos(x)$	результат в радианах
$arctg\ x$	$atan(x)$	результат в радианах
$e^x$	$exp(x)$	
$\ln x$	$\ln(x)$	$x > 0$
$\lg x$	$\log(x)$	$x > 0$
$ctg\ x$	$\cot(x)$	аргумент $x$ в радианах

Таблица 2.2. Форма записи математических выражений в Mathcad

Математическое написание	Обозначение в Mathcad
$\sin^3 x$	$\sin(x)^3$
$\ln^2 x$	$\ln(x)^2$
$\log_a b$	$\frac{\ln(b)}{\ln(a)}$
$2ab$	$2*a*b$
$\frac{\pi}{2} - arctg(x)$	$\frac{\pi}{2} - atan(x)$

В задаче 6 необходимо вычислить значение тригонометрической функции в случае, когда аргумент задан в градусах. Особенностью этой задачи является то, что в Mathcad аргумент тригонометрической функции должен быть задан в радианах. Для преобразования градусов в радианы используется специальная функция **deg**. Например, чтобы рассчитать значение  $\sin(5\text{градусов})$  в Mathcad, необходимо ввести выражение  $\sin(5\text{deg})$ .

В задаче 7 необходимо выполнить действия над выражением, содержащим комплексные числа. При вводе комплексных чисел необходимо соблюдать их алгебраическую форму:  $Z = \text{Re}(Z) + i \cdot \text{Im}(Z)$ . В этой форме числа содержат действительную  $\text{Re}(Z)$  и мнимую  $\text{Im}(Z)$  части.  $\text{Re}(Z)$  и  $\text{Im}(Z)$  могут иметь целые или действительные значения, но в записи числа  $Z$  мнимая часть содержит множитель  $i$  (или  $j$ ), который является квадратным корнем из  $-1$ . Для выполнения операций над комплексными числами предусмотрены специальные функции и операции, которые приведены в таблице 2.3.

Таблица 2.3. Встроенные функции и операции для работы с комплексными числами

Описание	Обозначение в Mathcad
Вычисление мнимой части комплексного числа.	$\text{Im}(z)$
Вычисление действительной части комплексного числа.	$\text{Re}(z)$
Определение значение угла, между радиусом – вектором и осью X	$\text{arg}(z)$
Вычисление модуля комплексного числа	$ Z $
Определение комплексного числа, сопряженного с комплексным числом $Z$ . Вводится нажатием клавиши « $\bar{\phantom{x}}$ »	$\bar{z}$

### 2.1.3. Варианты заданий

Варианты задания 2.1 представлены в нижеприведенных таблицах.

Таблица 2.4. Индивидуальные варианты задания 2.1 (задача 1)

№	Выражение	№	Выражение
1	$\left(3\frac{5}{8} + \frac{1}{4} + 2\frac{7}{12}\right) - \left(4\frac{8}{15} + \frac{11}{30} + \frac{17}{45}\right)$	2	$\frac{2\frac{3}{5}}{6\frac{1}{15}} + 1\frac{1}{14} - 1\frac{39}{73} \cdot \left(5\frac{5}{7} - 5\frac{1}{16}\right)$
3	$\left(14\frac{2}{3} + 1\frac{1}{2} + 3\frac{2}{5}\right) - \left(4\frac{3}{4} + 1\frac{7}{12} - 2\frac{4}{9}\right)$	4	$\frac{\left(1\frac{1}{2} + 2\frac{2}{3} + 3\frac{3}{4}\right) \cdot 3\frac{3}{5} + \frac{5}{7}}{14 - \frac{15\frac{1}{8}}{2\frac{1}{5}}}$
5	$\left(15\frac{3}{7} - 3\frac{11}{14}\right) + \left(1 - \frac{6}{7}\right) + \left(2\frac{3}{4} - 1\frac{5}{14}\right)$	6	$\frac{\left(4\frac{5}{7} - 1\frac{11}{14}\right) \cdot 4\frac{2}{3} + \left(3\frac{2}{9} - 1\frac{5}{6}\right) \cdot \frac{18}{25}}{2\frac{3}{4}}$
7	$\left(4\frac{5}{18} - 3\frac{1}{3}\right) + \left(2\frac{3}{5} - 1\frac{17}{25}\right) - 1\frac{209}{450}$	8	$\left(\frac{5\frac{5}{9} - \frac{7}{18}}{35} + \frac{40}{63} - \frac{8}{21} + \frac{83}{90} - \frac{41}{50}\right) \cdot 35$
9	$105 \cdot \left[ \left(12\frac{1}{2} + 28\frac{6}{7}\right) - \left(\frac{19}{21} + 34\frac{5}{21}\right) \right] - \left(103\frac{4}{21} - 72\frac{5}{18}\right)$		

№	Выражение	№	Выражение
10	$25\frac{1}{45} - \left(3\frac{4}{5} - 1\frac{14}{15}\right) - 10\frac{7}{9} - \left[15\frac{1}{90} - \left(7\frac{1}{5} + \frac{11}{15}\right)\right]$	11	$\frac{1}{3} \cdot \frac{14\frac{4}{5} - 6\frac{11}{12} + 12\frac{3}{4} - 7\frac{2}{15}}{10\frac{2}{3} - 3\frac{11}{12}} + 2\frac{2}{3} \cdot 3\frac{3}{4}$
12	$2\frac{7}{15} - \left[1\frac{1}{12} - \left[2\frac{7}{90} - \left(1\frac{1}{2} - \frac{8}{45}\right)\right]\right]$	13	$11\frac{2}{5} + 7\frac{1}{2} \cdot \frac{285\frac{3}{5} - 1\frac{23}{30} + \frac{13}{50}}{24\frac{2}{5} - 10\frac{23}{100}}$
14	$\frac{48\frac{3}{5} \cdot \frac{5}{12} - 2\frac{5}{6} + 1\frac{75}{94} \cdot \left(1\frac{1}{2} \cdot \frac{1}{3} - \frac{1}{2}\right)}{\frac{6\frac{3}{4}}{4}}$	15	$\frac{7}{9} \cdot 1\frac{2}{7} + \frac{43\frac{3}{4}}{11\frac{2}{3}} - 3\frac{18}{25} + 1\frac{1}{45} \cdot \left(\frac{37\frac{1}{2}}{2\frac{1}{12}} - 1\frac{3}{23} \cdot 9\right) + \frac{47}{100}$

Таблица 2.5. Индивидуальные варианты задания 2.1 (задача 2)

№	Выражение	№	Выражение
1	$\frac{4.06 \cdot 0.0058 + 3.3044895 - \left(\frac{0.7584}{2.37} + \frac{0.0003}{8}\right)}{0.03625 \cdot 80 - 2.43}$	2	$\frac{(1.238 + 2.762) \cdot 0.1 + (4.36 - 1.16) \cdot 0.3125}{\frac{(36.487 - 34.237)}{2.8125} \cdot 0.2 \cdot \frac{(47.8 - 45.55)}{0.225}}$ 20
3	$\frac{2.045 \cdot 0.033 + 10.518395 - \frac{0.464774}{0.0562}}{\frac{0.003092}{0.0001} - 5.188}$	4	$\frac{\left(\frac{81.624}{4.8} - 4.505\right)^2 + 125 \cdot 0.75}{\left(\frac{0.44^2}{0.88} + 3.53\right)^2 - 2.75^2} \cdot 0.01$ 0.52
5	$\frac{5.7 \cdot 16.2}{20.52} + \frac{127.68 \cdot 0.5}{4.56} + \frac{34.68 \cdot 15.4}{6.8 \cdot 3.57}$	6	$\left[ \frac{\left(\frac{5.2^2}{2.6} + 8.1\right)^2 - 6.5^2}{0.025} \cdot 0.01 \right]$ $\left[ \left(\frac{60.192}{2.4} - 1.08\right)^2 - 0.24 \cdot 1400 \right]$
7	$\frac{\left[ \frac{0.3 \cdot (3.6 - 2.8)}{0.25 \cdot (0.94 + 1.06)} + \frac{(0.2 - 0.15)}{0.001} \right]}{26.92} + \frac{(4.7 - 3.9) \cdot 10}{10}$	8	$20 - \frac{8.8077}{13.333 \cdot 0.3 + 0.0001} + 4.9 \cdot 2.004$
9	$32.52 - \left[ \left(6 + \frac{9.728}{3.2}\right) \cdot 2.5 - 1.6 \right] \cdot 1.2 - \frac{0.015}{0.01}$	10	$\frac{6.62^2 + 5.4 \cdot 3.38 + 1.22 \cdot 3.38}{20.1^2 - 13^2 + 33.1 \cdot 12.9} \cdot 1.5$
11	$50.32 - \frac{\left(20 + \frac{9.744}{2.4}\right) \cdot 0.5 - 1.63}{0.25} + \frac{0.0752}{0.04}$	12	$\frac{0.536^2 - 0.464^2}{3.6^2 - 7.2 \cdot 2.4 + 2.4^2} \cdot 2.5$

№	Выражение	№	Выражение
13	$\frac{57.24 \cdot 3.55 + 430.728}{2.7 \cdot 1.88 - 1.336} + \frac{127.18 \cdot 4.35 + 14.067}{18 + \frac{2.1492}{3.582}}$		$\frac{\quad}{50.34 - 10.34} \cdot 0.3$
14	$\frac{(4.561 + 5.439) \cdot 0.1}{\frac{7.01 - 5.01}{0.5}} - \frac{(4.45 - 2.2)}{0.3} \cdot \frac{0.45}{(0.823 + 0.177) \cdot 30} + \frac{0.45}{30}$		
15	$\frac{\left(\frac{9.126}{0.65} + 0.46\right) \cdot 7.18 + 1.45 \cdot 28.2}{3.45^2 - 0.55^2}$		$\left(\frac{16.8}{0.35} \cdot 1.75 - 1.29\right) \cdot \left(\frac{92.04}{0.16} - \frac{14.496}{0.16}\right)$

Таблица 2.6. Индивидуальные варианты задания 2.1 (задача 3)

№	Выражение	№	Выражение
1	$\frac{(0.875 - 0.7)}{5 \cdot \frac{2}{7} - 3 \cdot \frac{15}{28}} + \left[ \left( \frac{1}{4} - 0.05 \right) \cdot \frac{5}{13} + \frac{1}{\frac{3}{4} + \frac{1}{3}} \right] \cdot \frac{3}{8}$		
2	$\left( 6 \cdot \frac{7}{12} - 3 \cdot \frac{17}{36} \right) \cdot 2.5 - \frac{4 \cdot \frac{1}{3}}{0.65}$	3	$2 \cdot \frac{1}{4} - \frac{\left( \frac{1}{3} + 0.5 + 0.25 \right)}{\frac{2}{3} + \frac{5}{12}} - \frac{8}{(7.5 - 6.2) \cdot \frac{5}{13} + 6.2}$
4	$2.88 \cdot \frac{35}{72} + \left( 1.0625 - \frac{5}{12} \right) \cdot 16$	5	$\frac{\left( 11 \cdot \frac{11}{28} + 3 \cdot \frac{13}{36} - 12 \cdot \frac{61}{63} \right)}{\frac{15}{28}} + \frac{23.517}{0.3}$
6	$\left( 2 \cdot \frac{7}{12} + 2 \cdot \frac{19}{42} \right) \cdot 3 - \frac{64.5}{6} + 4 \cdot \frac{2}{7} \cdot 2.1 + 1.3 \cdot 4 \cdot \frac{1}{6}$	7	$\frac{\left( 20 \cdot \frac{4}{9} + 12 \cdot 25 - 31 \cdot \frac{1}{30} \right)}{299} + \frac{\left( 17 \cdot \frac{1}{9} - 2 \cdot 45 \cdot 5 + 5 \cdot \frac{1}{30} \right)}{13}$
8	$\frac{7}{0.2625} - \frac{3.6}{\frac{68.1}{7.5} - 7 \cdot \frac{17}{20} + 1 \cdot \frac{1}{50}} + 4 \cdot \frac{5}{6} \cdot \frac{33}{58}$	9	$\frac{\left( 19 \cdot \frac{1}{6} + 43.75 \right) \cdot \frac{6}{5} - \left( 26.8 - 23 \cdot \frac{3}{7} \right) \cdot \frac{35}{6}}{\left( 13.3 - 11 \cdot \frac{1}{2} \right) \cdot \frac{10}{18}} - \frac{0.5}{0.5}$
10	$\frac{24.57}{3.5} + \left( 3.35 - 2 \cdot \frac{13}{15} + \frac{5}{8} \right) \cdot \left( \frac{225}{12.5} - 3 \cdot \frac{14}{19} \cdot 2 \right)$	11	$\frac{\left( 4 \cdot \frac{1}{7} - 0.005 \cdot 700 \right)}{0.125} + \frac{1 \cdot \frac{89}{90}}{5 \cdot \frac{1}{72} - 3 \cdot \frac{1}{40}}$
12	$\left[ 8.6 \cdot \frac{1}{4} - \left( 5 \cdot \frac{61}{90} - 4 \cdot \frac{1}{12} \right) \right] \cdot \left( \frac{7}{\frac{40}{2} + 1.34} + \frac{11}{12} \right)$	13	$12 \cdot \frac{1}{2} + \left( 17 \cdot \frac{1}{2} - 8.25 \cdot \frac{10}{11} \right) \cdot \left( \frac{11 \cdot \frac{2}{3} + 3.5}{\frac{2}{9}} \right) - \frac{12.6}{2.5}$
14	$\frac{\left( 4.625 - \frac{13}{18} \cdot \frac{9}{26} \right)}{2.25} + \frac{2 \cdot \frac{1}{2}}{\frac{125}{6} \cdot \frac{3}{4}}$	15	$\left( 3 \cdot \frac{7}{18} - 2 \cdot \frac{25}{36} + \frac{7}{48} \right) \cdot 6 \cdot \frac{6}{11} + 1.5 \cdot \frac{20.15}{2 \cdot \frac{1}{2}} - 10.09$

Таблица 2.7. Индивидуальные варианты задания 2.1 (задача 4)

№	Выражение	№	Выражение
1	$x = -4.87 \quad y = 3 \cdot \frac{\pi}{2}$ $\sqrt{x^2 + y^2} \cdot \sin(\arctg(x)) + \frac{x}{y}$	2	$a = 2.3 \quad x = 0.3 \quad b = 0.75$ $\arcsin\left(\frac{x}{a}\right) - e^{-bx} \cdot \sqrt{1+x}$
3	$c = 0.75 \quad b = 2 \quad x = 1.5$ $e^{-\sqrt{x}} \cdot \cos^2(b \cdot x) + c \cdot \tg(x)$	4	$a = 1.2 \quad x = 3.5 \quad b = 0.7$ $\frac{2^x \cdot \ln(a \cdot x) - 3^x \cdot \ln(b \cdot x)}{\ctg^2(a \cdot x)}$
5	$a = 2 \quad b = 0.7 \quad c = 0.5 \quad x = 0.2$ $\frac{a \cdot \cos(b \cdot x \cdot \sin(x)) + c}{\sin(a) + \frac{1}{\tg(b)}}$	6	$a = 1.5 \quad b = 2 \quad t = 0.8$ $a \cdot \arctg\left(\frac{t}{a}\right) + b \cdot \arcsin\left(\frac{t}{b}\right)$
7	$a = 2 \quad b = 1.2 \quad x = 0.9$ $\sqrt[3]{a + b \cdot e^{\sin(x)} + 1}$	8	$a = 0.5 \quad b = -1.2 \quad x = 1.5$ $e^{-ax} \cdot \sqrt[3]{a \cdot x + b \cdot \sin^2(2 \cdot x) \cdot \tg(x)}$
9	$m = 0.7 \quad c = 2.1 \quad x = 1.5$ $\frac{\sin(x) \cdot \frac{1}{\tg(m)}}{\sqrt{1 + m^2 \cdot \sin^2(x)}} - c \cdot \ln(m \cdot x)$	10	$x := 0.5 \quad a := 0.3 \quad b := 0.9$ $\frac{a^{2x} + x \cdot b^{-x} \cdot \cos(a + b)}{x + 1} \cdot \ctg^2(a \cdot x)$
11	$x = 0.1 \quad a = -0.5 \quad b = 2.3$ $\frac{b \cdot x^2 - a}{e^{ax} - 1} \cdot \arccos(x)$	12	$x = 0.5 \quad a = 0.5 \quad b = 2.9$ $\frac{a \cdot x + e^{-x} \cdot \cos^2(b \cdot x)}{b \cdot x - e^{-x} \cdot \ctg(a \cdot x) + 1}$
13	$a = -0.5 \quad b = 2.3 \quad x = -0.5$ $\frac{b \cdot x \cdot e^{ax} + a \cdot \sqrt{x + 1.5}}{\arctg(x)}$	14	$a = 2.1 \quad b = -0.3 \quad x = 3.1$ $b \cdot \sin\left(a \cdot x^2 \cdot \cos(2 \cdot x)\right) - \arcsin\left(\frac{a}{x}\right)$
15	$a = 4.1 \quad b = -2.3 \quad x = 0.5$ $\frac{x + a \cdot \cos(2 \cdot x)}{x + \sqrt{a + b \cdot \sin(3 \cdot x)}} \cdot \frac{1}{\arctg(x)}$		

Таблица 2.8. Индивидуальные варианты задания 2.1 (задача 5)

№	Выражение	№	Выражение
1	$z1 = \frac{x^2 + 2 \cdot x - 3 + (x+1) \cdot \sqrt{x^2 - 9}}{x^2 - 2 \cdot x - 3 + (x-1) \cdot \sqrt{x^2 - 9}}$ $z2 = \sqrt{\frac{x+3}{x-3}}$	2	$z1 = \frac{\cos(x) + \sin(x)}{\cos(x) - \sin(x)}$ $z2 = \operatorname{tg}(2 \cdot x) + \frac{1}{\cos(2 \cdot x)}$
3	$y1 = \cos(x) + \sin(x) + \cos(3 \cdot x) + \sin(3 \cdot x)$ $y2 = 2 \cdot \sqrt{2} \cdot \cos(x) \cdot \sin\left(2 \cdot x + \frac{\pi}{4}\right)$		
4	$y1 = \cos(x) + \cos(2 \cdot x) + \cos(6 \cdot x) + \cos(7 \cdot x)$ $y2 = 4 \cdot \cos\left(\frac{x}{2}\right) \cdot \cos\left(5 \cdot \frac{x}{2}\right) \cdot \cos(4 \cdot x)$		
5	$y1 = 2 \cdot \sin^2(3 \cdot \pi - 2 \cdot x) \cdot \cos^2(5 \cdot \pi + 2 \cdot x)$ $y2 = \frac{1}{4} - \frac{1}{4} \cdot \sin\left(\frac{5}{2} \cdot \pi - 8 \cdot x\right)$		
6	$z1 = \left(\frac{1+x+x^2}{2 \cdot x+x^2} + 2 - \frac{1-x+x^2}{2 \cdot x-x^2}\right)^{-1} \cdot (5-2 \cdot x^2)$ $z2 = \frac{4-x^2}{2}$	7	$z1 = \left(\frac{x+2}{\sqrt{2 \cdot x}} - \frac{x}{\sqrt{2 \cdot x+2}} + \frac{2}{x-\sqrt{2 \cdot x}}\right) \cdot \frac{\sqrt{x}-\sqrt{2}}{x+2}$ $z2 = \frac{1}{\sqrt{x+\sqrt{2}}}$
8	$z1 = \cos^4(x) + \sin^2(y) + \frac{1}{4} \cdot \sin^2(2 \cdot x) - 1$ $z2 = \sin(x+y) \cdot \sin(y-x)$	9	$z1 = \frac{\sqrt{(3 \cdot x+2)^2 - 24 \cdot x}}{3 \cdot \sqrt{x} - \frac{2}{\sqrt{x}}}$ $z2 = \sqrt{x}$
10	$z1 = (\cos(x) - \cos(y))^2 - (\sin(x) - \sin(y))^2$ $z2 = -4 \cdot \sin^2 \frac{x-y}{2} \cos(x+y)$	11	$z1 = \frac{(m-1) \cdot \sqrt{m} - (n-1) \cdot \sqrt{n}}{\sqrt{m^3 \cdot n + m \cdot n + m^2 - m}}$ $z2 = \frac{\sqrt{m} - \sqrt{n}}{m}$
12	$z1 = \frac{\sin\left(\frac{\pi}{2} + 3 \cdot x\right)}{1 - \sin(3 \cdot x - \pi)}$ $z2 = \frac{1}{\operatorname{tg}\left(\frac{5}{4} \cdot \pi + \frac{3}{2} \cdot x\right)}$	13	$z1 = \frac{\sqrt{2 \cdot x+2} \cdot \sqrt{x^2-4}}{\sqrt{x^2-4+x+2}}$ $z2 = \frac{1}{\sqrt{x+2}}$
14	$z1 = \frac{\sin(4 \cdot x)}{1 + \cos(4 \cdot x)} \cdot \frac{\cos(2 \cdot x)}{1 + \cos(2 \cdot x)}$ $z2 = \frac{1}{\operatorname{tg}\left(\frac{3}{2} \cdot \pi - x\right)}$	15	$y1 = 1 - \frac{1}{4} \sin^2(2 \cdot x) + \cos(2 \cdot x)$ $y2 = \cos^2(x) + \cos^4(x)$

Таблица 2.9. Индивидуальные варианты задания 2.1 (задача 6)

№	Выражение	№	Выражение
1	$a = 4.82 \quad x = 10 \text{ градусов}$ $\frac{\ln(\cos(x))}{a \cdot \sin^2(x)} + \sqrt{a \cdot \operatorname{tg}(x)}$	2	$a = 2 \quad x = 4 \text{ градуса}$ $a \cdot \frac{\sin(a \cdot x)}{a - \cos(a - x)}$
3	$a = 1.25 \quad x = 10 \text{ градусов}$ $\sqrt[3]{\sin(3 \cdot x) \cdot \cos(3 \cdot x) - x \cdot \operatorname{arctg}(x)}$	4	$a = 3 \quad x = 2.5 \text{ градусов}$ $a \cdot \operatorname{tg}(\cos(a \cdot x) - \sin(a + x))$
5	$a = 5.32 \quad x = 20 \text{ градусов}$ $\frac{\cos(x)}{\sqrt[5]{\sin^2(x \cdot a)}} + a^2 \cdot \sqrt{1 + \operatorname{tg}(x)}$	6	$a = 1 \quad x = 14 \text{ градусов}$ $\frac{a \cdot \sin(x) + (a + 1) \cdot \sin(2 \cdot x)}{\cos(a + x)}$
7	$a = 2.51 \quad x = 36 \text{ градусов}$ $\frac{a \cdot \sin(x)}{\sqrt[3]{2 + 3 \cdot \cos(x)}} + e^{a \cdot x}$	8	$a = 2 \quad x = 1.5 \text{ градусов}$ $\frac{a \cdot \cos(x) + a^2 \cdot \cos^2(x)}{a^3 \cdot \cos^3(x)}$
9	$a = 0.123 \quad x = 16 \text{ градусов}$ $\frac{\sqrt{a} \cdot \cos(5 \cdot x)}{3 + 2 \cdot \sin(5 \cdot x)} - \operatorname{arcsin}(a)$	10	$a = 2 \quad x = 3 \text{ градусов}$ $\frac{\cos(2 \cdot a \cdot x) - \cos(x)}{\cos\left(x + \frac{\pi}{a + 3}\right) + \sin\left(x - \frac{\pi}{a - 3}\right)}$
11	$a = 0.4 \cdot \pi \quad x = 21 \text{ градус}$ $12 \cdot \cos^2\left(\frac{a}{3} - \frac{x}{4}\right) + \sin\left(\frac{a}{x}\right)$	12	$a = 2 \quad b = 3 \quad x = 31 \text{ градус}$ $\frac{a \cdot \sqrt{b} \cdot \operatorname{tg}(x) \cdot \sin(x)}{b \cdot \sqrt{a}(\operatorname{tg}(x) + \sin(x))}$
13	$a = \frac{\pi}{21} \quad x = 40 \text{ градусов}$ $\frac{a \cdot \sin\left(\sqrt{x + \frac{\pi}{18}}\right)}{\cos\left(\sqrt{x + \frac{\pi}{18}}\right)}$	14	$a = 2.1 \quad x = 3.2 \text{ градуса}$ $\frac{a}{\sin(a \cdot x)} + \frac{\cos(a \cdot x)}{a + 2}$
15	$a = 2.15 \quad x = 46 \text{ градусов}$ $\frac{\sin(a - x^2) - \sqrt{a^2 + x^2}}{a^2 \cdot \sqrt{x}}$		

Таблица 2.10. Индивидуальные варианты задания 2.1 (задача 7)

№	Выражение	№	Выражение
1	$\frac{1+i}{2i+1}$	2	$(3+4i) \cdot (4-3i) + (3-4i) \cdot (2+3i)$
3	$\frac{2-17i}{3} - \frac{(2+i)}{3+i}$	4	$(1-i)^4 + (2+i)^4$
5	$\frac{2-i}{i+2} + (i-1)^2$	6	$(1-i) \cdot (4+3i) \cdot (2+i) \cdot (3+i)$
7	$\frac{3+i}{3-i} + \frac{2-i}{2+i}$	8	$\left(\frac{1-\sqrt{3}i}{i-1.4}\right)^{20}$
9	$(2-i)^2 + (1+i)^4 - \frac{7-i}{2+i}$	10	$\frac{1}{2i} \cdot \left(i^7 - \frac{1}{i^7-1}\right)$
11	$\frac{2+i}{3i-1} + (2i-1)^2$	12	$(1-i)^2 \cdot (\sqrt{3}+i)^5$
13	$\frac{(2i+3)^2}{i-1} - \frac{i}{1+i}$	14	$\left(\frac{1+3i}{2}\right)^{-2}$
15	$\frac{(1+i) \cdot (2+i)}{2-i} - \frac{(1-i) \cdot (1+i)}{2+i}$		

#### 2.1.4. Примеры выполнения заданий

Примеры представим в виде листингов Mathcad.

Задача 1. Вычислить выражение и представить результат в виде смешанной дроби.

Листинг 2.1. Результаты решения задач 1 (задание 2.1)

$$\frac{\left(9\frac{1}{4} - 7\frac{2}{5}\right) \cdot 2\frac{1}{2} - 1\frac{1}{2}}{3\frac{1}{12}} + \frac{6 - 4 \cdot \frac{1}{10}}{7 + \frac{1}{\frac{3}{7}}} = 13\frac{1}{10}$$

Задача 2. Вычислить выражение и представить результат в виде десятичной дроби.

Листинг 2.2. Результаты решения задач 2 (задание 2.1)

$$\frac{3.05^2 - 2.55^2}{0.35 \cdot 388 - 28.8 \cdot \left(20.56 - \frac{14.501}{0.85}\right)} = 0.08$$

Задача 3. Вычислить выражение и выбрать формат для более точного представления результата

Листинг 2.3. Результат решения задачи 3 (задание 2.1)

Способ 1	Результат в виде десятичной дроби
$\frac{20 \frac{8}{15} \cdot 7.5 - \frac{54.6}{0.4}}{3 \frac{13}{21} \cdot 8.4 - \frac{34.4}{14 \frac{1}{3}}} + \frac{43.75}{11 \frac{2}{3}} + \frac{24.6}{1 \frac{3}{4}} = 18.432$	
Способ 2	Результат в виде смешанной дроби
$\frac{20 \frac{8}{15} \cdot 7.5 - \frac{54.6}{0.4}}{3 \frac{13}{21} \cdot 8.4 - \frac{34.4}{14 \frac{1}{3}}} + \frac{43.75}{11 \frac{2}{3}} + \frac{24.6}{1 \frac{3}{4}} = 18 \frac{121}{280}$	
<p>Ответ. Результат более точно представлен во втором способе.</p>	

Задача 4. Вычислить значение функции при заданных значениях аргумента. Результаты представить с точностью до 8 знаков после десятичной точки.

$$a = 0.5 \quad b = 3.1 \quad x = 1.3$$

$$\frac{\sqrt{a \cdot x \cdot \sin^2(2 \cdot x) + e^{-2x} \cdot (x + b)}}{\arccos\left(\frac{x}{b}\right)}$$

Листинг 2.4. Результат решения задачи 4 (задание 2.1)

$a := 0.5 \quad b := 3.1 \quad x := 1.3$ $\frac{\sqrt{a \cdot x \cdot \sin^2(2 \cdot x) + e^{-2x} \cdot (x + b)}}{\operatorname{acos}\left(\frac{x}{b}\right)} = 0.62103700$
--

Задача 5. Вычислить значение двух заданных функций. Задать для обеих функций одинаковые значения аргумента. Значения функций при этом должны быть равны

Листинг 2.5. Результат решения задачи 5 (задание 2.1)

$$z1(x) := \cos\left(\frac{3}{8}\pi - \frac{x}{4}\right)^2 - \cos\left(\frac{11}{8}\pi + \frac{x}{4}\right)^2 \quad z2(x) := \frac{\sqrt{2}}{2} \cdot \sin\left(\frac{x}{2}\right)$$

$$z1(3) = 0.705 \quad z2(3) = 0.705$$

Задача 6. Вычислить значение тригонометрической функции при задании значения аргумента в градусах

$$a = 4 \quad x = 56 \text{ градусов}$$

$$a^2 \frac{\operatorname{tg}(x) \cdot \sin^2(x)}{\ln(\cos(x + a))}$$

Листинг 2.6. Результат решения задачи 6 (задание 2.1)

$$a := 4 \quad f(x) := a^2 \cdot \frac{\tan(x) \cdot \sin(x)^2}{\ln(\cos(x + a))}$$

$$x := 56 \cdot \text{deg} \quad f(x) = -12.169$$

Задача 7. Упростить выражение, выделить действительную и мнимую часть результата, определить значение угла между радиусом-вектором комплексного числа и осью X. Определить комплексное число, сопряженное результату и модуль результата.

$$\frac{(1 - i)^5 - 1}{(1 + i)^5 + 1}$$

Листинг 2.7. Результат решения задачи 7 (задание 2.1)

$$\frac{(1 - i)^5 - 1}{(1 + i)^5 + 1} = -0.04 - 1.28i \quad z := -0.04 - 1.28i$$

$$\operatorname{Re}(z) = -0.04 \quad \operatorname{Im}(z) = -1.28$$

$$\operatorname{arg}(z) = -1.602 \quad \bar{z} = -0.04 + 1.28i$$

$$|z| = 1.281$$

## Контрольные вопросы

1. Какая функция используется для перевода значения аргумента тригонометрической функции в радианы, если оно задано в градусах?
2. Как указать формат отображения результатов расчетов?
3. В каких случаях используется оператор присваивания и знак равенства?
4. Что такое встроенные функции и функции пользователя?
5. Какие встроенные функции используются для работы с комплексными числами?
6. В какой последовательности выполняются вычисления в Mathcad?
7. Как задается функция пользователя?
8. Что используется для облегчения ввода встроенных математических функций?
9. Как определить комплексное число, сопряженное с комплексным числом  $Z$ ?

## 2.2. Символьные вычисления

**Цель работы:** познакомиться с особенностями выполнения символьных вычислений в Mathcad. Овладеть навыками символьного решения уравнений и систем уравнений. Освоить операции символьного дифференцирования и интегрирования, вычисления сумм, произведений ряда, пределов, разложения в ряд Тейлора.

**Время:** 2 часа.

**Литература:** Л1 с. 114-117.

### 2.2.1. Задание 2.2

1. Упростить символьное выражение.
2. Решить уравнение в символьном виде.
3. Решить систему уравнений в символьном виде.
4. Выполнить символьное интегрирование.
5. Выполнить символьное дифференцирование, найти первую и вторую производную заданной функции.
6. Найти сумму и произведение ряда.
7. Разложить функцию в ряд Тейлора и проверить полученный результат путем вычисления по прилагаемой формуле суммы ряда. Число членов ряда ( $k$ ) выбрать в диапазоне от 5 до 12. Значение аргумента  $x$  задается самостоятельно.
8. Вычислить предел функции.

### 2.2.2. Указания к выполнению задания

При выполнении символьных вычислений в Mathcad используется три способа: при помощи меню «Символика»; при помощи специального оператора символьного ввода, включающего знак символьного равенства  $[ \rightarrow ]$ ; с использованием стандартных функций, в сочетании с символьным знаком равенства.

При выполнении задачи 1 для упрощения символьного выражения можно использовать оператор *simplify* (упростить). Активизировать данный оператор можно на панели инструментов «Символика» или воспользоваться меню «Символика» команда «Упростить». При использовании оператора *simplify* результат отображается более наглядно, так как связывает посредством знака символьного равенства исходное выражение и итоговый результат.

Задачи 2 и 3 решаются аналогично. Для получения результатов может быть использован оператор «**■ solve, ■ →**». Слева от **solve** вместо **■** вставляется уравнение или система уравнений, а справа - имена переменных, относительно которых решается уравнение или система уравнений. Символьное решение может быть получено с использованием блока **given ... find**. В этом случае при записи уравнения для связи его левой и правой части используется символ равенства в виде булевого «**=**». Например,

$$a \cdot x^2 + b \cdot y = 0 \quad d \cdot y + p \cdot x = 0$$

$$\text{Find}(x, y) \rightarrow \begin{pmatrix} 0 & \frac{1}{d} \cdot b \cdot \frac{p}{a} \\ 0 & -b \cdot \frac{p^2}{a \cdot d^2} \end{pmatrix}$$

Задачи 4 и 5 могут быть реализованы посредством панели инструментов «**Исчисления**», используя знаки интегрирования  $\int$  и дифференцирования  $\frac{d}{dx}$ . Второй способ - с использованием меню «**Символика**» команда «**Переменная**» и далее выбрать «**Дифференцировать**» или «**Интегрировать**» в зависимости от выполняемой операции. При этом предварительно необходимо выделить переменную, относительно которой производится интегрирование или дифференцирование. Первый способ более показателен, так как отображает выполняемое математическое действие, и наиболее приемлем в случае дальнейшего оформления результатов в виде отчета. При вычислении неопределенного интеграла необходимо учитывать, что Mathcad не выводит постоянную интегрирования «**C**», поэтому если она необходима в дальнейших расчетах, то вводится принудительно.

В задаче 6 для задания условия задачи на панели инструментов «**Исчисления**» выбираются знаки  $\sum$  или  $\prod$  в зависимости от того, что необходимо вычислить, сумму или произведение членов ряда. Вместо маркеров, аналогично символике, принятой в классической математике, вводятся соответствующие данные. Результат отображается после ввода знака символьного равенства «**→**».

В задаче 7 для разложения функции  $f(x)$  в ряд Тейлора используется оператор **series** панели инструментов «**Символика**», который имеет следующий вид **■ series, ■, ■**. Самое левое поле предназначено для ввода функции. Среднее поле должно содержать имя переменной, по которой производится разложение функции в ряд Тейлора, в крайнее правое поле вводится число, определяющее порядок многочлена Тейлора. Дополнительно в задаче задается эталонная формула суммы членов ряда для данной функции. В задаче необходимо сравнить результаты, полученные путем разложения функции в ряд Тейлора, и на основе формулы суммы членов ряда. Значение аргументов функции и констант, используемых в формулах, могут быть взяты произвольными.

В задаче 8 при вычислении предела последовательности и функции, в зависимости от условия задачи, определяющего вычисление предела функции слева и справа, приняты обозначения

$$\lim_{x \rightarrow a} \quad \lim_{x \rightarrow a^+} \quad \lim_{x \rightarrow a^-}$$

Для ввода данных шаблонов используется панель инструментов «Символика».

Правила записи информации при вычислении пределов, аналогичны правилам, принятым в классической математике. Условия задачи и результаты связываются посредством символического равенства.

### 2.2.3. Варианты заданий

Варианты задания 2.2 представлены в нижеприведенных таблицах.

Таблица 2.11. Индивидуальные варианты задания 2.2 (задача 1)

№	Выражение	№	Выражение
1	$\frac{\left(m^2 - \frac{1}{n^2}\right)^m \cdot \left(n + \frac{1}{m}\right)^{n-m}}{\left(n^2 - \frac{1}{m^2}\right)^n \cdot \left(m - \frac{1}{n}\right)^{m-n}}$	2	$\frac{\frac{(a-b)^2 + ab}{(a+b)^2 - ab}}{\frac{a^5 + b^5 + a^2 \cdot b^3 + a^3 \cdot b^2}{(a^3 + b^3 + a^2 \cdot b + a \cdot b^2) \cdot (a^3 - b^3)}}$
3	$\frac{x^{-6} - 64}{4 + 2 \cdot x^{-1} + x^{-2}} \cdot \frac{x^2}{4 - \frac{4}{x} + \frac{1}{x^2}} - \frac{4 \cdot x^2 \cdot 2 \cdot x + 1}{1 - 2 \cdot x}$		
4	$\left(\frac{1}{t^2 + 3 \cdot t + 2} + \frac{2 \cdot t}{t^2 + 4 \cdot t + 3} + \frac{1}{t^2 + 5 \cdot t + 6}\right)^2 \cdot \frac{(t-3)^2 + 12t}{2}$		
5	$\frac{\frac{a-b}{2 \cdot a - b} - \frac{a^2 + b^2 + a}{2 \cdot a^2 + a \cdot b - b^2}}{\frac{(4b^4 + 4ab^2 + a^2)}{2b^2 + a}} \cdot (b^2 + b + a \cdot b + a)$		
6	$\frac{\frac{a^{-1} \cdot b^{-1}}{a^{-3} + b^{-3}}}{\frac{a^2 \cdot b^2}{(a+b)^2 - 3 \cdot a \cdot b} \cdot \left(\frac{a^2 - b^2}{ab}\right)^{-1}}$	7	$\frac{\left(2 - x + 4 \cdot x^2 + \frac{5 \cdot x^2 - 6 \cdot x + 3}{x-1}\right)}{2 \cdot x + 1 + \frac{2 \cdot x}{x-1}}$
8	$\frac{2 \cdot b + a - \frac{4 \cdot a^2 - b^2}{a}}{b^3 + 2 \cdot a \cdot b^2 - 3 \cdot a^2 \cdot b} \cdot \frac{a^3 \cdot b - 2 \cdot a^2 \cdot b^2 + a \cdot b^3}{a^2 - b^2}$	9	$\frac{\left(\frac{2-b}{b-1} + 2 \cdot \frac{a-1}{a-2}\right)}{b \cdot \frac{a-1}{b-1} + a \cdot \frac{2-b}{a-2}}$

Продолжение таблицы 2.11

10	$\frac{(6 \cdot a^2 + 5 \cdot a - 1 + \frac{a+4}{a+1})}{3 \cdot a - 2 + \frac{3}{a+1}}$	11	$\left(\frac{a}{b} + \frac{b}{a} + 2\right) \cdot \frac{\left(\frac{a+b}{2 \cdot a} - \frac{b}{a+b}\right)}{\left[\left(a+2 \cdot b + \frac{b^2}{a}\right) \cdot \left(\frac{a}{a+b} + \frac{b}{a-b}\right)\right]}$
12	$\frac{\frac{1-x^{-2}}{x^2-x} - \frac{2}{x^2} + \frac{x^{-2}-x}{x^2-x}}{\frac{1}{x^2-x} - \frac{-1}{x^2} + \frac{3}{x^2} + \frac{1}{x^2-x} - \frac{-1}{x^2}}$	13	$\frac{1+(a+x)^{-1}}{1-(a+x)^{-1}} \cdot \left[1 - \frac{1-(a^2+x^2)}{2 \cdot a \cdot x}\right]$
14	$\frac{3 \cdot a^2 + 2 \cdot a \cdot x - x^2}{(3 \cdot x + a) \cdot (a + x)} - 2 + 10 \cdot \frac{a \cdot x - 3 \cdot x^2}{a^2 - 9 \cdot x^2}$	15	$\frac{2 \cdot (x^4 + 4 \cdot x^2 - 12) + x^4 + 11 \cdot x^2 + 30}{x^2 + 6}$

Таблица 2.12. Индивидуальные варианты задания 2.2 (задача 2)

№	Выражение	№	Выражение
1	$x \cdot \sqrt{\frac{x+1}{x}} - a = 0$	2	$e^{x+b} - 3 \cdot a + e^{x-a} = 0$
3	$(x+b) \cdot \ln(x+a) = 0$	4	$\sin(a+x) + \sin(x) = \cos\left(\frac{a}{2}\right)$
5	$x + e^{-xb} - a \cdot x = 0$	6	$\cos(x)^2 + \frac{a}{2} \cdot \sin(x) = a - 3$
7	$\frac{1+x^2}{1-x} = \frac{a}{b}$	8	$\cos(x)^2 + \frac{a}{2} \cdot \sin(x) = a - 3$
9	$\sqrt{x^2+3 \cdot a^2} - \sqrt{x^2-3 \cdot a^2} = x \cdot \sqrt{2}$	10	$\ln(a \cdot x) = 2 \ln(x+a)$
11	$2 \cdot \sqrt{a+x} + \sqrt{a-x} = \sqrt{a-x} + \sqrt{x \cdot (a+x)}$		
12	$(1+a) \cdot \left(\frac{x^2}{x^2+1}\right)^2 - 3 \cdot a \cdot \frac{x^2}{x^2+1} + 4 \cdot a = 0$	13	$\sqrt{x-a} + \sqrt{x-b} = \sqrt{a+b}$
14	$\lg(x) + \ln(x) + b = 0$	15	$\cos\left(\frac{x-a}{2}\right)^2 - \sin(x-a) = 2$

Таблица 2.13. Индивидуальные варианты задания 2.2 (задача 3)

№	Выражение	№	Выражение
1	$x + y = a$ $x^2 - y^2 = a^2$	2	$x^{x-y} = a$ $a \cdot (x - y) = \ln(b)$
3	$x + y = a$ $x^2 + y^2 = a^2$	4	$x^{x-y} = a$ $\ln(x - y) = \ln(b)$
5	$x \cdot y - x + y = a$ $x - y = b$	6	$\frac{x^2}{y} + \frac{y^2}{x} = a^2$ $x + y = a$
7	$y^2 + z^2 = a \cdot x^2$ $y - x = z$ $z - x = y - b$	8	$\frac{1}{x} + \frac{1}{y} = a$ $x + y = b$
9	$x^3 + x \cdot y^2 = a$ $y^3 + x^2 \cdot y = b$	10	$(x - y) \cdot (x^2 - y^2) = 3 \cdot a^3$ $(x + y) \cdot (x^2 + y^2) = 15 \cdot a^3$
11	$x + y + z = 0$ $c \cdot x + a \cdot y + b \cdot z = 0$ $(x + b)^2 + (y + c)^2 + (z + a)^2 = a^2 + b^2 + c^2$		
12	$\ln(x) + y = a$ $x + y = b$	13	$x \cdot y = a$ $y \cdot z = b$ $z \cdot x = c$
14	$x^2 = a \cdot x + y$ $y = x + a \cdot y$	15	$x^2 + y^2 \cdot x^2 = a^2$ $x + x \cdot y = b$

Таблица 2.14. Индивидуальные варианты задания 2.2 (задача 4)

№	Выражение	№	Выражение
1	$\int \frac{1}{\sin(x)^2 \cdot \cos(x)} dx$	2	$\int x^2 + \sin(2x) dx$

№	Выражение	№	Выражение
3	$\int \frac{\cos(x) - \sin(x)}{(1 + \sin(x))^2} dx$	4	$\int x^3 \cdot e^{-x} dx$
5	$\int \frac{1}{\sin(x) \cdot (1 - \sin(x))} dx$	6	$\int \frac{x}{(x-2) \cdot (x-3)} dx$
7	$\int \frac{\cos(x)}{2 + 3 \cos(x)} dx$	8	$\int (2 \cdot x^2 + 1) \cdot \cos(3 \cdot x) dx$
9	$\int \frac{\cos(x)}{1 + \sin(x) - \cos(x)} dx$	10	$\int \frac{x}{\sqrt{x^2 - 5}} dx$
11	$\int \frac{\cos(x)}{5 + \sin(x)} dx$	12	$\int \frac{1 - \sqrt{x}}{x - 2 \cdot \sqrt{x}} dx$
13	$\int \frac{\cos(x)}{(1 - \cos(x))^2} dx$	14	$\int 2 \cdot x + \sqrt{2 \cdot x - 3} dx$
15	$\int \frac{1}{\cos(x) \cdot (1 - \cos(x))} dx$		

Таблица 2.15. Индивидуальные варианты задания 2.2 (задача 5)

№	Выражение	№	Выражение
1	$\operatorname{tg}(\sin(x))$	2	$\sin(x)^x$
3	$x^3 \cdot (x^2 - 1)$	4	$\operatorname{arctg}(\sin(2 \cdot x))$
5	$x^3 \cdot \sqrt{x^2 + 1}$	6	$\arcsin(\cos(x))$
7	$\sqrt{x^2 + x} \cdot \sqrt{x}$	8	$\sin(x \cdot \sin(x))$
9	$e^{x^3 - 3x + 1}$	10	$\operatorname{arctg}\left(\sin\left(\frac{1}{x}\right)\right)$
11	$\ln\left(\frac{1}{x+1}\right)$	12	$2 \cdot x^2 + x^2 \cdot \cos\left(\frac{1}{x}\right)$
13	$e^{\operatorname{tg}(x)}$	14	$x \cdot \arcsin(x^2)$
15	$\frac{\cos(x)}{\sin(2 \cdot x)}$		

Таблица 2.16. Индивидуальные варианты задания 2.2 (задача 6)

№	Выражение	№	Выражение
1	$\sum_{n=1}^{\infty} \left( \frac{1}{2^n} + \frac{1}{3^n} \right)$	2	$\sum_{n=0}^{\infty} \frac{x^{n+2}}{n! \cdot (n+2)}$
3	$\sum_{n=1}^{\infty} \frac{\cos\left(\frac{2 \cdot \pi \cdot n}{3}\right)}{2^n}$	4	$\sum_{k=1}^{\infty} \frac{1}{k^2}$
ВЫЧИСЛИТЬ			
5	$\sum_{n=0}^{\infty} \frac{1}{n!} \cdot \sum_{n=0}^{\infty} \frac{(-1)^n}{n!}$	6	$\sum_{k=1}^{\infty} \frac{1}{k^8}$
7	$\sum_{n=0}^{\infty} \frac{2^n \cdot (n+1)}{n!}$	8	$\sum_{n=1}^{\infty} \frac{x^n}{n}$
9	$\sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n}$	10	$\sum_{n=0}^{\infty} \frac{(2 \cdot n + 1) \cdot x^{2n}}{n!}$
11	$\sum_{n=1}^{\infty} \frac{n^2}{n!}$	12	$\sum_{n=1}^{\infty} n \cdot (n+2) \cdot x^n$
13	$\prod_{n=2}^{\infty} \frac{n^3 - 1}{n^3 + 1}$	14	$\left( \sum_{n=0}^{\infty} \frac{x^n}{2^n \cdot n!} \right)$
15	$\sum_{n=1}^{\infty} \frac{1}{n^2 \cdot (n+1)^2 \cdot (n+2)^2}$		

Таблица 2.17. Индивидуальные варианты задания 2.2 (задача 7)

№	Выражение	№	Выражение
1	$e^{-x}$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^n}{n!}$	2	$\arcsin(x)$ $x + \sum_{n=1}^k \frac{\left[ \prod_{p=1}^n (2 \cdot p - 1) \right] \cdot x^{2n+1}}{\left[ \prod_{d=1}^n (2 \cdot d) \right] \cdot (2 \cdot n + 1)}$

Продолжение таблицы 2.17

3	$\ln(x+1)$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^{n+1}}{n+1}$	4	$\arccos(x)$ $\frac{\pi}{2} - \left[ x + \frac{\sum_{n=1}^k \left[ \prod_{d=1}^n (2 \cdot d - 1) \right] \cdot x^{2n+1}}{\left[ \prod_{p=1}^n (2 \cdot p) \right] \cdot (2 \cdot n + 1)} \right]$
5	$\ln\left(\frac{1+x}{1-x}\right)$ $2 \cdot \sum_{n=0}^k \frac{x^{2n+1}}{2 \cdot n + 1}$	6	$a^x$ $1 + \sum_{n=1}^k \frac{(\ln(a) \cdot x)^n}{n!}$
7	$\ln(1-x)$ $-\sum_{n=1}^k \frac{x^n}{n}$	8	$\cos(2 \cdot x)$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^{2n} \cdot 2^{2n}}{(2 \cdot n)!}$
9	$\operatorname{arctg}(x)$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^{2n+1}}{2 \cdot n + 1}$	10	$\operatorname{arctg}(2 \cdot x)$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^{2n+1} \cdot 2^{2n+1}}{2 \cdot n + 1}$
11	$\frac{\sin(x)}{x}$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^{2n}}{(2 \cdot n + 1)!}$	12	$\sin(a \cdot x)$ $\sum_{n=0}^k \frac{a^{2n+1} \cdot x^{2n+1} \cdot (-1)^n}{(2 \cdot n + 1)!}$
13	$e^{-x^2}$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^{2n}}{n!}$	14	$e^{x+a}$ $e^a + \sum_{n=1}^k \frac{e^a \cdot x^n}{n!}$
15	$\cos(x)$ $\sum_{n=0}^k \frac{(-1)^n \cdot x^{2n}}{(2 \cdot n)!}$		

Таблица 2.18. Индивидуальные варианты задания 2.2 (задача 8)

№	Выражение	№	Выражение
1	$\lim_{x \rightarrow 0} \frac{\cos(x) - e^{-\frac{x^2}{2}}}{x^4}$	2	$\lim_{x \rightarrow 0} \frac{[e^x \cdot \sin(x) - x \cdot (x+1)]}{x^3}$
3	$\lim_{x \rightarrow 0} \left( \frac{1}{x} - \frac{1}{\sin(x)} \right)$	4	$\lim_{x \rightarrow 0} \frac{1 - \cos(x)^{\sin(x)}}{x^3}$
5	$\lim_{x \rightarrow 0} \frac{\sin(\sin(x)) - x \cdot \sqrt[3]{1-x^2}}{x^5}$	6	$\lim_{n \rightarrow \infty} \frac{\sqrt[3]{n^2} \cdot \sin(n!)}{n+1}$
7	$\lim_{n \rightarrow \infty} (\sqrt{n+1} - \sqrt{n})$	8	$\lim_{n \rightarrow \infty} \sqrt[n]{n}$
9	$\lim_{n \rightarrow \infty} \frac{a^n}{n!}$	10	$\lim_{x \rightarrow 0} \frac{\operatorname{tg}(x)}{x}$
11	$\lim_{x \rightarrow 0} \frac{e^{a \cdot x} - e^{b \cdot x}}{\sin(a \cdot x) - \sin(b \cdot x)}$	12	$\lim_{x \rightarrow 0} \frac{\sin(a+x) - \sin(a-x)}{x}$
13	$\lim_{x \rightarrow 0} \left( 1 + 3 \cdot \operatorname{tg}(x)^2 \right)^{\frac{1}{\operatorname{tg}(x)^2}}$	14	$\lim_{x \rightarrow 0} \frac{\sin\left(\frac{x}{3}\right)^2}{x^2}$
15	$\lim_{x \rightarrow \frac{\pi}{2}} \frac{\operatorname{tg}(x)}{x + 4}$		$\lim_{x \rightarrow \frac{\pi}{2}} \frac{\operatorname{tg}(x)}{x}$

### 2.2.4. Примеры выполнения заданий

Примеры выполнения задач задания 2.2 приведены в виде листингов Mathcad.

Задача 1. Упростить выражение

$$\frac{(a \cdot b^{-1} + a^{-1} \cdot b + 1) \cdot (a^{-1} - b^{-1})^2}{a^2 \cdot b^{-2} + a^{-2} \cdot b^2 - (a \cdot b^{-1} + a^{-1} \cdot b)}$$

Листинг 2.8. Результаты решения задачи 1 (задание 2.2)

$\frac{(a \cdot b^{-1} + a^{-1} \cdot b + 1) \cdot (a^{-1} - b^{-1})^2}{a^2 \cdot b^{-2} + a^{-2} \cdot b^2 - (a \cdot b^{-1} + a^{-1} \cdot b)} \text{ simplify } \rightarrow \frac{1}{a \cdot b}$
---

Задача 2. Решить уравнение в символьном виде

$$\frac{2 \cdot x - 1}{x - a} + \frac{2 \cdot x}{a} = \frac{a \cdot x - 2}{a^2 - a \cdot x}$$

Листинг 2.9. Результаты решения задачи 2 (задание 2.2)

$$\frac{2 \cdot x - 1}{x - a} + \frac{2 \cdot x}{a} = \frac{a \cdot x - 2}{a^2 - a \cdot x}$$

Выделим переменную x.  
 Воспользуемся меню  
 Символика --> Переменная --> Разрешить  
 После чего отображается результат вычислений

$$\left( \begin{array}{c} 1 \\ -\frac{1}{2} \cdot a - 1 \end{array} \right)$$

Задача 3. Решить систему уравнений

$$x - y = 8 \cdot a$$

$$\sqrt{x} + y = 4 \cdot a$$

Листинг 2.10. Результат решения задачи 3 (задание 2.2)

Given

$$x - y = 8 \cdot a$$

$$\sqrt{x} + y = 4 \cdot a$$

Find (x, y) →

$$\left[ \begin{array}{cc} 12 \cdot a + \frac{1}{2} + \frac{1}{2} \cdot (48 \cdot a + 1)^{\frac{1}{2}} & 12 \cdot a + \frac{1}{2} - \frac{1}{2} \cdot (48 \cdot a + 1)^{\frac{1}{2}} \\ 4 \cdot a + \frac{1}{2} + \frac{1}{2} \cdot (48 \cdot a + 1)^{\frac{1}{2}} & 4 \cdot a + \frac{1}{2} - \frac{1}{2} \cdot (48 \cdot a + 1)^{\frac{1}{2}} \end{array} \right]$$

Задача 4. Выполнить символьное интегрирование выражения

$$\int \frac{x^2}{\sqrt{4 \cdot x + 5}} dx$$

Листинг 2.11. Результаты решения задачи 4 (задание 2.2)

$$\int \frac{x^2}{\sqrt{4 \cdot x + 5}} dx \rightarrow \frac{1}{160} \cdot (4 \cdot x + 5)^{\frac{5}{2}} - \frac{5}{48} \cdot (4 \cdot x + 5)^{\frac{3}{2}} + \frac{25}{32} \cdot (4 \cdot x + 5)^{\frac{1}{2}}$$

Задача 5. Найти первую и вторую производную функции  $y(x) = x^2 \cdot \sqrt{\sin(x)}$

Листинг 2.12. Результаты решения задачи 5 (задание 2.2)

$$y(x) := x^2 \cdot \sqrt{\sin(x)}$$

$$\frac{d}{dx} y(x) \rightarrow 2 \cdot x \cdot \sin(x)^{\frac{1}{2}} + \frac{1}{2} \cdot \frac{x^2}{\sin(x)^{\frac{1}{2}}} \cdot \cos(x)$$

$$\frac{d^2}{dx^2} y(x) \rightarrow 2 \cdot \sin(x)^{\frac{1}{2}} + 2 \cdot \frac{x}{\sin(x)^{\frac{1}{2}}} \cdot \cos(x) - \frac{1}{4} \cdot \frac{x^2}{\sin(x)^{\frac{3}{2}}} \cdot \cos(x)^2 - \frac{1}{2} \cdot x^2 \cdot \sin(x)^{\frac{1}{2}}$$

Задача 6. Вычислить сумму числового ряда

$$\sum_{n=2}^{\infty} \frac{1}{n^2 - 1}$$

Листинг 2.13. Результат решения задачи 6 (задание 2.2)

$$\sum_{n=2}^{\infty} \left( \frac{1}{n^2 - 1} \right) \rightarrow \frac{3}{4}$$

Задача 7. Разложить функцию  $y(x) = a^{2x}$  в ряд Тейлора и проверить полученный результат путем вычисления суммы ряда по прилагаемой формуле

$$f(x, a) = 1 + \sum_{n=1}^k \frac{(\ln(a) * 2 * x)^n}{n!},$$

где значение  $a$  выбирается произвольным, а значение  $k$  задается исходя из количества членов ряда, полученных при разложении в ряд Тейлора.

Листинг 2.14. Результаты решения задачи 7 (задание 2.2)

$$a^{2 \cdot x} \text{ series, } x, 5 \rightarrow 1 + 2 \cdot \ln(a) \cdot x + 2 \cdot \ln(a)^2 \cdot x^2 + \frac{4}{3} \cdot \ln(a)^3 \cdot x^3 + \frac{2}{3} \cdot \ln(a)^4 \cdot x^4$$

$$f1(x, a) := 1 + 2 \cdot \ln(a) \cdot x + 2 \cdot \ln(a)^2 \cdot x^2 + \frac{4}{3} \cdot \ln(a)^3 \cdot x^3 + \frac{2}{3} \cdot \ln(a)^4 \cdot x^4$$

$$f2(x, a) := 1 + \sum_{n=1}^4 \frac{(\ln(a) \cdot 2 \cdot x)^n}{n!} \quad f1(2, 3) = 44.732 \quad f2(2, 3) = 44.732$$

Задача 8. Вычислить предел функции

$$\lim_{x \rightarrow 0} \frac{\sin(a \cdot x)}{\sin(b \cdot x)}$$

Листинг 2.15. Результаты решения задачи 8 (задание 2.2)

$$\lim_{x \rightarrow 0} \frac{\sin(a \cdot x)}{\sin(b \cdot x)} \rightarrow \frac{a}{b}$$

### Контрольные вопросы

1. Какими способами можно получить символическое решение уравнения?
2. Как используется блок **given ... find** для получения символического решения уравнения и системы уравнений?
3. Как найти производную n-го порядка?
4. Какие три разновидности вычисления пределов предусмотрены в Mathcad?
5. Какой оператор используется для разложения функции в ряд Тейлора и какие параметры для него задаются?
6. Выводится ли автоматически постоянная интегрирования при вычислении в Mathcad неопределенного интеграла?

### 2.3. Построение графиков

**Цель работы:** познакомиться с возможностями Mathcad для вывода данных в виде графиков и диаграмм. Освоить навыки построения графиков в декартовой и полярной системах координат. Освоить методы вывода и отображения поверхностей. Изучить средства Mathcad для отображения изолиний.

**Время:** 2 часа.

**Литература:** Л1 с.117-124.

### 2.3.1. Задание 2.3

1. Построить график функции  $y_1(x)$  в декартовой системе координат, Отобразить на одном графике три функции  $y_1(x)$ ,  $y_2(x)$ ,  $y_3(x)$  на интервале  $x \in [-10;10]$ . На втором графике изменить цвет графика функции  $y_3(x)$  и установить толщину каждой функциональной кривой равной 2.

2. Построить график функции, заданной параметрически. Отобразить сетку по осям координат.

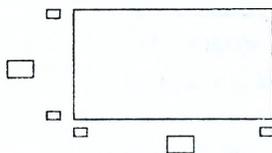
3. Построить график функции в полярной системе координат. Отобразить радианные и угловые линии сетки.

4. Построить график поверхности и отобразить в декартовой системе координат сечения поверхности в точках  $x=0$  и  $y=0$ .

5. Построить контурный график в виде изолиний с выводом числовых значений уровней изолиний.

### 2.3.2. Указания к выполнению задания

Для построения графиков в Mathcad используются шаблоны. Для начального построения графика достаточно войти в меню **ВСТАВКА, ГРАФИК** или задать тип графика на панели инструментов **“Графики”** и выбрать один из семи основных типов графиков. Шаблон для графика в декартовой системе координат (задача 1) представляет собой пустой прямоугольник, вокруг которого вдоль осей абсцисс и ординат будущего графика расположены области ввода в виде маленьких прямоугольников.



В эти прямоугольники надо поместить имя аргумента( $x$ ) для оси абсцисс и имя функции ( $y(x)$ ) для оси ординат. Если строятся графики нескольких функций в одном шаблоне, то для их разделения следует использовать запятые. Крайние маленькие прямоугольники служат для указания предельных значений координат по оси абсцисс и ординат, т. е. они задают диапазоны отображения

функции. Если предельные значения координат по оси абсцисс и ординат оставить незаполненными, то масштабы по осям будут устанавливаться автоматически. Предусмотрена возможность форматирования графика, т.е. задания толщины линий, отображаемых на графике, цвета, нанесения сетки, отображение характерных точек и т. д. Это достигается путем активизации диалогового окна **«Формат»**, на котором выбирается соответствующая закладка, и устанавливаются параметры настройки графика. Отличие задачи 2 от задачи 1 состоит в том, что по оси  $Ox$  задается обозначение  $x(t)$ , а по оси ординат -  $y(t)$ .

Построение графиков в полярной системе координат (задачи 3) аналогично построению графиков в декартовой системе координат. В нижней части шаблона задается имя угловой переменной, в левой части имя функции, определяющей радиус как функцию угла. В правой верхней части расположены два поля для задания нижнего и верхнего значения радиуса. Возможно отображение нескольких функций в рабочем поле графика. Для этого имена функций так же вводятся через запятую.

Шаблон для построения трехмерной поверхности  $F(x,y)$  в задаче 4, содержит единственное поле ввода, у левого нижнего угла основного шаблона. В него надо занести имя функции. При построении графиков сечений при  $x=0$  или  $y=0$  используется график в де-

картовой системе координат. В оси абсцисс задается имя изменяемой переменной, например для сечения в точке  $x=0$  необходимо занести  $y$ , а по оси ординат имя функции с фиксированным значением одного из аргументов. Например, для сечения в точке  $x=0$  необходимо задать вдоль оси ординат  $-F(0,y)$ . Контурный график (задача 5) представляет собой совокупность линий, каждая из которых соответствует одинаковому значению функции, зависящей от двух переменных (изолинии). Последовательность построения контурного графика аналогична последовательности построения графика поверхности. Форматирование изображения ( количество линий уровня, их значения, заливка) производится посредством диалогового окна **3-D Plot Format (Форматирование)**. Переключатели диалогового окна позволяют сформировать дизайн графика. Например, переключатели группы **Contour Options (Контурные опции)** устанавливают следующие опции графика. **Fill (Залить)** – обеспечивают закраску графика согласно цветовой палитры. **Draw Lines (Рисовать линии)** – позволяет на графике отображать линии уровня. **Auto Contour (Автоконтур)** – количество линий контура выбирается автоматически. **Numbered (Пронумерованные)** – на линии уровня выносятся их числовые значения.

### 2.3.3. Варианты заданий

Варианты задания 2.3 представлены в нижеприведенных таблицах.

Таблица 2.19. Индивидуальные варианты задания 2.3 (задача 1)

№	Выражение	№	Выражение
1	$y_1(x) = x - 5 \cdot \sin(x) \cdot$ $y_2(x) = e^{x-0.1} - 0.5$ $y_3(x) = 5 \cdot \ln(x^2 + 2)$	2	$y_1(x) = \cos(0.1 \cdot x - 1) + \sin(x) - 2$ $y_2(x) = \sqrt{\arctg(x - 1)^3 + 4} - 3.5$ $y_3(x) = \ln( x + 2  + 4) - \sqrt{ x } - 2$
3	$y_1(x) = 15 \cdot \cos(x)^2 - \sin\left(x + \frac{\pi}{4}\right)$ $y_2(x) = e^{\sqrt{ x +4}} - 10$ $y_3(x) = 10 \cdot \arctg(x - 1)$	4	$y_1(x) = 6 \cdot \cos\left[0.3x - (0.4 \cdot x)^2\right] - 1$ $y_2(x) = e^{-0.3x} + x - 4$ $y_3(x) = \frac{\ln( x^2 - x + 17  + 4)}{\ln(0.5)} + 5$
5	$y_1(x) = 10 \cdot \sin\left(x - \cos\left(x - \frac{\pi}{4}\right)\right) + 5$ $y_2(x) = \sqrt[2]{x^2 + 1}$ $y_3(x) = \ln( x - 1 ) + x$	6	$y_1(x) := 10 \cdot \sin\left[ x - 0.2  - \cos\left[(0.1x)^2\right]\right] - 1$ $y_2(x) := \sqrt[3]{x^2 + 4} - e^{x-7}$ $y_3(x) := \frac{(\ln( x^3 + x + 2  - 2))^2}{2} - 11$
7	$y_1(x) = \sin(x + \cos(x)) + \sqrt[3]{x}$ $y_2(x) = 0.1e^x - 2$ $y_3(x) = \ln(x + 2 \cdot  x  + 3) - 4$	8	$y_1(x) = 5 \cdot e^{2 \cdot x - x^2}$ $y_2(x) = 4 \cdot \ln(x^2 + 1) - 10$ $y_3(x) = ( x )^3 - 2 \cdot x^2 + 9 \cdot x - 2$

Продолжение таблицы 2.19

9	$y_1(x) = x + 15 \cdot (\sin(x))^2$ $y_2(x) = e^{\frac{2}{x}} - 0.5$ $y_3(x) = 3 - \ln(x^2 +  x )$	10	$y_1(x) = 11 \cdot e^{\sin(x)} - 10$ $y_2(x) = x \cdot \lg(x^2 + 4)$ $y_3(x) = ( x )^2 - 7 \cdot  x $
11	$y_1(x) = -4 \cos(x + 1)^2 + 8$ $y_2(x) = e^{\sin(x)} +  x $ $y_3(x) = \ln(x - \sqrt{x}) + 10$	12	$y_1(x) = \sqrt{ x } \cdot 2^{x- x }$ $y_2(x) = \frac{15}{4 + x^2}$ $y_3(x) = 3^{\cos(x)}$
13	$y_1(x) = 3 \cdot \sin(x + 1) - \cos(x)$ $y_2(x) = \sin(x) \cdot 1.3^{x-1}$ $y_3(x) = 3 \cdot \arctg(x^2) - 2$	14	$y_1(x) = 6 \cdot \cos \lfloor (0.4 \cdot x)^2 \rfloor - 1$ $y_2(x) = \ln( x  + 2) \cdot 6 \sin( x  - 3)$ $y_3(x) = e^{\frac{x}{4}} \cdot 2 \sin(x)$
15	$y_1(x) := 5 \cdot \cos \lfloor [0.2 \cdot (x)]^2 - \sin(x - 0.1) \rfloor$ $y_2(x) := \left( 1 + \frac{x^2}{8} \right)$ $y_3(x) := \ln( x - \cos(x) ) - x$		

Таблица 2.20. Индивидуальные варианты задания 2.3 (задача 2)

№	Выражение	№	Выражение
1	$x(t) = t^2$ $y(t) = t^3$	2	$x(t) = 2 \cdot \sin(t \cdot 2)^2$ $y(t) = \cos(t)^2$
3	$x(t) = 2 \cdot t^3$ $y(t) = \sin(t) t^3$	4	$x(t) = 2 \cdot \sin(2 \cdot t)^2$ $y(t) = \cos\left(t \cdot \frac{1}{2}\right)^2$
5	$x(t) = \frac{t^2}{1 + t^3}$ $y(t) = \frac{t}{1 + t^3}$	6	$x(t) = \sin(t) \cdot \cos(t)$ $y(t) = \sin(t) \cdot \cos(t)^2$
7	$x(t) := \frac{t}{1 + t^2}$ $y(t) := \frac{2 \sin(t) \cdot t^2}{1 + t^2}$	8	$x(t) = \sin(t)^2 \cdot \cos(t)$ $y(t) = \sin(t) \cdot \cos(t)^2$

Продолжение таблицы 2.20

9	$x(t) = \cos(t)^2 + \cos(t)$ $y(t) = \cos(t) \cdot \sin(t) + \sin(t)$	10	$x(t) = \operatorname{tg}(t)^3$ $y(t) = (1 - \operatorname{tg}(t)^2)^2$
11	$x(t) = 4 \cdot \cos\left(\frac{t}{4}\right)^3$ $y(t) = 4 \cdot \sin\left(\frac{t}{4}\right)^3$	12	$x(t) = \frac{\sin(t)^2 + \cos(t)^2}{t^2}$ $y(t) = \frac{\sin(t)^4 + \cos(t)^3}{t^3}$
13	$x(t) = \frac{-3 \cdot t^2}{1 + t^2}$ $y(t) = \frac{-3 \cdot \cos(t) \cdot t^2}{1 + 2 \cdot t^3}$	14	$x(t) = \frac{t^4 - t^2 + 1}{\sin(t)}$ $y(t) = \cos(t)^4$
15	$x(t) = \cos\left(\frac{2}{3}t\right) + 2 \cdot \cos\left(\frac{1}{3}t\right)$ $y(t) = \sin\left(\frac{2}{3}t\right) - 2 \sin\left(\frac{1}{3}t\right)$		

Таблица 2.21. Индивидуальные варианты задания 2.3 (задача 3)

№	Выражение	№	Выражение
1	спираль Архимеда $d = 2$ $R(a) = d \cdot a$	2	Гиперболическая спираль $k = 8$ $R(a) = \frac{k}{a}$
3	Спираль $d = 1.2$ $R(a) = d^a$	4	Улитка Паскаля $k = 4$ $R(a) = 2 + k \cdot \cos(a)$
5	2k_лепестковая_роза $b = 5 \quad k = 2$ $R(a) = b \cdot \sin(k \cdot a)^2$	6	Кохлеоида $k = 5$ $R(a) = k \cdot \frac{\sin(a)}{a}$

Продолжение таблицы 2.21

7	Кардиоида $b = 3$ $R(a) = b \cdot (1 + \sin(a))$	8	Строфоида $k = 10$ $R(a) = -k \cdot \frac{\cos(2 \cdot a)}{\cos(a)}$
9	Каппа $k = 2$ $R(a) = \frac{k}{\operatorname{tg}(a)}$	10	Кардиоида $k = 2$ $R(a) = k \cdot (1 - \cos(a))$
11	Лемниската Бернулли $d = 2$ $k = 4$ $R(a) = \sqrt{d^2 \cdot \cos(k \cdot a)}$	12	Логарифмическая _спираль $k = 2$ $b = 0.2$ $R(a) = k \cdot e^{b \cdot a}$
13	Окружность $k = 8$ $R(a) = k \cdot \sin(a)$	14	Конхоида Никомеда $k = 2$ $b = 5$ $R(a) = \frac{k}{\sin(a)} + b$
15	Квадратриса_Динострата $k = 4$ $R(a) = k \cdot \frac{(\pi - 2 \cdot a)}{\pi \cdot \cos(a)}$		

Таблица 2.22. Индивидуальные варианты задания 2.3 (задача 4,5)

№	Выражение	№	Выражение
1	$a = 2$ $b = 4$ $Z(x,y) = \left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2$	2	$a = 2$ $b = 4$ $Z(x,y) = a \cdot x^2 - b \cdot y^2$
3	$a = 2$ $b = 4$ $Z(x,y) = \left(\frac{x}{a}\right)^2 - \left(\frac{y}{b}\right)^2$	4	$a = 2$ $b = 4$ $Z(x,y) = x^3 - x \cdot y + b \cdot y^3 + a \cdot x^2$
5	$b = 4$ $a = 4$ $Z(x,y) = \frac{\cos\left(\frac{x^2 + y^2}{a}\right)}{x^2 + y^2 + 4 \cdot b}$	6	$a = 2$ $b = 4$ $Z(x,y) = \frac{x^4}{a} + \left(\frac{y}{b}\right)^4 - \frac{x^2 \cdot y^2}{a \cdot b}$

Продолжение таблицы 2.22

7	$a = 2 \quad b = 4$ $Z(x, y) = \sqrt{\left  \frac{x}{a} \right } + \sqrt{\left  \frac{y}{b} \right }$	8	$a = 2 \quad b := 4$ $Z(x, y) = \frac{\sin\left(\frac{x}{a^2}\right)}{a^2} + \frac{\cos\left(\frac{y}{b^2}\right)}{b^2}$
9	$a = 2 \quad b = 4$ $Z(x, y) = \cos\left(\frac{a \cdot x}{10}\right) \cdot \cos\left(\frac{b \cdot y}{10}\right)$	10	$a = 2 \quad b = 4$ $Z(x, y) = \sqrt{ (x+a) \cdot (y+b) }$
11	$a = 2 \quad b = 4$ $Z(x, y) = \frac{a \cdot x + b \cdot y}{x^2 + y^2 + 1 + a \cdot b}$	12	$a = 2 \quad b = 4$ $Z(x, y) = a \cdot x^2 - b \cdot y^3$
13	$a = 2 \quad b = 4$ $Z(x, y) = \sqrt{\left  \frac{x^2 + y^2}{a + b} \right }$	14	$a = 2 \quad b = 4$ $Z(x, y) = a \cdot x^2 - \left( \sin\left(\frac{y}{b}\right) \right)^2 \cdot y^2$
15	$a = 2 \quad b = 4$ $Z(x, y) = \left  \frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{x \cdot y}{a \cdot b} \right $		

### 2.3.4. Примеры выполнения заданий

Примеры выполнения задач задания 2.3 приведены в виде листингов Mathcad.

Задача 1. Построить график функции  $y_1(x)$  в декартовой системе координат. Отобразить на втором графике три функции  $y_1(x)$ ,  $y_2(x)$ ,  $y_3(x)$  на отрезке от -10 до 10. На втором графике изменить цвет для отображения функции  $y_3(x)$  и установить толщину функциональных кривых  $y_1(x)$ ,  $y_2(x)$  и  $y_3(x)$  равным 2.

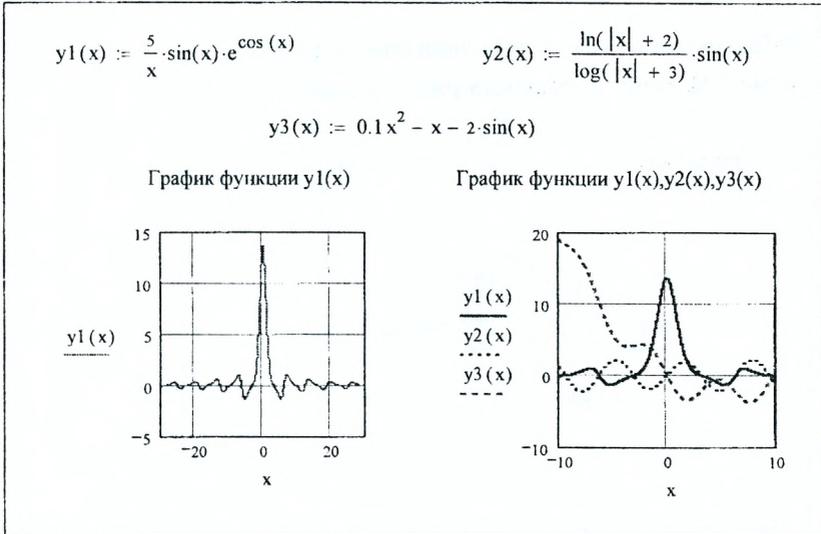
Для примера взять следующие функциональные зависимости

$$y_1(x) = \frac{5}{x} \cdot \sin(x) \cdot e^{\cos(x)}$$

$$y_2(x) = \frac{\ln(|x| + 2)}{\lg(|x| + 3)} \cdot \sin(x)$$

$$y_3(x) = 0.1 x^2 - x - 2 \cdot \sin(x)$$

Листинг 2.16. Результаты решения задачи 1 (задание 2.3)



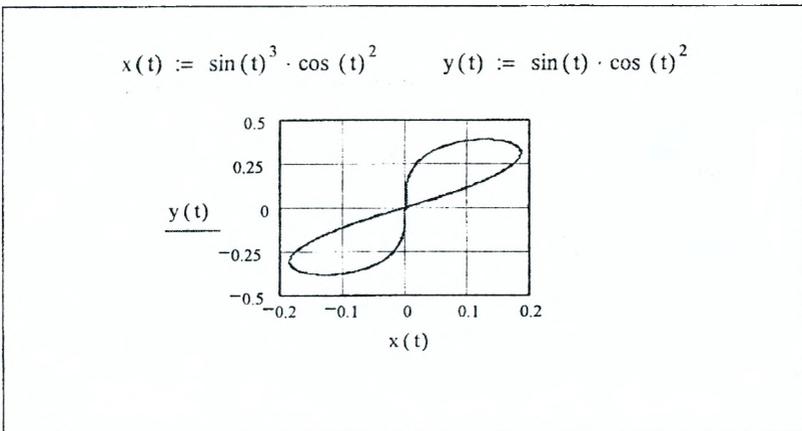
Задача 2. Построить график функции, заданной параметрически.

$$x(t) = \sin^3 t \cdot \cos^2 t$$

$$y(t) = \sin(t) \cdot \cos^2 t$$

Отобразить сетку по осям координат.

Листинг 2.17. Результат решения задачи 2 (задание 2.3)

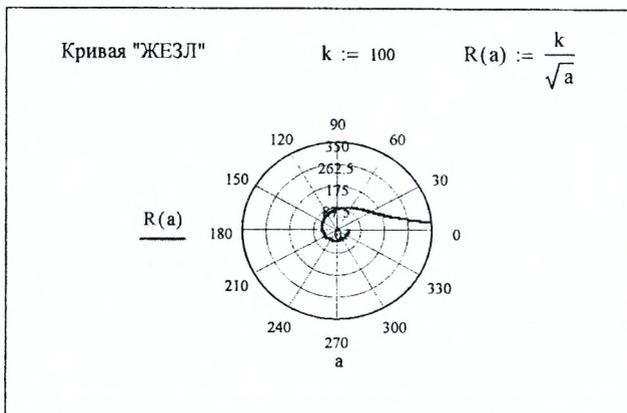


Задача 3. Построить график функции в полярной системе координат

$$R(a) = \frac{k}{\sqrt{a}}, \text{ при } k = 100$$

Отобразить радианные и угловые линии сетки на графике

Листинг 2.18. Результат решения задачи 3 (задание 2.3)

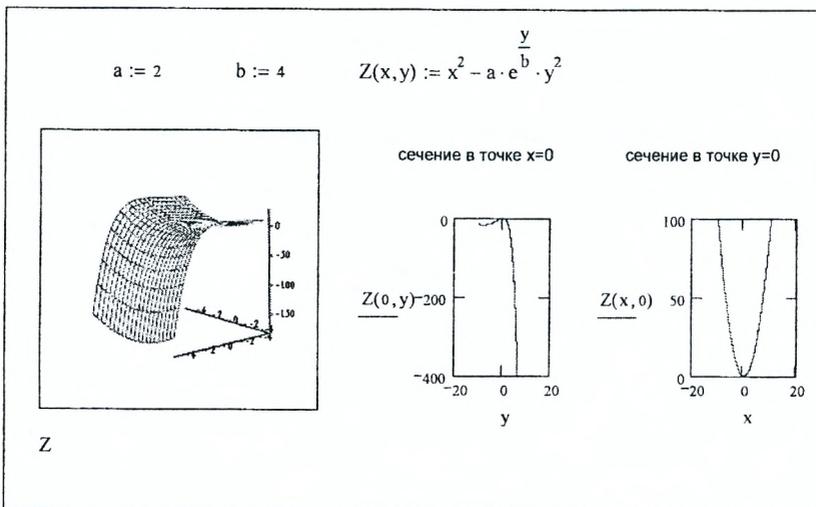


Задача 4. Построить поверхность, заданную функцией

$$Z(x, y) = x^2 - a e^{\frac{y}{b}} y^2, \text{ при } a = 2, \quad b = 4$$

Отобразить в декартовой системе координат сечения поверхности в точках  $x=0$  и  $y=0$

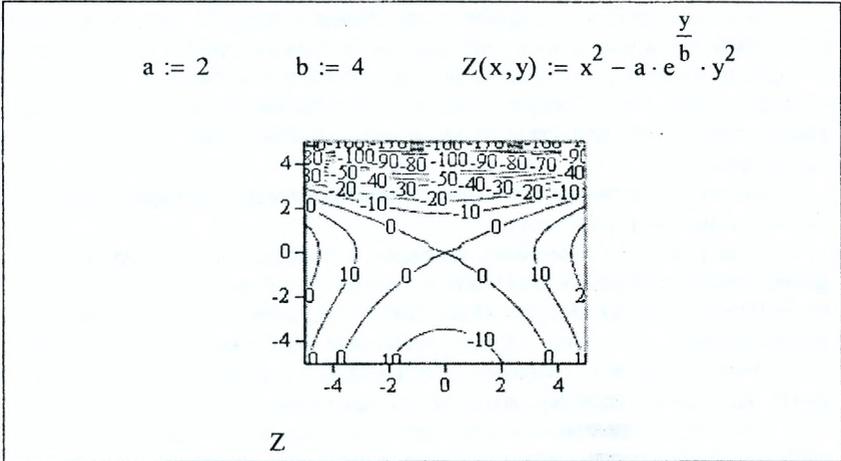
Листинг 2.19. Результат решения задачи 4 (задание 2.3)



Задача 5. Построить контурный график в виде изолиний с выводом значений уровней изолиний для функции

$$Z(x, y) = x^2 - a e^{\frac{x}{b}} y^2, \text{ при } a = 2, \quad b = 4$$

Листинг 2.20 Результат решения задачи 5 (задание 2.3)



### Контрольные вопросы

1. Как отобразить на одном графике в декартовой системе координат несколько функциональных зависимостей?
2. Какие поля для задания параметров предусмотрены для графика в декартовой системе координат?
3. Какие поля для задания параметров предусмотрены для графика в полярной системе координат?
4. Как отобразить координатную сетку, задать толщину линий для графиков в декартовой системе координат?
5. Какие поля заполняются при построении графика поверхности?
6. Как для контурного графика задать режим отображения числовых значений линий уровня?

## 2.4. Численные вычисления

**Цель работы:** познакомиться с возможностями Mathcad при выполнении численных вычислений. Овладеть навыками табулирования функций, решения уравнений, систем линейных и нелинейных уравнений, нахождения корней полиномов. Познакомиться с функциями отыскания максимума и минимума функции на отрезке. Освоить методы вычисления определенных интегралов и численного дифференцирования.

**Время:** 4 часа.

**Литература:** Л1 с.125-132.

### 2.4.1. Задание 2.4

1. Построить график функции  $y(x)$  в декартовой системе координат на отрезке  $[-10, 10]$ , затем протабулировать функцию в указанном диапазоне с заданным шагом и отобразить результаты табулирования на отрезке табулирования.

2. На основе функций и графиков, полученных в первой задаче, определить минимум (максимум) функции на отрезке табулирования, сравнить с графическими данными.

3. На основе функций и графиков, используемых в задаче 1, на отрезке табулирования найти корни уравнения с использованием функции **root()** и блока **given...find**. Отобразить результаты с точностью до 7 знаков после десятичной точки. Оценить погрешность вычислений. Сделать вывод о целесообразности использования функции **root()** и блока **given...find** для решения данного уравнения с позиций точности полученного результата.

4. Найти корни полинома с помощью функции **polyroots()**. Полученные действительные корни проверить графически.

5. Решить систему нелинейных уравнений с помощью блока **given...find** и блока **given...minerr**. Отобразить результаты с точностью до 7 знаков после десятичной точки. Рассчитать невязку ERR для обоих случаев. Сделать повторные вычисления при других начальных значениях искомых переменных. Полученные результаты сравнить.

6. Решить систему линейных алгебраических уравнений посредством блока **given...find**, сделать проверку, используя функцию **Isolve**.

7. Вычислить определенный интеграл, используя средства Mathcad. Полученный результат проверить посредством предварительного вычисления неопределенного интеграла.

8. Определить численное значение первой и второй производной функции в указанной точке.

### 2.4.2. Указания к выполнению задания

При табулировании (задача 1) необходимо задать значения дискретного аргумента на интервале табулирования, для чего вводится идентификатор дискретного аргумента и определяется область его значений. Одним из способов задания значений дискретного аргумента, является задание счетчика значений дискретного аргумента, который определяет количество точек на интервале табулирования, в которых рассчитывается значение функции, например,

$i := \text{начальное значение} .. \text{конечное значение}$

По умолчанию шаг изменения значений счетчика дискретного аргумента равен 1.

Двоеточие ".." вводится символом точка с запятой ";" или кнопкой  панели инструментов «Матрицы».

Изменение аргумента задается как вектор в формате

$$x_i := x_0 + i * \Delta x$$

В некоторых случаях возможно непосредственное задание значений дискретного аргумента, при этом нет необходимости задавать счетчик значений дискретного аргумента. Отображение функции и результатов табулирования производится на одном графике в декартовой системе координат.

В задаче 2 для нахождения минимума и максимума функции на отрезке табулирования используются функции **Minimize()** и **Maximize()**. Так как функция может иметь несколько локальных экстремумов, а функции **Minimize()** и **Maximize()** позволяют найти только одно значение, то дополнительно анализируется график, полученный в задаче 1. На основе графика определяются минимумы и максимумы на отрезке для задания начального приближения переменной  $x$ . В результате находится значение экстремума функции  $y(x)$ , ближайшее к заданному начальному приближению.

В задаче 3 для отыскания корней на отрезке табулирования для функции  $y(x)$ , приведенной в задаче 1, используются функция **root()** и блок **given...find**, в которых требуется задание начального приближения искомой переменной  $x$ . Для задания начального приближения, анализируется график, полученный в задаче 1. Повышение точности представления результатов решения задачи (с точностью до 7 знаков после десятичной точки) обеспечивается посредством меню **Формат, Результат**.

Для нахождения корней полинома в задаче 4 применяется функция **polyroots()**. Функция позволяет найти как действительные, так и комплексные корни полинома. Для оценки результатов решения задачи строится график функции в декартовой системе координат и оценивается правильность нахождения действительных корней полинома.

В задаче 5 для решения системы нелинейных уравнений используются блоки **given...find** и **given...minerr**. При применении данных блоков также задаются начальные приближения искомых переменных. В данной задаче предлагается исследовать влияние начальных значений на точность результатов решения задачи. Для чего требуется произвести двойной расчет при различных начальных значениях и представить результаты вычислений с точностью до 7 знаков после десятичной точки.

В задаче 6 для решения системы линейных алгебраических уравнений используется блок **given...find**. Для получения численного решения также требуется задание начальных значений искомых переменных, но на результаты решения задачи они не влияют. Для проверки результатов решения используется функция **Isolve(M,b)**, где  $M$  – матрица коэффициентов при неизвестных,  $b$  – вектор свободных членов. Ввод и формирование  $M$  и  $b$  осуществляется при помощи панели инструментов **«Матрица»** или меню **Вставка, Матрица**.

В основу расчетов (задача 7) положена формула

$$\int_a^b f(x) dx = \int f(x) d(x) \Big|_{x=a} - \int f(x) dx \Big|_{x=a}$$

В соответствии с вышеприведенной формулой в задаче требуется получить результат посредством вычисления определенного интеграла, а затем проверить решение путем определения неопределенного интеграла с последующей подстановкой значений  $x = b$  и  $x = a$ .

При решении задачи 8 первоначально выполняется операция символьного дифференцирования с последующим расчетом численного значения производной в указанной точке. При этом можно ввести функцию пользователя как производную от указанной функции с последующим вычислением числового значения функции пользователя от указанного значения числового аргумента. Производную второго порядка можно получить либо повторным дифференцированием, либо использованием специального шаб-

лона  $\frac{d^2}{dx^2}$  на панели инструментов **«Исчисление»**

### 2.4.3. Варианты заданий

Варианты задания 2.4 представлены в нижеприведенных таблицах.

Таблица 2.23. Индивидуальные варианты задания 2.4 (задача 1, 2, 3)

№	Выражение	№	Выражение
1	$y(x) = 0.1x^3 - 2 \cdot x^2 + 8 \cdot x - 4$ <p>Диапазон изменения <math>x=[-1,6]</math> с шагом 0.5</p>	2	$y(x) = \frac{\sin(x) + 12}{e^{\left(\frac{x}{4}\right)}} \cdot \cos(x)$ <p>Диапазон изменения <math>x=[-3,3]</math> с шагом 0.5</p>
3	$y(x) = \ln( x  + 4) \cdot \operatorname{arctg}(4 \cdot x)^2 - 3$ <p>Диапазон изменения <math>x=[-3,3]</math> с шагом 0.5</p>	4	$y(x) = e^{\left(\frac{x}{4}\right)} \cdot \cos(x)$ <p>Диапазон изменения <math>x=[0,6]</math> с шагом 0.5</p>
5	$y(x) = \sqrt{\frac{\cos(x)^2 + 1}{\sin(0.1 \cdot x) + 1}} - 2$ <p>Диапазон изменения <math>x=[-8,-5]</math> с шагом 0.25</p>	6	$y(x) = 4 \cdot \frac{\ln(\cos(x) + 2) - 1}{e^{\sin(x)}}$ <p>Диапазон изменения <math>x=[0,6]</math> с шагом 0.5</p>
7	$y(x) = \frac{\sqrt{7 \cdot ( x )^2 - 5 \cdot  x  + 2} - 9}{e^{\sin(x)}}$ <p>Диапазон изменения <math>x=[-2,4]</math> с шагом 0.5</p>	8	$y(x) = (1+x) \cdot \sqrt{2+x^2} \cdot \sqrt[3]{3+x^3} - 3$ <p>Диапазон изменения <math>x=[-3,3]</math> с шагом 0.5</p>
9	$y(x) = \cos(0.5 \cdot x)^{\frac{1-x^2}{1+x^2}} - 3$ <p>Диапазон изменения <math>x=[-3,3]</math> с шагом 0.5</p>	10	$y(x) = \sin(x) \sqrt{2+x^2} \cdot \sqrt[3]{3+x^3} - 3$ <p>Диапазон изменения <math>x=[-2,2]</math> с шагом 0.4</p>
11	$y(x) = \frac{2+x}{\cos(2-x)^3} \cdot x^{(-\cos(4x)^2)} + 8 \cdot \sin(x)$ <p>Диапазон изменения <math>x=[1,3]</math> с шагом 0.25</p>	12	$y(x) := \frac{\cos(x)}{\sin(x)^2 + 1}$ <p>Диапазон изменения <math>x=[-3,3]</math> с шагом 0.5</p>

Продолжение таблицы 2.23

13	$y(x) = 10e^{-x^2} \cdot \cos\left(\sqrt[3]{x^9 - 1}\right)$ <p>Диапазон изменения <math>x=[1, 1.8]</math> с шагом 0.08</p>	14	$y(x) = \frac{x^2 - 5}{\sqrt[3]{1 + x \cdot \sin(x)} + \sqrt[3]{\cos(x)}}$ <p>Диапазон изменения <math>x=[-3, 3]</math> с шагом 0.5</p>
15	$y(x) = 3\sqrt[4]{4 - \cos(x)} \cdot \frac{x^2 - 3}{(3 + x) \cdot (4 + x)}$ <p>Диапазон изменения <math>x=[-2, 3]</math> с шагом 0.5</p>		

Таблица 2.24. Индивидуальные варианты задания 2.4 (задача 4)

№	Выражение	№	Выражение
1	$t^3 - 3 \cdot t^2 + 9 \cdot t - 8 = 0$	2	$t^3 + 0.2 \cdot t^2 + 0.5 \cdot t - 2 = 0$
3	$t^3 - 6 \cdot t - 8 = 0$	4	$t^3 - 3 \cdot t^2 + 12 \cdot t - 9 = 0$
5	$t^3 - 3 \cdot t^2 + 6 \cdot t + 3 = 0$	6	$t^3 - 0.2 \cdot t^2 + 0.3 \cdot t - 1.2 = 0$
7	$t^3 - 0.1 \cdot t^2 + 0.4 \cdot t - 1.5 = 0$	8	$t^3 - 3 \cdot t^2 + 6 \cdot t - 2 = 0$
9	$t^3 - 3 \cdot t^2 + 9 \cdot t + 2 = 0$	10	$t^3 + 3 \cdot t^2 + 6 \cdot t - 1 = 0$
11	$t^3 + t - 5 = 0$	12	$t^3 + 0.1 \cdot t^2 + 0.4 \cdot t - 1.2 = 0$
13	$t^3 + 0.2 \cdot t^2 + 0.5 \cdot t - 1.2 = 0$	14	$t^3 + 4 \cdot t - 6 = 0$
15	$t^3 + 3 \cdot t + 1 = 0$		

Таблица 2.25. Индивидуальные варианты задания 2.4 (задача 5)

№	Выражение	№	Выражение
1	$a = 1 \quad b = 7.5$ $a \cdot x - \operatorname{tg}(x \cdot y) = 0$ $(y^2 - b^2) + \ln(x) = 0$	2	$y - \sin(x) = 0$ $x^2 + y^2 - 1 = 0$
3	$b = 2.1$ $a = 0.16 \quad c = 1$ $a \cdot x + b \cdot y + x^2 \cdot y = 0$ $\cos(y) + c \cdot x = 0$	4	$4^x \cdot 5^y = 20$ $2 \cdot \frac{\ln(x)}{\ln(4)} = \frac{\ln(y)}{\ln(0.2)} \cdot \frac{\ln(5)}{\frac{1}{2} \cdot \ln(4)}$

Продолжение таблицы 2.25

5	$a = 0.4 \quad b = 3.5 \quad c = -1.5$ $d = 0.2 \quad f = 0.5$ $\sin(x + a) + b \cdot y + c = 0$ $\cos(y + d) + f \cdot x = 0$	6	$x \cdot y = 3$ $\left  \frac{\ln(x+y)}{\ln(2)} \right  + \left  \frac{\ln(x-y)}{\ln(2)} \right  = 3$
7	$a = 1 \quad b = 1.5 \quad c = 2$ $\operatorname{tg}(x \cdot a) - \cos(b \cdot y) = 0$ $c \cdot y^3 - x^2 - 4 \cdot x - 3 = 0$	8	$\frac{\ln(y^2 - 2 \cdot y + 1)}{\ln(1+x)} + \frac{\ln(x^2 + 5 \cdot x + 1)}{\ln(1+y)} = 4$ $\left  \frac{\ln(2 \cdot y + 1)}{\ln(1+x)} \right  + \left  \frac{\ln(2 \cdot x + 1)}{\ln(1+y)} \right  = 3$
9	$a = 3.0 \quad b = 14.0 \quad c = 3.0$ $d = 0.1 \quad f = -1$ $a \cdot x^3 + b \cdot y^2 - 1 = 0$ $\sin(c \cdot x + d \cdot y) - f \cdot x = 0$	10	$2x^2 - y^2 = 1$ $y - x^{\frac{2}{3}} = 0$
11	$a = 2$ $\frac{2}{x^3} + \frac{2}{y^3} = 2 \cdot a$ $x^2 - a \cdot y = 0$	12	$2x + 2^x = y + 2^y$ $x^2 + x \cdot y + y^2 = 8$
13	$a = 2$ $y - \sqrt{x+1} = 0$ $x^2 + y^2 - a \cdot y = 0$	14	$x^2 - 2 \cdot x - 3 = \ln(y - 4)$ $\ln(x) + (y + 3) = 12$
15	$y \cdot \sin(x) = \ln\left(\left \frac{y \cdot \sin(x)}{1 + 3 \cdot y}\right \right)$ $y = \frac{2 \cdot 4^{\sin(x)} - 3}{25 - 6 \cdot (4^{\sin(x)} + 4^{\cos(x)})^2}$		

Таблица 2.26. Индивидуальные варианты задания 2.4 (задача 6)

№	Выражение	№	Выражение
1	$8.2x1+3.4x2+1.2x3-1.5x4=-13.3$ $-1.1x1+7.7x2+1.5x3-3.2x4=8.4$ $-0.5x1+1.2x2+8.6x3+1.8x4=-11.6$ $-1.2x1-0.8x2-0.6x3+10x4=5.7$	2	$8.5x1-0.5x2+0.8x3-1.4x4=4.8$ $-3.2x1+11.3x2+1.2x3-1.1x4=12.4$ $-1.7x1-0.6x2+10.8x3-1.2x4=11.5$ $-2.1x1+1.6x2-3.6x3+10x4=-8.8$
3	$8.7x1-2.3x2+4.4x3+0.5x4=21.3$ $-2.4x1+10x2+3.1x3-1.5x4=-1.8$ $-0.6x1-1.5x2+10x3+2.3x4=14.4$ $-1.2x1+0.8x2+0.5x3+10x4=24.4$	4	$10x1-2.2x2+1.1x3-3.1x4=27$ $-3.8x1+10x2+1.2x3-2.2x4=-15$ $-1.1x1-2.3x2+10x3+4.1x4=12$ $-1.7x1+2.1x2-3.1x3+10x4=-1.7$
5	$8.3x1-3.1x2+1.8x3-2.2x4=-17.1$ $2.1x1+10x2-3.3x3-2.2x4=6.2$ $-3.2x1+1.8x2+9.5x3+1.9x4=-8.9$ $-1.2x1-2.8x2+1.4x3+10x4=9.4$	6	$9.3x1+0.8x2-1.1x3+1.8x4=-5.1$ $-1.8x1+4.8x2-2.1x4=11.7$ $-1.3x1-3.1x2+10x3=-10.2$ $-0.8x1+3.3x3+7.2x4=-2.8$
7	$7.7x1+1.4x2-0.6x3+1.2x4=12.1$ $-1.2x1+10x2-3.2x3+1.8x4=-7.2$ $-0.8x1+1.2x2+7.7x3-3.2x4=-5.8$ $-2.5x1-2.2x2-1.4x3+8.6x4=15.6$	8	$9.5x1+0.6x2+1.2x3-1.4x4=-21.7$ $-0.4x1+11.2x2-0.8x3-1.1x4=14$ $-3.4x1-0.8x2+10.6x3-1.4x4=-21$ $-1.1x1-1.2x2+10.3x4=-8$
9	$8.7x1-2.7x2+2.2x3+1.8x4=12.1$ $2.1x1+10x2+1.5x3-1.8x4=-3.3$ $-1.2x1-1.3x2+13.3x3-1.8x4=-4.8$ $-3.3x1+0.5x2-0.6x3+12.8x4=-1.7$	10	$9.2x1+0.3x2+0.4x4=-12$ $6.0x2-2.7x3+0.8x4=8.1$ $-3.3x1+10.7x3-2.1x4=-9.2$ $-1.1x1-0.3x3+4.2x4=1.7$

Продолжение таблицы 2.26

11	$8.6x1 - 2.3x2 - 1.8x3 - 1.7x4 = -14.2$ $-1.2x1 + 11.4x2 - 0.8x3 - 0.9x4 = -8.3$ $-1.6x1 - 2.4x2 + 10x3 + 3.5x4 = 12.1$ $-2.3x1 + 0.8x2 - 0.5x3 + 7.5x4 = 6.5$	12	$8.8x1 + 2.3x2 - 2.5x3 + 1.6x4 = 12.4$ $-1.4x1 + 6.6x2 + 1.8x3 - 2.4x4 = -8.9$ $-3.3x1 - 0.3x2 + 8.4x3 + .2x4 = 11.5$ $-1.2x1 + 0.5x2 + 8.5x4 = -5.7$
13	$7.6x1 - 2.1x2 - 0.6x3 + 3.4x4 = 14.2$ $-0.5x1 + 10x2 - 3.2x3 - 1.2x4 = -5.7$ $-3.5x1 + 2.7x2 + 10x3 + 0.5x4 = 6.8$ $1.2x1 - 4.3x2 - 0.4x3 + 12.1x4 = -1.4$	14	$9.9x1 - 0.2x2 + 0.2x3 - 0.8x4 = -13$ $-0.3x1 + 7.2x2 - 3.3x3 + 0.7x4 = 11$ $-0.9x1 - 1.3x2 + 5.8x3 - 2.8x4 = -17$ $-1.9x1 + 2.3x2 - 0.8x3 + 6.3x4 = 15$
15	$8.3x1 - 0.27x2 + 1.3x3 + 1.1x4 = -14.2$ $-1.3x1 + 11.2x2 - 0.9x3 + 0.6x4 = 4.8$ $-1.1x1 - 0.5x2 + 10.2x3 - 1.2x4 = -23.4$ $-1.3x1 - 1.8x2 - 2.4x3 + 5.7x4 = 7.2$		

Таблица 2.27. Индивидуальные варианты задания 2.4 (задача 7)

№	Выражение	№	Выражение
1	$\int_1^3 \frac{x^2}{\sqrt{9+x^3}} dx$	2	$\int_{\frac{\pi}{6}}^{\frac{\pi}{2}} \frac{1 + \frac{1}{\operatorname{tg}(x)}}{\sin(x)} dx$
3	$\int_2^4 \frac{x^3 + x^2 + 2}{x \cdot (x^2 - 1)^2} dx$	4	$\int_2^5 \cos(x) \cdot (x^2 + x + 1) dx$
5	$\int_0^{\frac{\pi}{2}} \frac{\cos(x)}{\sin(x)^2 + 1} dx$	6	$\int_2^4 \cos(x) \cdot (x - \sin(x)) dx$

Продолжение таблицы 2.27

7	$\int_2^5 \frac{1}{\sqrt{5+4 \cdot x-x^2}} dx$	8	$\int_2^4 \frac{(x+2)}{\sqrt{6-x}} dx$
9	$\int_0^1 \frac{\arcsin\left(\frac{x}{2}\right)}{\sqrt{4-x^2}} dx$	10	$\int_2^3 x^2 \cdot e^{-2x} dx$
11	$\int_0^{\frac{\pi}{2}} \frac{\cos(x)}{4+\sqrt{\sin(x)}} dx$	12	$\int_0^{\pi} x e^{3-x^2} dx$
13	$\int_2^5 \frac{x^2}{(x-1) \cdot \sqrt{x-1}} dx$	14	$\int_{\frac{\pi}{10}}^{\frac{\pi}{4}} \frac{x}{\cos(x)^2} dx$
15	$\int_0^{\ln(5)} \frac{e^x \cdot \sqrt{e^x-1}}{e^x+3} dx$		

Таблица 2.28. Индивидуальные варианты задания 2.4 (задача 8)

№	Выражение	№	Выражение
1	$f(x) = (3 \cdot x^2 + x - 1.6) \cdot \sin(x - 1.1)$ $x = 2.3$	2	$f(x) = \frac{x^3 - 2 \cdot x + 1}{x^2 + 3}$ $x = 5$
3	$f(x) = \ln(1+x) \cdot \cos(x-3)$ $x = 3$	4	$f(x) = \frac{e^{\cos(x)}}{x}$ $x = 1$
5	$f(x) = (x+2)^2 \cdot \sin(x+3)$ $x = 2$	6	$f(x) = \sqrt{\operatorname{tg}(x) + \ln(x)}$ $x = 22$
7	$f(x) = (3 \cdot x^2 + x - 3) \cdot \sin(x-2)$ $x = 3$	8	$f(x) = \ln(x) \cdot \arccos(x-2)$ $x = 2$
9	$f(x) = (x^2+3) \cos(2x-3)$ $x = 3$	10	$f(x) = \cos(x) \cdot \arccos(x-2)$ $x = 1.5$
11	$f(x) = \ln(x) \cdot \operatorname{tg}(x-2)$ $x = 1$	12	$f(x) = \frac{\sin(x)}{\arccos(x)}$ $x = 0.5$

Продолжение таблицы 2.28

13	$f(x) = \sqrt{\ln(x)} \cdot \operatorname{tg}(x)$ $x = 3$	14	$f(x) = \cos(x)^{\sin(x)}$ $x = 1.5$
15	$f(x) = e^{\sin(x)} \cdot \sin(x)$ $x = \frac{\pi + 1}{2}$		

### 2.4.4 Примеры выполнения заданий

Примеры выполнения задания 2.3 приведены в виде листингов Mathcad.

Задача 1. Построить график функции

$$y(x) = \frac{\cos(x) + 0,5}{2 + \sin(x)}$$

в декартовой системе координат на отрезке  $[-10; 10]$ . Затем протабулировать функцию в диапазоне  $[-3; 3]$  с шагом 0,5. Отобразить результаты табулирования.

Листинг 2.21. Результаты решения задачи 1 (задание 2.4)

$$y(x) := \frac{\cos(x) + 0.5}{2 + \sin(x)}$$

Графические результаты табулирования

$$i := 0..12$$

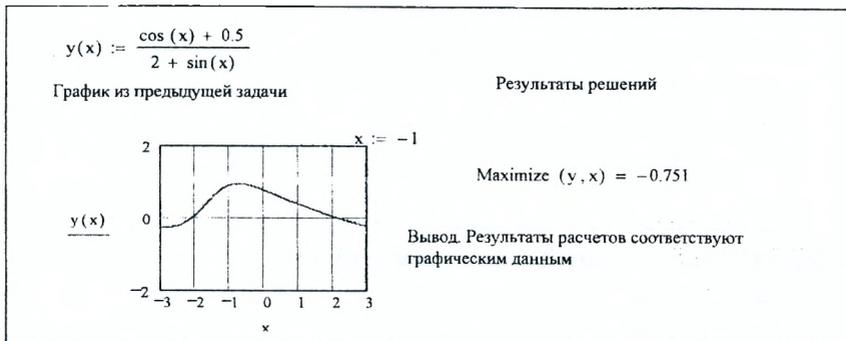
$$t_i := -3 + i \cdot 0.5$$

Численные результаты табулирования

$t_i =$	$y(t_i) =$
-3	-0.264
-2.5	-0.215
-2	0.077
-1.5	0.569
-1	0.898
-0.5	0.906
0	0.75
0.5	0.556
1	0.366
1.5	0.19
2	0.029
2.5	-0.116
3	-0.229

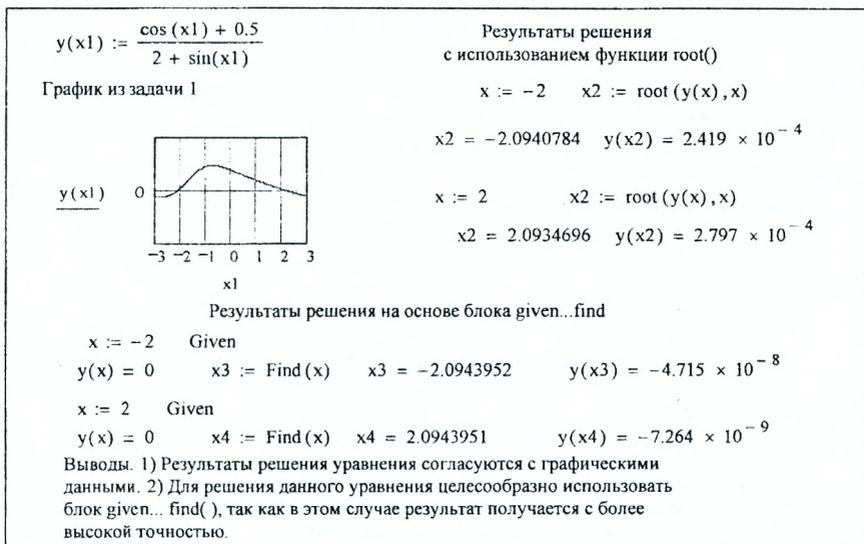
Задача 2. Для функции  $y(x) = \frac{\cos(x) + 0,5}{2 + \sin(x)}$  в диапазоне  $[-3; 3]$ , руководствуясь графиком, полученным в предыдущей задаче, определить экстремум (максимум или минимум) функции на отрезке и сравнить его с графическими данными.

Листинг 2.22. Результаты решения задачи 2 (задание 2.4)



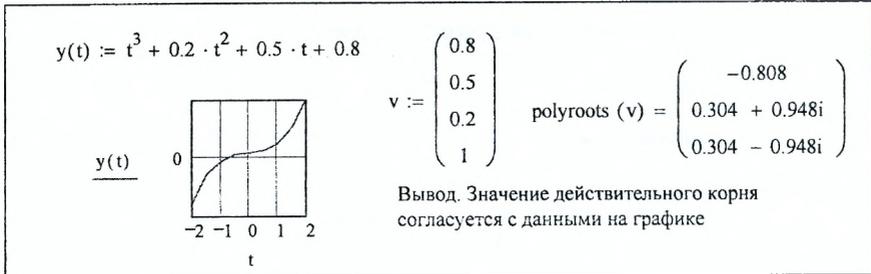
Задача 3. Для функции  $y(x) = \frac{\cos(x) + 0,5}{2 + \sin(x)}$  в диапазоне  $[-3; 3]$ , руководствуясь графиком, задать начальные приближения и с использованием функции root() и блока given... find определить корни. Отобразить результаты с точностью до 7 знаков после десятичной точки. Оценить погрешность вычислений. Сделать вывод о целесообразности использования функции root() и блока given... find для решения данного уравнения с позицией точности полученного результата.

Листинг 2.23. Результаты решения задачи 3 (задание 2.4)



Задача 4. Определить корни полинома  $y(t) = t^3 + 0,2t^2 + 0,5t + 0,8$ . Полученные действительные корни проверить графически.

Листинг 2.24. Результаты решения задачи 4 (задание 2.4)

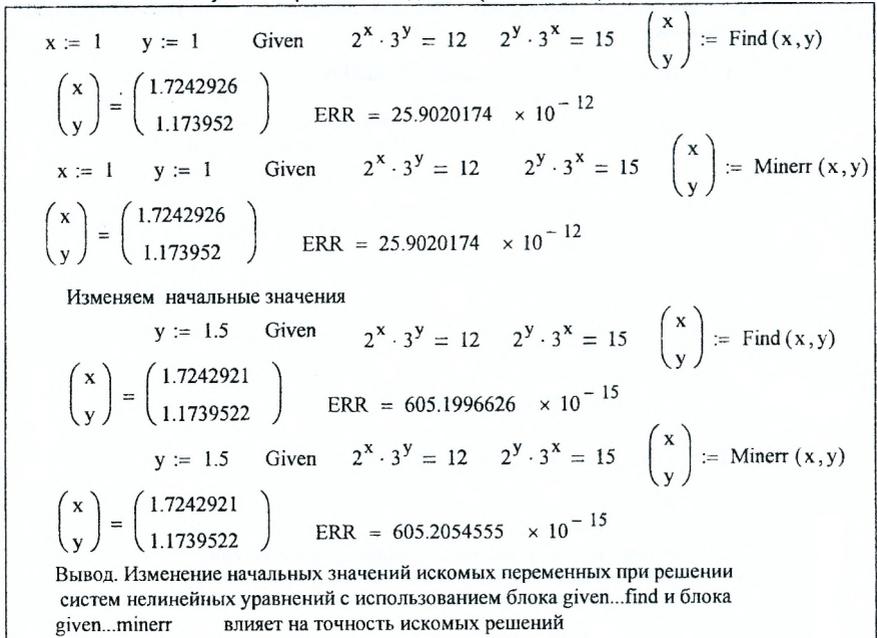


Задача 5. Решить систему нелинейных уравнений вида

$$\begin{cases} 2^x \cdot 3^y = 12 \\ 2^y \cdot 3^x = 15 \end{cases}$$

с помощью блока given...find и блока given...minerr. Представить результаты с точностью до 7 знаков после десятичной точки. Рассчитать невязку ERR для обоих случаев. Сделать повторные расчеты при других начальных значениях искомых переменных. Сравнить полученные результаты.

Листинг 2.25. Результаты решения задачи 5 (задание 2.4)



Задача 6. Решить систему линейных алгебраических уравнений вида

$$5.8 \cdot x_1 + 3 \cdot x_2 - 0.3 \cdot x_3 = 4.4$$

$$-1.1 \cdot x_1 + 12.6 \cdot x_2 + 3.6 \cdot x_3 = 14.2$$

$$-1.2 \cdot x_1 - 0.8 \cdot x_2 + 11.4 \cdot x_3 + 2.4 \cdot x_4 = -8.$$

$$-1.5 \cdot x_1 + 3.5 \cdot x_2 + 1.8 \cdot x_3 + 10 \cdot x_4 = -14.$$

с использованием блока given ... find. Сделать проверку, используя функцию Isolve().

Листинг 2.26. Результаты решения задачи 6 (задание 2.4)

```

x1 := 1      x2 := 1      x3 := 1      x4 := 1

Given

5.8 · x1 + 3 · x2 - 0.3 · x3 = 4.4
-1.1 · x1 + 12.6 · x2 + 3.6 · x3 = 14.2
-1.2 · x1 - 0.8 · x2 + 11.4 · x3 + 2.4 · x4 = -8.3
-1.5 · x1 + 3.5 · x2 + 1.8 · x3 + 10 · x4 = -14.2

⎛ x1 ⎞      ⎛ x1 ⎞      ⎛ 0.1192608 ⎞
⎛ x2 ⎞ := Find(x1 , x2 , x3 , x4)      ⎛ x2 ⎞ = ⎛ 1.2105069 ⎞
⎛ x3 ⎞      ⎛ x3 ⎞      ⎛ -0.2558889 ⎞
⎛ x4 ⎞      ⎛ x4 ⎞      ⎛ -1.7797283 ⎞
ERR = 0

M := ⎛ 5.8   3   -0.3  0 ⎞      b := ⎛ 4.4 ⎞
⎛ -1.1  12.6  3.6   0 ⎞      ⎛ 14.2 ⎞
⎛ -1.2  -0.8  11.4  2.4 ⎞      ⎛ -8.3 ⎞
⎛ -1.5  3.5   1.8  10 ⎞      ⎛ -14.2 ⎞

⎛ x1 ⎞      ⎛ x1 ⎞      ⎛ 0.1192608 ⎞
⎛ x2 ⎞ := Isolve(M , b)      ⎛ x2 ⎞ = ⎛ 1.2105069 ⎞
⎛ x3 ⎞      ⎛ x3 ⎞      ⎛ -0.2558889 ⎞
⎛ x4 ⎞      ⎛ x4 ⎞      ⎛ -1.7797283 ⎞
ERR = 0

Вывод. Результаты вычислений в обоих случаях одинаковы

```

Задача 7. Вычислить определенный интеграл вида

$$\int_0^{-2} \frac{1}{\sqrt{1+x} + \sqrt{(1+x)^3}} dx$$

Проверить полученный результат путем предварительного нахождения неопределенного интеграла.

Листинг 2.27. Результаты решения задачи 7 (задание 2.4)

$$\int_0^2 \frac{1}{\sqrt{1+x} + \sqrt{(1+x)^3}} dx = 0.524$$

$$f(x) := \int \frac{1}{\sqrt{1+x} + \sqrt{(1+x)^3}} dx \rightarrow \frac{2}{\left[ \frac{(1+x)^3}{(1+x)^2} \right]^{\frac{1}{2}}} \cdot \operatorname{atan} \left[ \frac{\left[ \frac{(1+x)^3}{(1+x)^2} \right]^{\frac{1}{2}}}{(1+x)^{\frac{3}{2}}} \right] \cdot (1+x)^{\frac{1}{2}}$$

$$f(2) - f(0) = 0.524$$

Задача 8. Определить численное значение для первой и второй производной для функции  $f(x) = \ln(x) \cdot \arcsin(x)$  в точке  $x=0,5$

Листинг 2.28. Результаты решения задачи 8 (задание 2.4)

$$f(x) := \ln(x) \cdot \arcsin(x) \quad x := 0.5$$

$$f1(x) := \frac{d}{dx} f(x) \rightarrow \frac{1}{x} \cdot \arcsin(x) + \frac{\ln(x)}{(1-x^2)^{\frac{1}{2}}} \quad f1(x) = 0.247$$

$$f2(x) := \frac{d^2}{dx^2} f(x) \rightarrow \frac{-1}{x^2} \cdot \arcsin(x) + \frac{2}{x \cdot (1-x^2)^{\frac{1}{2}}} + \frac{\ln(x)}{(1-x^2)^{\frac{3}{2}}} \cdot x \quad f2(x) = 1.991$$

### Контрольные вопросы

1. В чем отличие результатов вычисления корней полинома, получаемых при помощи функции **root()** и **polyroots()**?
2. Для каких вычислений используется функция **Isolve()**?
3. В каких задачах используются функции **Minimize(y,x)** и **Maximize(y,x)**?
4. Что определяют начальные значения переменных при использовании функции **root()**, **Minimize(y,x)** и **Maximize(y,x)**?
5. Как вычислить численное значение производной в точке?
6. Для чего используется **ERR**?
7. Как можно решить систему нелинейных уравнений?

## 2.5. Матричные вычисления

**Цель работы:** познакомиться с возможностями Mathcad при работе с векторами и матрицами. Изучить функции, используемые для создания, редактирования и преобразования векторов и матриц. Освоить методы использования векторов и матриц при решении прикладных задач.

**Время:** 2 часа.

**Литература:** Л1 с.132-136.

### 2.5.1. Задание 2.5

1. Создать матрицы **A** и **B** размерностью 4 x 4 и вектор – столбец **d** с четырьмя элементами. Заполнить матрицы и вектор значениями, каждое из которых вычисляется по формуле. Ввести значение константы  $\pi$ .

Для матрицы **A**

$$F_a(x, y) = \frac{[20 \cdot (x + 1)]}{\pi \cdot (y + 1)} \cdot \sin\left(\frac{k \cdot \sin(k \cdot x + 1)}{\cos(y \cdot k + 1) + 1}\right) \cdot \frac{\sqrt{x + 1}}{\sqrt{y + 1}}$$

для матрицы **B**

$$F_b(x, y) = \left(\frac{x + 1}{y + 1}\right)^{\sin(k+1)} \cdot \ln\left[\frac{[k + (x + 1) \cdot (y + 1)]}{k}\right] \cdot 10 e^{\cos(k+2)}$$

для вектора **d**

$$F_d(x, y) = \frac{\ln[(x + 1) + k] \cdot \cos[2 \cdot k \cdot (x + 1)]}{\lg[(x + 1) + k]}$$

для константы  $\pi$

$$\pi = \sin(k) \cdot 10$$

где  $x$  - номер строки;  $y$  – номер столбца, для вектора **d** значение  $y$  задается = 1;  $k$  - номер варианта.

Вычислить минимальный и максимальный элемент матрицы и вектора. Определить среднее значение элементов матрицы и вектора.

2. Выполнить указанные ниже операции над матрицами, вектором и константой.

а)  $A+B$ ; б)  $A-B$ ; в)  $A*B$ ; г)  $B*A$ ; д)  $A*d$ ; е)  $\pi*B$ ; ж)  $\pi*d$ ; з)  $A^T$ ; и)  $B^2$ ; к)  $A^{-1}$ ; л)  $B^{-3}$ ;

Проверить справедливость тождеств для матрицы **A**:  $A^T A = A^2$ ;  $A^{-3} = (A^{-1})^T (A^{-1})^T (A^{-1})$

Для каждого варианта вычислить матричное выражение (табл. 2.29) по частям и по единой формуле. Результаты сравнить.

3. Создать новую матрицу **C** из элементов матрицы **A** и **B**.

В матрицу **C** включить: в первый столбец матрицы **C** - первый столбец матрицы **A**; во второй столбец матрицы **C** - второй столбец матрицы **B**; в правую верхнюю четверть матрицы **C** - правую верхнюю четверть матрицы **A**; в правую нижнюю четверть матрицы **C** - правую нижнюю четверть матрицы **B**.

4. Решить систему линейных алгебраических уравнений вида  $C*x = d$ , где **C** - матрица коэффициентов при неизвестных, полученная в задаче 3, **d** – вектор свободных членов, сформированный в задаче 1,  $x$  – вектор искомых неизвестных. Для решения системы линейных алгебраических уравнений использовать метод обратной матрицы, метод Крамера и функцию `lsolve()`.

Таблица 2.29. Варианты заданий для вычисления матричных выражений

№ варианта	Выражение	№ варианта	Выражение
1	$2*(A+B)*(2*B-A)$	9	$2*A - (A^2+B)*B$
2	$3*A - (A+2*B)*B$	10	$3*(A^2 - B^2) - 2*A*B$
3	$2*(A-B)*(A^2 + B)$	11	$(2*A-B)*(3*A+B) - 2*A^2$
4	$(A^2 - B^2) * (A + B)$	12	$A*(A^2*B) - 2*(B+A)*B$
5	$(A - B)^2 * (2*A + B)$	13	$(A + B)*A - B*(2*A + 3*B)$
6	$(A - B)*A + 2*B^2$	14	$A*(2*A + B) - B*(A - B)$
7	$2*(A - 3*B) + A*B$	15	$3*(A+B)*(A*B - 2*A)$
8	$(A-B)*A + 3*B$		

### 2.5.2. Указания к выполнению задания

В задаче 1 для формирования матрицы, вектора и ввода значений используется функция  $matrix(L,N,f)$ , где  $L$  – число строк матрицы,  $N$  – число столбцов матрицы,  $f$  – функция  $f(l,n)$  при  $l = \overline{1,L}$ ,  $n = \overline{1,N}$ . При создании вектора - столбца количество столбцов матрицы принимается равным 1. Для определения минимального и максимального элемента матрицы  $M$  используются функции  $min(M)$  и  $max(M)$ . Среднее значение элементов матрицы  $M$  позволяет вычислить функция  $mean(M)$ . Для выполнения операций над векторами и матрицами используются операции, приведенные в таблице 2.30.

Таблица 2.30. Операции над векторами и матрицами Mathcad

Операция	Обозначение	Способ ввода	Описание
Изменение знака	$-A$	$-A$	Умножает каждый элемент массива $A$ на $-1$
Умножение массива на скаляр	$A*n$	$A*z$ или $n*A$	Умножает каждый элемент массива $A$ на скаляр $n$
Сложение массивов	$A1 + A2$	$A1 + A2$	Элементы массива $A1$ суммируются с соответствующими элементами $A2$
Матричное умножение	$A1*A2$	$A1*A2$	Возвращает произведение массива $A1$ на массив $A2$ . Число столбцов $A1$ должно быть равно числу строк $A2$
Деление массива на скаляр	$A/n$	$\frac{A}{n}$	Делит каждый элемент массива $A$ на скаляр $n$
Обращение матрицы $M$	$M^{-1}$	$M^{-1}$	Находится матрица, обратная заданной.
Возведение матрицы в степень $m$	$M^m$	$M^m$	$m$ раз перемножается матрица $M$ посредством матричного умножения.
Определитель матрицы $M$	$ M $	$ M $	Рассчитывается определитель квадратной матрицы $M$ , результат-скаляр
Транспонирование	$A^T$	$A^T$	Транспонирует массив $A$ , т.е. меняет местами строки со столбцами.
Выделение $p$ -го столбца матрицы	$M^{<math>p</math>}$	$M + CTRL \wedge p$	Возвращает вектор в виде $p$ – го столбца матрицы $M$
Выделение $ij$ – го	$M_{i,j}$	$M[i,j]$	Возвращает элемент матрицы $i$ -ой строки

Операция	Обозначение	Способ ввода	Описание
элемента матрицы			j-го столбца

В таблице приняты следующие обозначения:  $A$  – массив, под которым понимается вектор или матрица,  $M$  – матрица,  $p$  – скаляр,  $d$  – вектор.

Для создания новой матрицы  $C$  из элементов матриц  $A$  и  $B$  (задача 3) используются функции **submatrix()**, **augment()**, **stack()**. Выделить подматрицу из матрицы  $M$  можно посредством функции **submatrix( $M, r1, r2, c1, c2$ )**, где  $M$  – исходная матрица,  $r1$  и  $r2$  – нижний и верхний номер строки матрицы  $M$ , включаемых в результирующую подматрицу, а  $c1$  и  $c2$  – нижний и верхний номер столбца матрицы  $M$ , включаемых в результирующую подматрицу. Слияние матриц можно осуществить используя функции **augment( $A,B,...$ )** и **stack( $A,B,...$ )**. Функция **augment( $A,B,...$ )** предназначена для слияния матриц  $A, B$  и т.д. слева направо. Причем количество строк в матрицах должно быть одинаково. Функция **stack( $A,B,...$ )** выполняет слияние матриц сверху вниз. Количество столбцов в матрицах должно быть также одинаково. Данные функции могут быть применены и к векторам.

В задаче 4 для решения системы линейных алгебраических уравнений используются 3 метода. Первый метод - метод обратной матрицы, суть которого определяется выражением  $x=A^{-1}d$ , где  $x$  – вектор искомых неизвестных,  $A$  – матрица коэффициентов при неизвестных,  $d$  – вектор свободных членов. Второй метод - метод Крамера. Данный метод основан на вычислении определителей. Значения неизвестных определяются по формулам

$$x_1 = \frac{\Delta_1}{\Delta}, \quad x_2 = \frac{\Delta_2}{\Delta}, \dots, \quad x_n = \frac{\Delta_n}{\Delta}$$

где  $\Delta$  - определитель матрицы коэффициентов при неизвестных ( $A$ );  $\Delta_i$  – определитель добавочной матрицы, получаемой путем замены  $i$  – го столбца в матрице коэффициентов при неизвестных на вектор свободных членов:  $n$  – порядок системы линейных алгебраических уравнений, в нашем случае  $n=4$ .

В данном методе при формировании добавочной матрицы удобно воспользоваться функциями **submatrix()** и **augment()**. Третий метод позволяет определить вектор-столбец искомых неизвестных  $x$  на основе функции **Isolve( $A,d$ )**, где  $A$  – матрица коэффициентов при неизвестных, а  $d$  – вектор свободных членов.

### 2.5.3. Варианты заданий

В разделе 2.5.1. полностью определены все варианты, которые формируются аналитически на основе формул, приведенных в пункте 1. Параметр  $k$ , используемый в формулах, определяет номер варианта и позволяет для каждого варианта получить свой индивидуальный набор данных.

### 2.5.4. Примеры выполнения заданий

Примеры выполнения задач задания 2.5 приведены в виде листингов Mathcad.

Задача 1. Создать матрицы  $A$  и  $B$  размерностью  $4 \times 4$  и вектор столбец  $d$ , имеющий четыре элемента. Заполнить матрицы и вектор значениями, каждое из которых вычисляется по формуле, приведенной в 5.2.1. Ввести значение константы  $n$ . Вычислить минимальный и максимальный элемент матрицы и вектора. Определить среднее значе-

ние элементов матрицы и вектора.

Результаты решения задачи 1 приведены на листинге 2.29.

Листинг 2.29. Результаты решения задачи 1 (задание 2.5)

```

k := 16      Fa(x,y) :=  $\frac{[20 \cdot (x + 1)]}{\pi \cdot (y + 1)} \cdot \sin\left(\frac{k \cdot \sin(k \cdot x + 1)}{\cos(y \cdot k + 1) + 1}\right) \cdot \frac{\sqrt[k]{x + 1}}{\sqrt[k]{y + 1}}$ 

A := matrix(4,4,Fa)      A =  $\begin{pmatrix} 4.022 & -0.828 & 1.746 & -1.167 \\ 7.083 & -4.429 & -0.49 & 2.053 \\ -16.779 & -0.791 & -3.085 & -1.23 \\ 12.882 & -10.722 & -2.076 & 4.712 \end{pmatrix}$ 

Fb(x,y) :=  $\left(\frac{x + 1}{y + 1}\right)^{\sin(k+1)} \cdot \ln\left[\frac{[k + (x + 1) \cdot (y + 1)]}{k}\right] \cdot 10e^{\cos(k+2)}$ 

B := matrix(4,4,Fb)      B =  $\begin{pmatrix} 1.173 & 4.439 & 9.564 & 16.375 \\ 1.171 & 4.319 & 9.101 & 15.28 \\ 1.157 & 4.174 & 8.637 & 14.282 \\ 1.139 & 4.03 & 8.214 & 13.415 \end{pmatrix}$ 

Fd(x,y) :=  $\frac{\ln[(x + 1) + k] \cdot \cos[2 \cdot k \cdot (x + 1)]}{\log[(x + 1) + k]}$ 

d := matrix(4,1,Fd)      d =  $\begin{pmatrix} 1.921 \\ 0.902 \\ -0.415 \\ -1.595 \end{pmatrix}$       n := sin(k) * 10      n = -2.879

min(A) = -16.779      max(A) = 12.882      mean(A) = -0.569
min(B) = 1.139      max(B) = 16.375      mean(B) = 7.279
min(d) = -1.595      max(d) = 1.921      mean(d) = 0.203

```

Задача 2. Выполнить следующие операции над вектором и матрицами:  $A+B$ ;  $A-B$ ;  $A*B$ ;  $B*A$ ;  $A*d$ ;  $n*B$ ;  $n*d$ ;  $A^T$ ;  $B^2$ ;  $A^{-1}$ ;  $B^{-1}$ . Проверить справедливость тождеств  $A*A = A^2$ ;  $A^{-3} = A^{-1} * A^{-1} * A^{-1}$ . Вычислить выражение  $2*A + 3*B*(A*B - 2*A)$ .

Результаты решения задачи 2 приведены на листинге 2.30

Листинг 2.30. Результаты решения задачи 2 (задание 2.5)

$$\begin{array}{l}
 n = -2.88 \\
 A = \begin{pmatrix} 4.02 & -0.83 & 1.75 & -1.17 \\ 7.08 & -4.43 & -0.49 & 2.05 \\ -16.78 & -0.79 & -3.08 & -1.23 \\ 12.88 & -10.72 & -2.08 & 4.71 \end{pmatrix} \quad B = \begin{pmatrix} 1.17 & 4.44 & 9.56 & 16.37 \\ 1.17 & 4.32 & 9.1 & 15.28 \\ 1.16 & 4.17 & 8.64 & 14.28 \\ 1.14 & 4.03 & 8.21 & 13.42 \end{pmatrix} \quad d = \begin{pmatrix} 1.92 \\ 0.9 \\ -0.42 \\ -1.6 \end{pmatrix} \\
 A+B = \begin{pmatrix} 5.2 & 3.61 & 11.31 & 15.21 \\ 8.25 & -0.11 & 8.61 & 17.33 \\ -15.62 & 3.38 & 5.55 & 13.05 \\ 14.02 & -6.69 & 6.14 & 18.13 \end{pmatrix} \quad A-B = \begin{pmatrix} 2.85 & -5.27 & -7.82 & -17.54 \\ 5.91 & -8.75 & -9.59 & -13.23 \\ -17.94 & -4.96 & -11.72 & -15.51 \\ 11.74 & -14.75 & -10.29 & -8.7 \end{pmatrix} \\
 A*B = \begin{pmatrix} 4.44 & 16.86 & 36.43 & 62.49 \\ 4.9 & 18.55 & 40.07 & 68.87 \\ -25.58 & -95.73 & -204.42 & -347.4 \\ 5.53 & 21.2 & 46.38 & 80.66 \end{pmatrix} \quad B*A = \begin{pmatrix} 86.63 & -203.77 & -63.62 & 73.13 \\ 79.43 & -191.13 & -59.87 & 68.3 \\ 73.27 & -179.4 & -56.32 & 63.89 \\ 68.12 & -169.13 & -53.18 & 60.05 \end{pmatrix} \\
 A*d = \begin{pmatrix} 8.12 \\ 6.54 \\ -29.7 \\ 8.42 \end{pmatrix} \quad n*d = \begin{pmatrix} -5.53 \\ -2.6 \\ 1.2 \\ 4.59 \end{pmatrix} \quad A^T = \begin{pmatrix} 4.02 & 7.08 & -16.78 & 12.88 \\ -0.83 & -4.43 & -0.79 & -10.72 \\ 1.75 & -0.49 & -3.08 & -2.08 \\ -1.17 & 2.05 & -1.23 & 4.71 \end{pmatrix} \\
 B^2 = \begin{pmatrix} 36.29 & 130.29 & 268.73 & 443.3 \\ 34.36 & 123.42 & 254.63 & 420.14 \\ 32.5 & 116.77 & 240.96 & 397.66 \\ 30.84 & 110.81 & 228.71 & 377.51 \end{pmatrix} \quad A^{-1} = \begin{pmatrix} 0.22 & -6.21 & -0.64 & 2.6 \\ -0.09 & -3.51 & -0.43 & 1.4 \\ -0.73 & 26.46 & 2.51 & -11.06 \\ -1.12 & 20.65 & 1.86 & -8.58 \end{pmatrix}
 \end{array}$$

Продолжение листинга 2.30

$$A = \begin{pmatrix} 4.02 & -0.83 & 1.75 & -1.17 \\ 7.08 & -4.43 & -0.49 & 2.05 \\ -16.78 & -0.79 & -3.08 & -1.23 \\ 12.88 & -10.72 & -2.08 & 4.71 \end{pmatrix}$$

$$B = \begin{pmatrix} 1.17 & 4.44 & 9.56 & 16.37 \\ 1.17 & 4.32 & 9.1 & 15.28 \\ 1.16 & 4.17 & 8.64 & 14.28 \\ 1.14 & 4.03 & 8.21 & 13.42 \end{pmatrix}$$

$$B^{-3} = \begin{pmatrix} -100539574140.98 & 388166774071.27 & -489740535863.45 & 201952628211.41 \\ 102376291304.22 & -395254979511.63 & 498680372472.29 & -205638010174.98 \\ -59232450050.94 & 228683529025.55 & -288521066876.17 & 118975275505.13 \\ 14047905536.31 & -54235597271.98 & 68426628664.78 & -28216471487.59 \end{pmatrix}$$

$$A \cdot A = \begin{pmatrix} -34.01 & 11.47 & 4.47 & -14.04 \\ 31.79 & -7.88 & 11.78 & -7.09 \\ -37.18 & 33.02 & -16.84 & 15.96 \\ 71.4 & -12.05 & 24.36 & -12.3 \end{pmatrix}$$

$$A^2 = \begin{pmatrix} -34.01 & 11.47 & 4.47 & -14.04 \\ 31.79 & -7.88 & 11.78 & -7.09 \\ -37.18 & 33.02 & -16.84 & 15.96 \\ 71.4 & -12.05 & 24.36 & -12.3 \end{pmatrix}$$

$$A^{-3} = \begin{pmatrix} 16.52 & -518.52 & -52.49 & 211.42 \\ 8.71 & -273.39 & -27.68 & 111.47 \\ -72.41 & 2271.67 & 229.94 & -926.29 \\ -55.95 & 1755.42 & 177.68 & -715.78 \end{pmatrix}$$

$$A^{-1} \cdot A^{-1} \cdot A^{-1} = \begin{pmatrix} 16.52 & -518.52 & -52.49 & 211.42 \\ 8.71 & -273.39 & -27.68 & 111.47 \\ -72.41 & 2271.67 & 229.94 & -926.29 \\ -55.95 & 1755.42 & 177.68 & -715.78 \end{pmatrix}$$

$$2 \cdot A + 3 \cdot B \cdot (A \cdot B - 2 \cdot A) = \begin{pmatrix} -893.33 & -177.89 & -2539.43 & -5308.91 \\ -828.46 & -204.57 & -2449.96 & -5081.89 \\ -822.47 & -206.71 & -2349.84 & -4852.64 \\ -716.5 & -230.55 & -2246.65 & -4618.89 \end{pmatrix}$$

По частям P1 := 2 · A P2 := A · B P3 := P2 - P1 P4 := 3 · B · P3 REZ := P1 + P4

$$REZ = \begin{pmatrix} -893.33 & -177.89 & -2539.43 & -5308.91 \\ -828.46 & -204.57 & -2449.96 & -5081.89 \\ -822.47 & -206.71 & -2349.84 & -4852.64 \\ -716.5 & -230.55 & -2246.65 & -4618.89 \end{pmatrix}$$

Резюме Результаты, полученные в обоих случаях равны

Задача 3. Создать матрицу C из элементов матриц A и B, включив в нее:

- первый столбец матрицы A;
- второй столбец матрицы B;
- правую верхнюю четверть матрицы A;
- правую нижнюю четверть матрицы B.

Результаты решения задачи 3 приведены на листинге 2.31.

Листинг 2.31. Результаты решения задачи 3 (задание 2.5)

$$A = \begin{pmatrix} 4.02 & -0.83 & 1.75 & -1.17 \\ 7.08 & -4.43 & -0.49 & 2.05 \\ -16.78 & -0.79 & -3.08 & -1.23 \\ 12.88 & -10.72 & -2.08 & 4.71 \end{pmatrix} \quad B = \begin{pmatrix} 1.17 & 4.44 & 9.56 & 16.37 \\ 1.17 & 4.32 & 9.1 & 15.28 \\ 1.16 & 4.17 & 8.64 & 14.28 \\ 1.14 & 4.03 & 8.21 & 13.42 \end{pmatrix}$$

$c1 := \text{submatrix}(A, 1, 4, 1, 1)$        $c2 := \text{submatrix}(B, 1, 4, 2, 2)$   
 $c3 := \text{submatrix}(A, 1, 2, 3, 4)$        $c4 := \text{submatrix}(B, 3, 4, 3, 4)$   
 $c34 := \text{stack}(c3, c4)$        $C := \text{augment}(c1, c2, c34)$

$$C = \begin{pmatrix} 4.02 & 4.44 & 1.75 & -1.17 \\ 7.08 & 4.32 & -0.49 & 2.05 \\ -16.78 & 4.17 & 8.64 & 14.28 \\ 12.88 & 4.03 & 8.21 & 13.42 \end{pmatrix}$$

Задача 4. Решить систему линейных алгебраических уравнений вида  $C \cdot x = d$  методом обратной матрицы, методом Крамера и с использованием функции `lsolve()`.

Результаты решения задачи 4 приведены на листинге 2.32

### Контрольные вопросы

1. Какие операции над матрицей определяют следующие записи в Mathcad:  $|M|$ ;  $M^{-1}$ ;  $M^T$ ;  $M^{k \times n}$ , где  $M$  – матрица.
2. Какая функция предназначена для определения минимального, максимального элемента матрицы и подсчета среднего значения элементов матрицы?
3. Для чего предназначена функция **submatrix** ( $M, r1, r2, c1, c2$ )?
4. Какие функции осуществляют слияние матриц?
5. Что обозначают параметры  $A, b$  и  $x$  в матричной записи решения системы линейных алгебраических уравнений:  $x = A^{-1} \cdot b$ ?
6. Какую операцию выполняет функция **augment**( $A, B, \dots$ )?
7. Какую операцию выполняет функция **stack**( $A, B, \dots$ )?

Листинг 2.32. Результаты решения задачи 4 (задание 2.5)

Метод обратной матрицы

$$C = \begin{pmatrix} 4.02 & 4.44 & 1.75 & -1.17 \\ 7.08 & 4.32 & -0.49 & 2.05 \\ -16.78 & 4.17 & 8.64 & 14.28 \\ 12.88 & 4.03 & 8.21 & 13.42 \end{pmatrix} \quad d = \begin{pmatrix} 1.92 \\ 0.9 \\ -0.42 \\ -1.6 \end{pmatrix}$$

$$C^{-1} = \begin{pmatrix} -0 & 0 & -0.03 & 0.03 \\ 0.1 & 0.14 & 0.04 & -0.06 \\ 0.21 & -0.27 & -0.01 & 0.07 \\ -0.16 & 0.13 & 0.03 & 0.01 \end{pmatrix} \quad x := C^{-1} \cdot d \quad x = \begin{pmatrix} -0.04 \\ 0.39 \\ 0.05 \\ -0.23 \end{pmatrix}$$

Метод Крамера

$$\det := |C| \quad \det = 7123.12 \quad Cd1 := \text{augment}(d, \text{submatrix}(C, 1, 4, 2, 4))$$

$$\det1 := |Cd1| \quad \det1 = -311.67 \quad x1 := \frac{\det1}{\det} \quad x1 = -0.04$$

$$Cd2 := \text{augment}(\text{submatrix}(C, 1, 4, 1, 1), d, \text{submatrix}(C, 1, 4, 3, 4))$$

$$\det2 := |Cd2| \quad \det2 = 2803.34 \quad x2 := \frac{\det2}{\det} \quad x2 = 0.39$$

$$Cd3 := \text{augment}(\text{submatrix}(C, 1, 4, 1, 2), d, \text{submatrix}(C, 1, 4, 4, 4))$$

$$\det3 := |Cd3| \quad \det3 = 353.66 \quad x3 := \frac{\det3}{\det} \quad x3 = 0.05$$

$$Cd4 := \text{augment}(\text{submatrix}(C, 1, 4, 1, 3), d)$$

$$\det4 := |Cd4| \quad \det4 = -1606.55 \quad x4 := \frac{\det4}{\det} \quad x4 = -0.23$$

Функция Isolve()

$$y := \text{Isolve}(C, d) \quad y = \begin{pmatrix} -0.04 \\ 0.39 \\ 0.05 \\ -0.23 \end{pmatrix}$$

## 2.6. Решение дифференциальных уравнений

**Цель работы:** познакомиться с возможностями Mathcad для решения дифференциальных уравнений, изучить функции, предназначенные для этого, освоить методы решения краевых задач.

**Время:** 2 часа.

**Литература** Л1 с.137-141.

### 2.6.1. Задание 2.6

1. Решить дифференциальное уравнение вида  $y'=f(x, y)$  на отрезке  $[a,b]$  при начальных условиях  $y(a)=c$  с использованием блока **given... odesolve**. Результат проверить с использованием функции **rkfixed()**, результаты отобразить графически.

2. Решить дифференциальное уравнение 2 – го порядка с использованием функции **rkfixed()**. Результаты проверить с использованием **rkadapt()**. Отобразить график функции и ее первой производной. Сравнить значения функции и ее первой производной в конечной точке с точностью до 7 знаков после десятичной точки .

3. Решить краевую задачу вида

$$\begin{cases} y'' + p(x)y' + q(x)y = f(x) \\ y(a) = ya, \quad y(b) = yb \end{cases}$$

с использованием функции **odesolve()**, функции **rkfixed()** совместно с **sbsval()** и методом конечных разностей с шагом  $(b-a)/10$ .

### 2.6.2. Указания к выполнению задания

В задаче 1 и задаче 2 требуется найти решение задачи Коши при помощи различных функций.

**odesolve(x,b,step)**-используется для решения обыкновенного дифференциального уравнения, заданного как в виде задачи Коши, так и в виде краевой задачи. Начальные условия и дифференциальное уравнение должны быть определены в блоке **given...odesolve()**. Параметры функции:  $x$  –переменная интегрирования;  $b$  - конечное значение промежутка решения;  $step$  – величина шага численного метода (параметр необязательный).

**rkfixed(u,a,b,N,D)** – реализует численное решение задачи Коши по методу Рунге – Кутта с фиксированным шагом. Имеет следующие преимущества перед **odesolve(x,b,step)**: может быть использована в программных модулях и позволяет оперативно пересчитывать результаты при изменении параметров. Параметры функции:  $u$ -вектор начальных условий;  $a$   $u$   $b$  – граничные значения отрезка решения задачи;  $N$  – число интервалов разбиения отрезка  $[a,b]$ ;  $D(x,y)$  – вектор - функция, содержащая правые части первых производных, записанные в символьном виде.

**Rkadapt(u, x1, x2, n, D)** - возвращает матрицу, содержащую таблицу значений решения задачи Коши на интервале от  $x1$  до  $x2$  для уравнения или системы обыкновенных дифференциальных уравнений, вычисленную методом Рунге-Кутта с переменным шагом и начальными условиями в векторе  $u$ ,  $D(x,y)$  – вектор - функция, содержащая правые части первых производных, записанная в символьном виде,  $n$  - число шагов. Решение задач можно получить как в численном виде, так и отобразить графически.

Для решения краевой задачи (задача 3) предлагается так же использовать блок **given...odesolve()**. Причем последовательность решения аналогичная задаче 1. При использовании функции **rkfixed()** для решения краевой задачи необходимо свести краевую задачу к начальной, что достигается трансформацией граничных условий на конце отрезка в начальные условия в начале отрезка. Это осуществляется посредством функции **sbsval()**. Данная функция определяет недостающие условия в начальной точке для двухточечных краевых задач. Функция имеет следующий синтаксис: **sbsval(z,a,b,D,load,score)**, где  $z$  – вектор приближений недостающих начальных условий на левой границе;  $a,b$  – левая и правая граница интервала решений;  $D(x,y)$  – вектор - функция, содержащая правые части первых производных, записанная в символьном виде; **load(a,z)** – вектор - функция, описывающая начальные условия на левой границе интервала; **score(b,y)** – вектор функция для задания правых граничных условий.

При решении краевых задач методом конечных разностей применяются формулы для аппроксимации производных соответствующими конечно – разностными отноше-

ниями. Это позволяет свести решение дифференциальных уравнений к решению системы линейных уравнений. Результаты получают в дискретных  $i$ -ых точках интервала решения задачи. При этом отрезок  $[a, b]$  разбивается на  $n$  частей с шагом  $h = (b-a)/n$ . Для аппроксимации соответствующих производных используют следующие формулы

$$y'_i = \frac{y_{i+1} - y_{i-1}}{2 * h}$$

$$y''_i = \frac{y_{i+1} - 2 * y_i + y_{i-1}}{h^2}, i = 1, 2, \dots, n-1$$

Таким образом, сделав соответствующую замену, получаем систему линейных уравнений, решение которой средствами Mathcad может быть получено с использованием функции *Isolve()*.

### 2.6.3. Варианты заданий

Варианты задания 2.6 представлены в нижеприведенных таблицах.

Таблица 2.31. Индивидуальные варианты задания 2.6 (задача 1)

№	Функция $y'=f(x, y)$	a	b	y(a)	h
1	$y' = x * y^3 - x^2$	4	6	0.7	0.1
2	$y' = -\sqrt{4 * x^2 + 1} - 3 * y^2$	2.6	3.6	1.8	0.1
3	$y' = \cos(1.5 * x - y^3) - 1.3$	-1	1	0.2	0.1
4	$y' = x^2 + x * y + y^2$	2	5	1.2	0.1
5	$y' = e^{-(y^2+1)} + 2 * x$	0	1	0.3	0.05
6	$y' = \cos(1.5 * y + x)^2 + 1.4$	1	3	0.9	0.1
7	$y' = 4.1 * x - y^2 + 0.6$	0.6	2.6	3.4	0.1
8	$y' = \frac{1}{1+x^3*y} + 2 * y$	1.5	2.5	2.1	0.05
9	$y' = x + \cos \frac{y}{\sqrt{11}}$	2.1	4.1	2.5	0.1
10	$y' = \frac{2 * x * y}{x+4} - 0.4$	3	5	1.7	0.1
11	$y' = 2.5 * x + \cos(y + 0.6)$	1	3	1.5	0.1
12	$y' = x + 2.5 * y^2 + 2$	1	3	0.9	0.1
13	$y' = 2 - \sin(x + y)^2$	2	4	2.3	0.1
14	$y' = \frac{2}{x+2} + x + 1$	0.1	1	1.25	0.05
15	$y' = x + \cos \frac{y}{2}$	-2	0	3	0.1

Таблица 2.32. Индивидуальные варианты задания 2.6 (задача 2)

№	Функция $y''=f(x, y', y)$	a	b	y(a)	y'(a)
1	$y'' + y' / x + 2y = x$	0.7	2.2	0.5	1.2
2	$y'' - xy' + 2y = x + 1$	0.9	2.4	2	1.2
3	$y'' + xy' + y = x + 1$	0.8	2.3	1	1.2
4	$y'' + 2y' - y / x = 3y$	0.2	1.7	2	1

Продолжение таблицы 2.32

№	Функция $y''=f(x, y', y)$	a	b	y(a)	y'(a)
5	$y''+2y'-xy=x^2$	0.6	2.1	0.7	1
6	$y''+y'+2y/x=x+0.4$	1.4	2.9	2	4
7	$y''+3y'+y/x=1$	0.4	1.9	2	0.7
8	$y''+3y'-y/x=x+1$	1.2	2.7	0.5	1
9	$y''-y'/2+3y=2x^2$	1.3	2.8	1	0.6
10	$y''+1.5y'-xy=0.5$	1.6	3.1	3	1
11	$y''+y'-y/x=x^{0.5}$	1.6	3.1	3	1
12	$y''+x^2(y')^2-xy^2=x^3$	1	2.5	2	1
13	$(x+1)y''+xy'-y^2=(x-2)^3$	1	2.5	1.3	2
14	$(x+1)^2y''+y'-y=(x-2)^5$	1	2.5	1.3	2
15	$y''+x^*(y'-2)^2+2yx=(x+1)^2$	0.9	2.4	2	1.2

Таблица 2.33. Индивидуальные варианты задания 2.6 (задача 3)

№	$p(x)$	$q(x)$	$f(x)$	a	b	ya	yb
1	1.6	$3.6\cos(x-1)$	-10	1	2	0	8
2	0.7	4	$7-14x^2$	0	1	0	4
3	0.4	5	$-5e^{-x-1}$	1	2	0	-4
4	1	7	$x^2+3$	1	2	3	0
5	0.8	4	$1/(1+10x)$	0	1	8	4
6	0.6	12	$20x$	2	3	2	6
7	0.3	$0.6x$	$8\sin(x)$	1	3	2	2
8	$0.3x$	5	$3/(3-x)$	1	2	0	-2
9	$2x^2$	6	$15+15x$	0	1	7	2
10	0.2	$3x$	$\ln(1+x)$	1	2	1	6
11	$0.4x^2$	$5x$	10	0	1	-2	2
12	1.2	$5x^2$	$10+10\sin(x)$	0	1	-3	2
13	0.8	$8\cos(x)$	$(x-5)^3$	1	2	0	5
14	0.4	6	$1/(1+x)^2$	0	1	1	2
15	0.7	3	$0.5e^x$	3	4	3	0

### 2.6.4. Примеры выполнения заданий

Примеры задания 2.6 приведены в виде листингов Mathcad.

Задача 1. Решить дифференциальное уравнение вида  $y' = f(x)$  на отрезке  $[1.5; 2.5]$  при начальных условиях  $y(1.5) = 0.5$ . В качестве примера взять уравнение

$$\frac{d}{dx}y(x) = \sin(x + y(x)) + 1.5$$

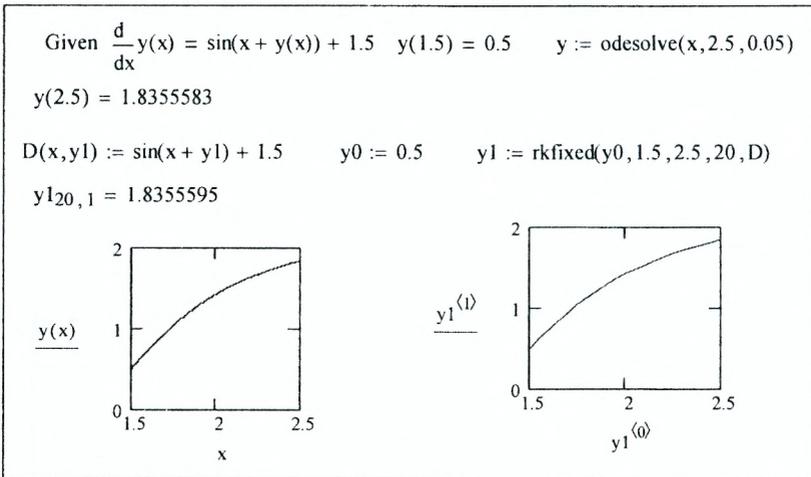
Решение выполнить с использованием блока `given...odesolve()`. Результат проверить

посредством функции rkfixed() и отобразить графически. Результаты решения задачи 1 приведены на листинге 2.33.

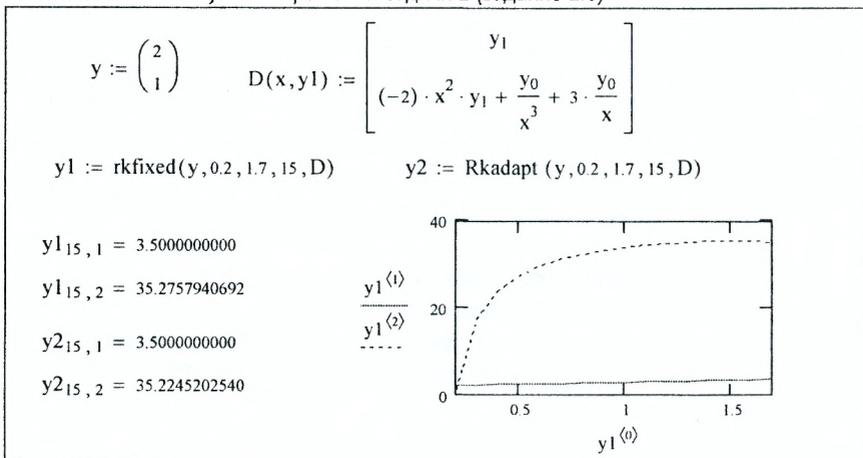
Задача 2. Решить дифференциальное уравнение второго порядка вида

$y'' + 2x^2 y' - \frac{y}{x^3} = \frac{3y}{x}$  при начальных условиях  $y(0,2)=2$   $y'(0,2)=1$  на отрезке  $[0,2; 1,7]$ . Для решения использовать функцию rkfixed(). Результат проверить посредством rkadapt() для  $y'$  и  $y$  в конечной точке. Результаты решения задачи 2 приведены на листинге 2.34.

Листинг 2.33. Результаты решения задачи 1 (задание 2.6)



Листинг 2.34. Результаты решения задачи 2 (задание 2.6)



Задача 3. Решить краевую задачу вида  $y''+0,6y'+2,6y=15\cos(x)$  на отрезке  $[0;1]$  при следующих краевых условиях  $y(0)=0$ ,  $y(1)=-2$  с использованием функции `odesolve()`, функции `rkfixed()` в сочетании с функцией `sbval()` и методом конечных разностей с шагом  $0,1$ . Результаты решения задачи 3 приведены на листинге 2.35

Листинг 2.35. Результаты решения задачи 3 (задание 2.6)

```

1 способ      Given  y(0) = 0  y(1) = -2

$$\frac{d^2}{dx^2}y(x) + 0.6 \cdot \frac{d}{dx}y(x) + 2.6 \cdot y(x) = 15 \cdot \cos(x)$$
      y := Odesolve ( x, 1, 0, 1 )

2 способ      yn := 0  yk := -2  t := 0, 0.1 .. 1
Зададим начальные приближения y'(0)
z0 := 1      load (x1, z) :=  $\begin{pmatrix} y_n \\ z_0 \end{pmatrix}$ 

$$D(x, y) := \begin{pmatrix} y_1 \\ -0.6 \cdot y_1 - 2.6 \cdot y_0 + 15 \cdot \cos(x) \end{pmatrix}$$

score (x2, y) := y0 - yk      z0 := sbval (z, 0, 1, D, load, score)      z0 = (-13.95 )
y4 :=  $\begin{pmatrix} 0 \\ -13.95 \end{pmatrix}$       u4 := rkfixed (y4, 0, 1, 10, D)

3 способ      dx := 0.1  i := 0 .. 10  xi := 0.1 * i  bi := 2 * dx^2 * 15 * cos (xi)  b0 := 0
b10 := -2  k1 := 2 - 0.6 * dx  k2 := -4 + 5.2 * dx^2  k3 := 2 + 0.6 * dx  k := 1 .. 9
M0,0 := 1  M10,10 := 1  Mk,k := k2  Mk,k-1 := k1  Mk,k+1 := k3

g := Isolve (M, b)



|    | 0      |
|----|--------|
| 0  | 0      |
| 1  | -1.279 |
| 2  | -2.307 |
| 3  | -3.073 |
| 4  | -3.579 |
| 5  | -3.83  |
| 6  | -3.842 |
| 7  | -3.637 |
| 8  | -3.24  |
| 9  | -2.683 |
| 10 | -2     |



| y(t) = |        |
|--------|--------|
|        | 0      |
| 1      | -1.275 |
| 2      | -2.299 |
| 3      | -3.063 |
| 4      | -3.567 |
| 5      | -3.819 |
| 6      | -3.832 |
| 7      | -3.628 |
| 8      | -3.233 |
| 9      | -2.679 |
| 10     | -2     |



|    | 0   | 1      | 2       |
|----|-----|--------|---------|
| 0  | 0   | 0      | -13.95  |
| 1  | 0.1 | -1.275 | -11.517 |
| 2  | 0.2 | -2.299 | -8.95   |
| 3  | 0.3 | -3.063 | -6.335  |
| 4  | 0.4 | -3.567 | -3.757  |
| 5  | 0.5 | -3.818 | -1.291  |
| 6  | 0.6 | -3.831 | 0.995   |
| 7  | 0.7 | -3.627 | 3.04    |
| 8  | 0.8 | -3.233 | 4.796   |
| 9  | 0.9 | -2.679 | 6.225   |
| 10 | 1   | -2     | 7.3     |


```

### Контрольные вопросы

1. Назначение функции `odesolve(x,b,step)`?
2. Назначение функции `rkfixed(u,a,b,N,D)`?
3. Назначение функции `Rkadapt(u, x1, x2, n, D)`?
4. Назначение параметров функции `sbval(z,a,b,D,load,score)`?
5. Назначение функций `load(a,z)` и `score(b,y)`?
6. Как записываются первая и вторая производные в методе конечных разностей?

## 2.7. Решение прикладных задач

Цель работы: познакомиться с возможностями Mathcad для решения задач обработки данных и решения задач линейной оптимизации.

Время: 2 часа.

Литература Л.1 с.142-151.

### 2.7.1. Задание 2.7

1. Решить задачу линейной оптимизации посредством блоков *given.. minimize()* и *given..maximize()*. Вычислить значение целевой функции.

2. Выполнить линейную и сплайн – интерполяцию данных и произвести расчет значений в точках  $x = 2$  и  $x = 1.55$ . При интерполяции использовать функции *linterp()*, *cspline()*, *pspline()*, *lspline()*, *interp()*. Результаты представить с точностью до 7 знаков после десятичной точки.

3. Для исходных данных, используемых в предыдущей задаче, разработать регрессионные модели с использованием функций *line()*, *loess()*, *regress()*, *expfit()*, *logfit()*. На основании коэффициента детерминации выбрать наилучшее регрессионное уравнение.

4. Выполнить на основе заданного набора данных экстраполяцию значений посредством функции *predict()*. Экстраполяцию следует провести в трех вариантах: используя 3 ближайших к правой границе точки; используя 5 ближайших к правой границе точки; используя 7 ближайших к правой границе точки. Результаты представить графически.

### 2.7.2. Указания к выполнению задания

Одним из способов решения задач линейной оптимизации (задача 1) в Mathcad является использование блока *given* с функциями *minimize()* и *maximize()*. При этом сначала задается вид целевой функции и вектор начальных приближений искомых переменных. Затем после ключевого слова *given*, описываются все ограничения задачи. Решение находится путем использования функций *minimize()* и *maximize()*.

В задаче 2 линейная интерполяция реализуется посредством функции *linterp(vx,vy,x)*, где *vx, vy* - векторы данных. Причём данные должны быть упорядочены по возрастанию; *x* - аргумент, для которого возвращается значение *y*. Сплайн-интерполяция осуществляется двумя функциями. Вначале вычисляется вектор вторых производных в рассматриваемых точках, затем вычисляется значение функции в точке  $x \rightarrow \text{interp}(vs,vx,vy,x)$ . Для построения вектора вторых производных в Mathcad имеется набор из 3-х функций, которые отличаются лишь граничными условиями: *cspline(vx,vy)* - генерирует кривую, являющуюся кубическим полиномом в граничных точках; *pspline(vx,vy)* - соответственно, параболу; *lspline(vx,vy)* - прямую.

В Mathcad существует набор функций, позволяющих рассчитать различные регрессионные модели. В задаче 3 предлагается использовать функцию *line(vx,vy)* – реализующую линейную регрессию; *regress(vx,vy,n)* – полином *n*-ой степени; *loess(vx,vy,span)* - позволяющую построить уравнение регрессии из фрагментов полиномов 2-ой степени (*span* – параметр, определяющий область, на которой строится очередной фрагмент полинома 2-ой степени); *expfit(vx,vy,g)*- экспоненциальная регрессионная модель (параметр *g* является вектором начальных приближений для неизвестных функций регрессии.); *logfit(vx,vy,g)*- логарифмическая регрессионная модель. Для выбора наилучшей регрессионной модели предлагается использовать коэффициент детерминации, который численно равен коэффициенту корреляции в квадрате. Значение коэффициента корреляции в Mathcad позволяет рассчитать функция *corr(A,B)*, где *A* и *B* – два вектора значений.

В задаче 4 для экстраполяции используется функция *predict(v,m,n)*. Здесь *v* - вектор эмпирических значений, *m* - количество ближайших к правой границе точек, на основе которых производится экстраполяция, *n* - количество точек, в которых производится экстраполяция данных. В задаче требуется визуально оценить по графику влияние параметра *m* на результаты экстраполяции.

### 2.7.3. Варианты заданий

Варианты задания 2.7 представлены в нижеприведенных таблицах.

Таблица 2.34. Индивидуальные варианты задания 2.7 (задача 1)

(все значения  $x_i \geq 0, i = 1, 2, 3$ )

№	Задание	№	Задание
1	$L(x) = x_1 - x_2 - 3x_3 \rightarrow \max$ $\begin{cases} 2x_1 - x_2 + x_3 \leq 1, \\ -4x_1 - 2x_2 - x_3 \leq 2, \\ 3x_1 + x_3 \leq 5. \end{cases}$	2	$L(x) = x_1 + x_2 \rightarrow \max$ $\begin{cases} 2x_1 + 11x_2 \leq 38, \\ x_1 + x_2 \leq 7, \\ 4x_1 - 5x_2 \leq 5. \end{cases}$
3	$L(x) = x_1 + 4x_2 \rightarrow \max$ $\begin{cases} -x_1 + 2x_2 \leq 2, \\ 3x_1 + 2x_2 \leq 6. \end{cases}$	4	$L(x) = 3x_1 + 4x_2 \rightarrow \max$ $\begin{cases} 3x_1 + 2x_2 \leq 8, \\ x_1 + 4x_2 \leq 10. \end{cases}$
5	$L(x) = x_1 + x_2 \rightarrow \max$ $\begin{cases} 2x_1 + x_2 \leq 7, \\ 2x_1 + 3x_2 \leq 10. \end{cases}$	6	$L(x) = x_1 - 4x_2 + 2x_3 \rightarrow \min$ $\begin{cases} 2x_1 + 2x_3 \leq 3, \\ -4x_1 - 3x_2 - x_3 \leq 3, \\ 3x_1 + x_2 - x_3 \leq 6. \end{cases}$
7	$L(x) = 2x_1 \rightarrow \max$ $\begin{cases} 13x_1 + 9x_2 \leq 38, \\ 2x_2 \leq 7, \\ -x_1 + 9x_2 \leq 5. \end{cases}$	8	$L(x) = 5x_1 - 3x_2 \rightarrow \max$ $\begin{cases} x_1 - 3x_2 \leq 2, \\ 5x_1 + x_2 \leq 6. \end{cases}$
9	$L(x) = 7x_1 - x_2 \rightarrow \min$ $\begin{cases} 5x_1 + x_2 \leq 4, \\ 5x_1 - 3x_2 \leq 6. \end{cases}$	10	$L(x) = 2x_1 + x_2 \rightarrow \max$ $\begin{cases} 3x_1 + x_2 \leq 6, \\ 5x_1 - x_2 \leq 9. \end{cases}$
11	$L(x) = 4x_1 + 5x_2 + x_3 \rightarrow \max$ $\begin{cases} 3x_1 + 2x_2 \leq 10, \\ x_1 + 4x_2 \leq 11, \\ 3x_1 + 3x_2 + x_3 \leq 13. \end{cases}$	12	$L(x) = 2x_1 + x_2 \rightarrow \max$ $\begin{cases} 13x_1 - 7x_2 \leq 34, \\ 2x_1 + x_2 \leq 3, \\ -x_1 + 13x_2 \leq 1. \end{cases}$
13	$L(x) = 5x_1 - 2x_2 \rightarrow \max$ $\begin{cases} x_1 - 4x_2 \leq 4, \\ 5x_1 + 4x_2 \leq 8. \end{cases}$	14	$L(x) = 7x_1 + 2x_2 \rightarrow \max$ $\begin{cases} 5x_1 - 4x_2 \geq 8, \\ 5x_1 - 2x_2 \leq 10. \end{cases}$
15	$L(x) = 2x_1 + x_2 \rightarrow \max$ $\begin{cases} 3x_1 + 3x_2 \leq 6, \\ 5x_1 + x_2 \leq 9. \end{cases}$		

Таблица 2.35. Индивидуальные варианты задания 2.7 (задача 2 и 3)

x	Значения y для каждого варианта								
	Вариант №								
	1	2	3	4	5	6	7	8	9
1.2	0.7788	1.284	0.2474	0.2526	1.031	0.2449	0.2553	1.271	0.235
1.3	0.7334	1.363	0.3051	0.3150	1.048	0.3004	0.3203	1.297	0.254
1.4	0.6977	1.433	0.3523	0.3678	1.066	0.3452	0.3764	1.310	0.263
1.6	0.6771	1.477	0.3802	0.4000	1.077	0.3714	0.4111	1.336	0.282
1.7	0.6505	1.537	0.4169	0.4434	1.094	0.4053	0.4586	1.350	0.291
1.9	0.6250	1.600	0.4529	0.4875	1.112	0.4382	0.5080	1.377	0.309
2.1	0.5945	1.682	0.4969	0.5438	1.138	0.4777	0.5726	1.448	0.354
2.2	0.5712	1.751	0.5312	0.5897	1.161	0.5080	0.6269	1.462	0.363
2.4	0.5273	1.896	0.5972	0.6846	1.212	0.5649	0.7445	1.522	0.397
2.6	0.5169	1.935	0.6131	0.7090	1.226	0.5784	0.7761	1.632	0.455
2.7	0.4916	2.034	0.6518	0.7712	1.263	0.6107	0.8595	1.804	0.533

Продолжение таблицы 2.35

Значения x	Значения y для каждого варианта					
	Вариант №					
	10	11	12	13	14	15
1.2	1.029	0.7866	0.2423	0.2447	0.2374	0.0792
1.3	1.034	0.7711	0.2629	0.2660	0.2571	0.1139
1.4	1.037	0.7634	0.2733	0.2768	0.2667	0.1461
1.6	1.042	0.7483	0.2941	0.2984	0.2860	0.2041
1.7	1.045	0.7408	0.3045	0.3093	0.2955	0.2304
1.9	1.052	0.7261	0.3255	0.3314	0.3146	0.2788
2.1	1.069	0.6907	0.3785	0.3879	0.3616	0.3222
2.2	1.073	0.6839	0.3892	0.3994	0.3709	0.3424
2.4	1.090	0.6570	0.4325	0.4466	0.4078	0.3802
2.6	1.122	0.6126	0.5098	0.5334	0.4706	0.4150
2.7	1.179	0.5543	0.6248	0.6696	0.5564	0.4314

Таблица 2.36. Индивидуальные варианты задания 2.7 (задача 4)

Значения x	Значения y для каждого варианта								
	Вариант №								
	1	2	3	4	5	6	7	8	9
1	13.50	251	0.91	1.89	6.3	105	0.32	1.80	95
2	12.77	249	0.87	1.91	6.21	108	0.34	1.78	97
3	12.05	248	0.85	2.00	7.15	112	0.37	1.70	96
4	11.93	246	0.82	2.05	7.0	117	0.51	1.64	99
5	11.55	259	0.79	2.19	6.8	140	0.65	1.59	93
6	12.26	262	0.75	2.27	5.75	132	0.79	1.51	91
7	12.88	255	0.7	2.03	5.05	126	0.52	1.45	95
8	12.75	240	0.86	2.17	4.85	128	0.48	1.42	90
9	11.62	238	0.62	2.29	5.5	150	0.31	1.40	91
10	10.51	225	0.6	2.35	5.2	134	0.54	1.37	96
11	9.98	220	0.58	2.42	4.34	128	0.67	1.67	89
12	10.05	223	0.55	2.54	4.53	140	0.71	1.55	97

Продолжение таблицы 2.36

Значения x	Значения y для каждого варианта					
	Вариант №					
	10	11	12	13	14	15
1	341	21.3	278	0.52	172	1.85
2	347	22.1	273	0.59	154	1.73
3	339	21.7	275	0.49	138	1.68
4	355	22.4	269	0.51	144	1.59
5	374	21.9	274	0.48	149	1.62
6	361	21.7	268	0.57	163	1.83
7	350	22.2	270	0.67	191	1.51
8	359	21.8	271	0.75	158	1.78
9	338	21.6	256	0.58	177	1.91
10	360	22.5	281	0.61	183	1.42
11	330	24.3	290	0.44	199	1.66
12	370	27.8	295	0.32	209	1.33

#### 2.7.4. Примеры выполнения заданий

Примеры выполнения задач задания 2.7 приведены в виде листингов Mathcad.

Задача 1. Решить задачу линейной оптимизации.

$$L(x) = -2 \cdot x_1 - x_2 - 4 \cdot x_3 \rightarrow \min$$

При ограничениях

$$x_1 - x_2 \leq 13$$

$$2 \cdot x_2 - 3 \cdot x_3 \geq 5$$

$$x_1 + 3 \cdot x_2 + 2 \cdot x_3 \leq 70$$

$$x_1 \geq 0; x_2 \geq 0; x_3 \geq 0$$

Решение задачи приведено на листинге 2.36.

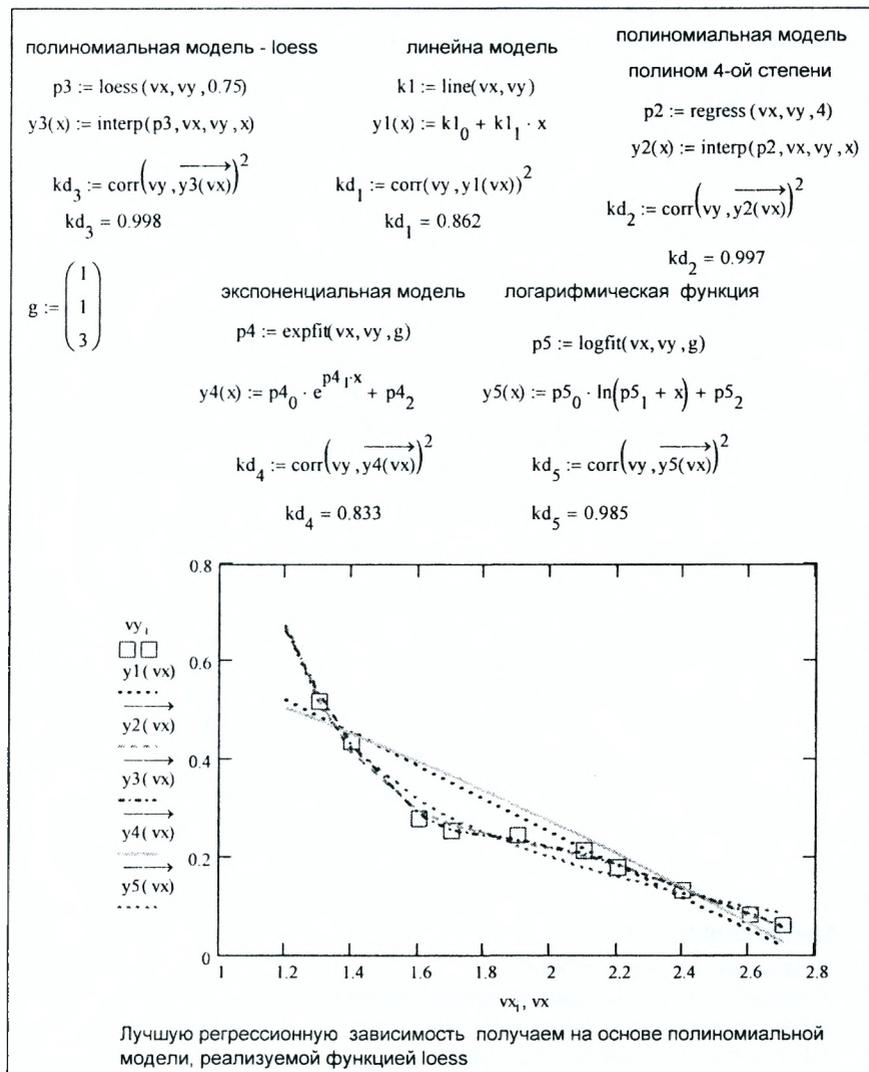
Листинг 2.36. Результаты решения задачи 1 (задание 2.7)

ORIGIN := 1	$F(x) := -2x_1 - x_2 - 4x_3$	$x := \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$
Given		
$x_1 \geq 0$	$x_2 \geq 0$	$x_3 \geq 0$
$x_1 - x_2 \leq 13$	$2 \cdot x_2 - 3 \cdot x_3 \geq 5$	
$x_1 + 3 \cdot x_2 + 2 \cdot x_3 \leq 70$	$x := \text{Minimize}(F, x)$	$x = \begin{pmatrix} 24.313 \\ 11.313 \\ 5.875 \end{pmatrix}$
$F(x) = -83.438$		



Задача 3. Для набора данных, приведенного в предыдущем примере, разработать различные регрессионные модели и на основании коэффициента детерминации выбрать наилучшее регрессионное уравнение. Результаты решения задачи 3 приведены на листинге 3.38

Листинг 2.38. Результаты решения задачи 3 (задание 2.7)

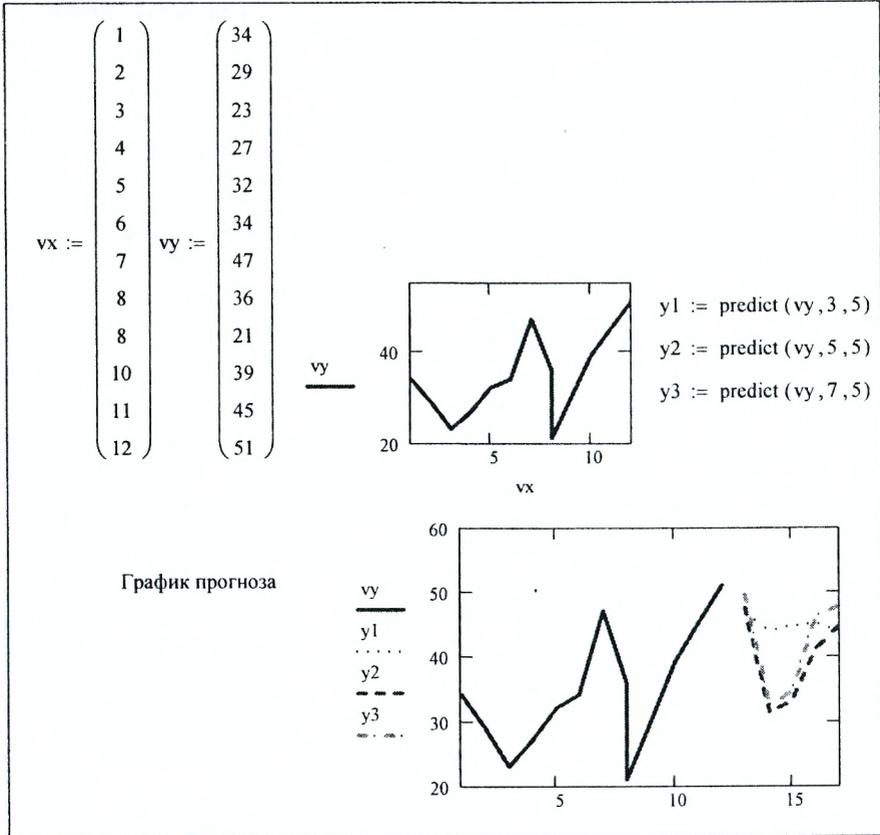


Задача 4. На основе заданного набора данных

x	1	2	3	4	5	6	7	8	8	10	11	12
y	34	29	23	27	32	34	47	36	21	39	45	51

выполнить экстраполяцию данных посредством функции `predict()`, используя: 3 ближайших к правой границе точки; 5 ближайших к правой границе точки; 7 ближайших к правой границе точки. Результаты представить графически на листинге 2.39.

Листинг 2.39. Результаты решения задачи 4 (задание 2.7)



### Контрольные вопросы

1. Назначение функции `linterp()`?
2. Назначение функций `cspline()`, `pspline()`, `lspline()`?
3. Назначение функций `loess()`, `regress()`?
4. Назначение функций `expfit()`, `logfit()`?
5. Какой показатель может быть использован для выбора наилучшего уравнения регрессии?
6. Назначение функции `predict()`?

## 3. АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

### 3.1. Базовые структуры схем алгоритмов

Цель работы: познакомиться с типовыми схемами алгоритмов и элементами схем алгоритмов, приобрести навыки в изображении схем алгоритмов базовых структур.

Время: 2 часа.

ЛИТЕРАТУРА: Л1. С. 172 – 176, Л2 С. 172 – 177.

#### 3.1.1. Задание

- Изучите теоретический материал, согласно рекомендованной литературе.
- Изобразите основные элементы схем алгоритмов: блок начала и окончания алгоритма, ввода-вывода информации, процесс, выбор.
- Изобразите схему алгоритма "следование".
- Изобразите схему алгоритма "решение": а) с одним вариантом решения; б) с двумя вариантами (альтернативами) решения.
- Изобразите схему алгоритма циклической структуры:
  - схему цикла с постусловием и заданным числом повторений (цикл типа "ДО") ;
  - схему цикла с предусловием и заданным числом повторений;
  - схему цикла с параметром (цикл типа "ПОКА") и с предусловием;
  - схему цикла с параметром и с постусловием.
- Изобразите схему алгоритма, включающую последовательность алгоритмов следования, решения и циклов согласно варианту задания. Пронумеруйте блоки.

#### Варианты заданий

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Начало	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Ввод/Вывод	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Следование	+					+					+	+		+	
Решение	+	+			+	+	+	+				+	+		+
Следование		+					+	+	+				+		
Цикл типа "ДО", с предусловием	+		+				+			+					
Цикл типа "ДО" с постусловием		+		+				+				+			
Цикл типа "ПОКА", с предусловием					+						+		+		+
Цикл типа "ПОКА" с постусловием						+			+					+	
Следование	+	+			+				+	+		+		+	+
Решение					+	+				+	+			+	+
Следование	+	+	+								+	+		+	
Конец программы	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

### 3.1.2. Пример выполнения задания по п. 6

**Пример 3.1.1.** Изобразите схему алгоритма включающую последовательность блоков: начало, ввод данных, процесс, решение с одним вариантом, цикл с предусловием, процесс, вывод результатов, конец программы. Пронумеруйте блоки.

**Решение.** См. рис.3.1.1.

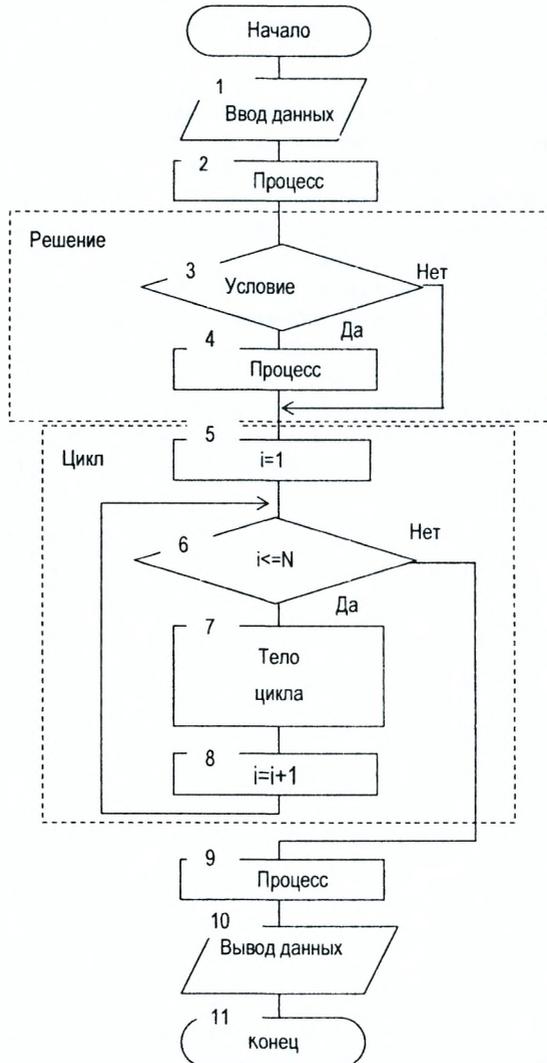


Рис.3.1.1. Пример схемы алгоритма

## Контрольные вопросы

1. Перечислите основные элементы схем алгоритмов.
2. Что представляет собой алгоритм следования?
3. Что представляет собой алгоритм решения?
4. Какие виды алгоритмов решения Вам известны?
5. Какие виды циклов Вам известны?
6. Чем отличаются циклы с постусловием от циклов с предусловием?
7. Чем отличаются циклы типа "ДО" от циклов типа "ПОКА"?
8. Какие элементы схем алгоритмов включает схема цикла?
9. Каким образом осуществляется нумерация блоков в схемах алгоритмов?
10. Какие элементы схем алгоритмов используются для указания перехода между блоками: а) в пределах одного листа; б) между листами.

## 3.2. Изучение среды разработки проекта

Цель работы: приобрести начальные навыки работы в среде Visual Basic. Изучить элементы среды программирования, порядок установки элементов на форму и управления размещением элементов.

Время – 2 часа.

Литература: Л1 с. 219 - 231, Л3 с. 8-18

### 3.2.1. Общие сведения

Рабочее окно (рис. 3.2.1) представляет собой интегрированную среду разработки – интер-

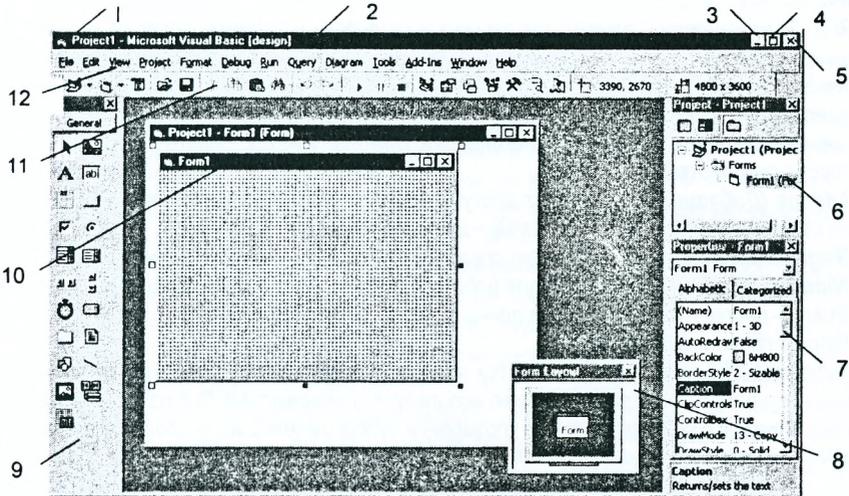


Рис. 3.2.1. Окно программы Visual Basic

- 1 – кнопка системного меню, 2 – заголовок, 3 – кнопка свертывания окна в значок,
- 4 – кнопка развертывания окна, 5 – кнопка закрытия окна, 6 – окно проекта, 7 – окно свойств,
- 8 – окно позиционирования формы, 9 – панель элементов управления, 10 – форма,
- 11 – стандартная панель инструментов, 12 – меню

фейс языка программирования Visual Basic. Эта среда может настраиваться с помощью диалогового окна, вызываемого командой **Tools, Options**. Рабочее окно предлагает целый набор инструментальных средств, которые можно использовать при создании программ.

Большинство элементов рабочего окна, характерны для приложений Windows: заголовков, меню, панели инструментов. Интегрированная среда разработки проекта многооконная. Состав окон приведен на рис. 3.2.1.

### **Меню (Menu)**

Строка Меню содержит список команд, предназначенных для управления разработкой проекта, пункты меню могут иметь несколько уровней вложения:

**File (Файл)** - содержит команды для работы с файлами создаваемых приложений, загрузки, сохранения, вывода на печать.

**Edit (Правка)** - содержит команды редактирования, предназначенные для создания исходного текста программы, включая средства поиска и замены.

**View (Просмотр)** - обеспечивает доступ к различным частям приложения и среды разработки VB.

**Project (Проект)** - предназначен для добавления новых объектов VB к разрабатываемым проектам, добавления или удаления элементов управления на панель элементов управления, настройки свойств проекта.

**Format (Формат)** – содержит команды управления размещением элементов управления на форме.

**Debug (Отладка)** – содержит средства, предназначенные для отладки программ или поиска ошибок.

**Run (Выполнение)** – служит для запуска и остановки программ непосредственно из среды разработки.

**Tools (Инструменты)** – обеспечивает доступ к работе с процедурами и меню внутри приложения. Этот пункт меню имеет важную команду **Options**, которая открывает одноименную диалоговую панель с закладками **Options**, где настраивается практически вся среда разработки Visual Basic.

**Add-Ins (Добавить в ...)** – дает доступ к инструментам, которые могут быть добавлены к окружению VB: мастера, ActiveX – элементы и другое.

**Diagram (Диаграммы)** – содержит средства для оформления диаграмм.

**Window (Окно)** – используется для работы с окнами в среде разработки.

**Query** – доступ к внешним базам данных.

**Help** – справочная система.

Выбор пунктов меню осуществляется мышью или клавишами. При управлении с помощью клавиатуры для входа в меню используется клавиша **Alt**. Выбор пунктов меню осуществляется с помощью клавиш управления курсором или с использованием горячих клавиш (подчеркнутые символы в командах меню): [**Alt- клавиша**].

### **Панели инструментов (Toolbars)**

VB имеет четыре стандартные панели инструментов: *Standard* – стандартная, *Edit* – редактирования, *Debug* – отладки и *Form Editor* – редактор форм. Стандартная панель инструментов выводится на экран по умолчанию и содержит команды, дублирующие основные команды меню.

## Панель инструментов элементов управления и компонентов пользователя (Toolbox)

Панель Toolbox содержит элементы управления. *Элементы управления* – это объекты, которые используются при разработке интерфейса создаваемых приложений. Общее количество доступных к использованию элементов управления зависит от того, какая версия VB используется.

Для добавления новых компонентов к панели инструментов Toolbox из числа зарегистрированных необходимо:

- ввести команду *Project, Components*, выбрать закладку *Controls*;
- найти в списке нужный элемент управления и установить напротив него флажок;
- выйти из окна диалога, щелкнув кнопку *Ok*.

## Форма (Form)

Создаваемые в Visual Basic окна называются *формами*. *Форма* – это основное окно интерфейса разрабатываемой программы, форма – это также основа для создания окон диалога.

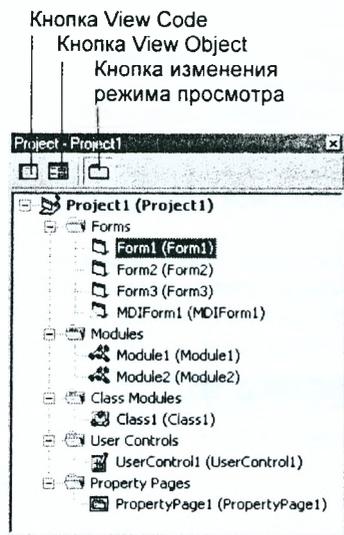


Рис. 3.2.2. Окно Проект

в виде списка компонентов или в виде дерева групп компонентов. В качестве объектов могут быть формы, MDI-формы, модули, классы, управляющие элементы, страницы свойств. Модули и классы не имеют визуальных компонентов.

## Окно Свойств (Properties)

В окне свойств (рис. 3.2.3.) задаются свойства выбранного элемента управления. В строке заголовка окна свойств рядом с текстом *Properties* указывается

## Окно Проект (Project)

В окне проекта (броузер проектов) (рис. 3.2.2.) отображаются все элементы приложения: формы, модули, классы и т.п., сгруппированные по категориям. В VB все разрабатываемые приложения называются *проектами*. Проект представляет группу связанных файлов и может содержать несколько групп компонентов (формы, модули и т.д.). Все проекты VB строятся по модульному принципу, поэтому и текст программы состоит не из одного большого файла, а из нескольких частей – процедур. Несколько проектов также могут объединяться в группы. Ниже заголовка окна проекта размещены три кнопки: кнопка просмотра текста программы; кнопка просмотра объекта; кнопка изменения режима просмотра: в виде

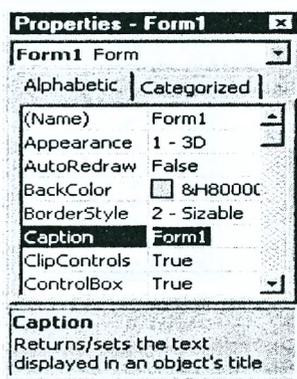


Рис. 3.2.3. Окно Свойства

имя формы, которой принадлежит элемент управления. Поле со списком под строкой заголовка позволяет выбрать требуемый элемент управления. В списке, расположенном ниже, перечислены свойства этого элемента. Эти свойства могут быть упорядочены в алфавитном порядке (Alphabetic) или расположены по категориям (Categorized). Набор свойств зависит от типа элемента управления.

Свойства объекта вводятся тремя способами:

- ввести значение в поле справа от свойства;
- выбрать из списка, который открывается щелчком мыши по полю;
- установить с помощью окна диалога. При щелчке мышью по полю появляется кнопка (...) – трюеточие (или эллипсис). При щелчке по кнопке трюеточие появляется окно диалога для настройки соответствующего свойства.

### Окно Программы (Code)

Окно Программы открывается двойным щелчком мыши по форме или щелчком по кнопке *View Code* в окне *Project*.

В верхней строке окна программы (рис.3.2.4) располагается строка заголовка, в которой указано имя проекта и управляющие кнопки (системного меню, закрытия окна, свертывания и развертывания окна). Ниже строки заголовка расположены два раскрывающихся списка: список объектов (левый) и список процедур (правый).

**Список объектов** содержит все объекты текущей формы. В их число входит и специальный объект *General*, содержащий общий код, используемый всеми процедурами формы.

**Список процедур** содержит список всех событий, распознаваемых текущим объектом. Если для объекта уже написаны процедуры обработки событий, то они выделяются здесь жирным шрифтом. Если выбрать любой пункт данного списка, то VB выведет соответствующую процедуру либо ее шаблон в окне программы и поместит в нее курсор.

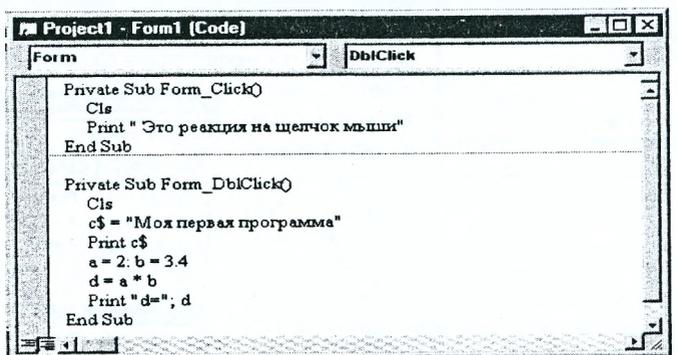


Рис. 3.2.4. Окно Программы (Code)

### Окно позиционирования формы (Form Layout)

Данное окно позволяет визуально установить место, где будет размещаться форма при запуске программы.

### 3.2.2. Задание

1. Изучите среду разработки проекта.
2. Изучите приемы установки элементов управления на форму и управления размещением элементов управления на форме.

#### Порядок выполнения задания

1. Запустите программу Visual Basic: введите команду **Пуск, Программы**, найдите в подменю команды Программы значок Visual Basic 6.0 и щелкните по нему дважды мышью. Если данного значка нет в меню, найдите его в компьютере командой **Пуск, Найти, Файлы и папки**. Имя исполняемого файла – Vb6.exe (могут быть и другие способы запуска Vb6 в зависимости от опыта пользователя и установленного программного обеспечения).

2. Щелкните в окне диалога *New Project* дважды по пиктограмме *Standard.EXE* для создания стандартного приложения Windows (это окно может не появляться на экране, а сразу появится рабочее окно Visual Basic).

3. Изучите элементы рабочего окна VB (среда разработки программы) и команды меню. Измените размеры и положение формы в окне, перетаскивая ее мышью.

4. Закройте окна Проект и Свойства кнопками закрытия окон.

5. Откройте окна Проект и Свойства командами **View, Project Explorer** и **View, Properties Window**, соответственно. Найдите в стандартной панели инструментов одноименные кнопки.

6. Переместите окно Проект в нижний правый угол рабочего окна. Переместите окно Проект в первоначальное положение.

7. Закройте панель элементов управления (Toolbox). Выведите панель Toolbox в рабочее окно командой **View, Toolbox**.

8. Откройте новую форму. Присвойте имя, наименование, установите размеры формы и ее положение в проекте.

– Откройте окно *Form Layout*, если его нет на экране.

– Запустите программу командой **Run, Start**. Появится пустая форма.

– Закройте проект. Установите положение формы в проекте с помощью окна *Form Layout*. Снова запустите проект и проверьте положение формы в окне проекта.

9. Установите на форму пять меток и пять окон ввода (Label, TextBox). Опишите в таблице их свойства и присвойте значения этих свойств элементам управления по форме Таблицы 3.2.1.

Таблица 3.2.1 Форма для описания свойств объектов

Тип объекта	Имя	Наименование	Положение верхнего левого угла		Размеры	
			Top	Left	Ширина	Высота
Label						
TextBox						

Можно также добавить такие свойства, как цвет, шрифт и т.п.

10. Упорядочите размещение элементов управления на форме с помощью опций команды главного меню **Format** (см. Приложение 1).

11. Установите на форму рамку. Скопируйте в нее несколько элементов управления с формы, используя контекстное меню и выровняйте их по вертикали.

12. Добавьте на панель элементов управления сетку *Microsoft Flex Grid*. Введите команду **Project, Components, Controls**, найдите в списке объект *Microsoft FlexGrid Control 6.0 (SP3)* и установите щелчком мыши напротив него флажок. Щелкните по кнопке **Ok**.

13. Добавьте к проекту еще одну форму командой **Project, Add Form**.

14. Создайте новую папку на рабочем диске и сохраните проект в этой папке командой **File, Save As**.

15. Откройте новый проект командой **File, New Project**.

16. Откройте сохраненный ранее проект командой **File, Open Project**.

17. Выведите на печать форму **Form1**. Введите команду **File, Print**, установите флажок **Form Image** и щелкните по кнопке **Ok** – будет напечатана форма со всеми установленными на ней элементами (или пустая форма).

18. Выйдите из программы *Visual Basic*.

#### **Указание к выполнению задания**

Выполните последовательно все пункты задания. По ходу выполнения задания оформите отчет, отражая в нем порядок выполнения задания, вводимые команды и результаты выполнения команд. Вводимые команды описывайте, например, следующим образом: *Файл, Сохранить*, указать диск, маршрут и имя файла.

#### **Контрольные вопросы**

1. Как запустить программу VB?
2. Назовите основные элементы рабочего окна VB.
3. Как установить элементы управления на форму?
4. Какая команда используется для управления размещением элементов управления на форме?
5. Как добавить новые элементы управления на панель инструментов *Toolbox*?
6. Для чего предназначено окно *Проект*?
7. Каково назначение окна *Свойства*?
8. Для чего предназначено окно *Программа*?
9. Как вызвать окно *Программа*?
10. Где записывается текст программы объекта?
11. Как сохранить программу на диске?
12. Как загрузить существующую программу?
13. Как вывести на печать текст программы?
14. Как вывести на печать сведения об установленных свойствах формы?

### **3.3. Разработка простой формы**

**Цель занятия:** закрепить знания, полученные на первом занятии по языку программирования *Visual Basic*.

**Время** – 2 часа.

**Литература:** Л1 с.240 - 249, Л3 с. 29-42

#### **3.3.1. Задание**

Создайте форму для вычисления среднего балла успеваемости студента. Число предметов обучения - четыре. На форме разместите пять окон для ввода/вывода данных (*TextBox*), 6 надписей (*Label*) и две кнопки (*CommandButton*).

1. Разработайте проект вычисления среднего балла успеваемости студента согласно примеру рис.3.3.1.
2. Разработайте форму для вычисления площади поверхности и объема фигуры по заданию преподавателя.

### Порядок выполнения работы

1. Запустите программу Visual Basic 6.
2. Создайте новый проект командой File, New Project.
3. Присвойте форме имя и заголовок (Name, Caption) в окне Свойств (Properties).

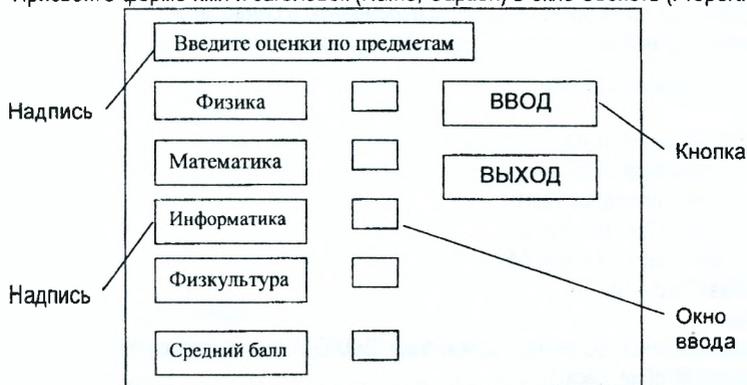


Рис. 3.3.1. Пример размещения элементов на форме

4. Разместите на форме надписи (*Label*) и для каждой из них определите свойства:
  - имя (Name);
  - текст, выводимый в окно метки (*Caption*);
  - выравнивание текста – по центру (*Alignment, Center*).
5. Разместите на форме окна для ввода оценок, используя элемент управления *TextBox*. Для каждого окна определите свойства:
  - имя (Name);
  - начальное значение поля – пусто (в свойстве *Text*, удалите информацию);
6. Разместите командные кнопки, используя элемент управления *CommandButton*. Для каждой кнопки определите свойства:
  - имя (Name);
  - наименование (*Caption*);
 Остальные свойства объектов принимаются по умолчанию.
7. Откройте окно программы (*View Code*) и введите текст программы:
  - а) Запишите текст программы в обработчик события *Click* для кнопки "ВВОД" (имя кнопки *cmdVvod*):
    - откройте список объектов и выберите объект *cmdVvod*. Откройте список свойств и методов и выберите свойство *Click*. Введите текст программы, например, `fiz = Val(txtFizika.Text)`

Здесь fiz – имя переменной, Val – имя функции преобразования символьной переменной в числовую переменную, txtFizika.Text – значение свойства Text поля ввода txtFizika - оценки по физике .

Аналогично вводятся оценки и по другим предметам обучения. Затем введите формулу для вычисления среднего балла, например,

$$Sb = (Fiz + Mat + Inf + Fizk) / 4$$

Значение среднего балла присвойте свойству Text окна "Средний балл" (имя окна ввода txtSrBall):

```
txtSrBall.Text = Str(Sb)
```

Функция Str преобразует числовую переменную в символьную.

#### Пример программы:

```
Private Sub cmdVvod_Click()  
    Cls  
    fiz = Val(txtFizika.Text)  
    mat = Val(txtMatematika.Text)  
    inf = Val(txtInformatika.Text)  
    fizk = Val(txtFizkultura.Text)  
    Sb = (fiz + mat + inf + fizk) / 4  
    txtSrBall.Text = Str(Sb)
```

```
End Sub
```

б) Напишите текст программы для кнопки "ВЫХОД", имя кнопки cmdExit:

```
Private Sub cmdExit_Click()  
    End  
End Sub
```

Сохраните проект на диске R: в отдельной папке, например, ЛабРабота2. Введите команду Save As ..., укажите имя файла, например, Успеваемость и щелкните по кнопке Сохранить.

Запустите программу для отладки: Run, Start.

После отладки сохраните проект.

Теперь можно приступить к красочному оформлению проекта. Для этой цели можно использовать такие свойства объектов, как BackColor – цвет фона, ForeColor - цвет символов в форме или цвет символов объектов TextBox и Label, Font – шрифт.

Сохраните оформленный проект на диске.

Выведите текст проекта на печать: введите команду File, Print, установите флажки Current Project и Code и нажмите клавишу OK.

#### Контрольные вопросы

1. Перечислите основные свойства формы.
2. Перечислите основные свойства метки (Label).
3. Перечислите основные свойства окна ввода/вывода (TextBox).
4. Перечислите основные свойства и события кнопки (CommandButton).
5. Приведите пример присвоения значения свойства объекта переменной.
6. Приведите пример присвоения значения переменной свойству объекта.

## Справка

Параллелепипед:  $S=2(ab+bc+ca)$ ;  $V=abc$ .  $S$  – площадь;  $V$  – объем,  $a$ ,  $b$ ,  $c$  – стороны параллелепипеда.

Призма:  $S= M+2F$ ;  $V=Fh$ .  $M$  – площадь боковой поверхности,  $F$  – площадь основания,  $h$  – высота.

Пирамида:  $S= pb/2+F$ ;  $V=Fh/3$ ,  $p$  – периметр,  $b$  – высота боковой грани (апофема).

Круговой прямой цилиндр:  $S=2\pi R(R+h)$ ,  $V= \pi R^2h$ .

Конус:  $S=\pi R(R+l)$ ,  $V= \pi R^2h/3$ , где  $l$  – образующая конуса,  $l= \sqrt{R^2 + h^2}$ .

Шар:  $S=4\pi R^2$ ,  $V= \pi R^3/3$ .

## 3.4. Разработка линейных программ

**Цель работы:** познакомиться со средой программирования и приобрести навыки в записи выражений по правилам языка Visual Basic, анализа области определения функций. Познакомиться со способами ввода/вывода информации (операторы Let, Print, функции Tab(N), Spc(N), операторы CurrentX, CurrentY, метод Print объекта Debug, функции InputBox и MsgBox).

**Время** – 4 часа.

**Литература:** Л1 с. 177 – 182, 243 - 249, Л2 с. 136 – 143, с. 190 – 203, Л3. 20 – 42.

### 3.4.1. Общие сведения

Выражения представляют собой комбинацию переменных и операндов, соединенных знаками отношения. В зависимости от используемых знаков операций различают арифметические выражения, логические выражения и строковые выражения.

Синтаксис всех выражений одинаков:

<операнд> знак\_операции <операнд>

В качестве операндов в выражениях могут использоваться константы, переменные, функции и другие выражения. Математические функции, используемые в языке Visual Basic, приведены в табл. 3.4.1.

Кроме этого, необходимо помнить всегда об одной из особенностей работы с формой и другими объектами в среде VB6: числовые данные вводятся в программу и выводятся в объекты и на форму в символьном виде. Поэтому при вводе данные должны переводиться из символьной формы в числовую с помощью функции Val(C), а при выводе – переводиться из числовой формы в символьную с помощью функции Str(N) или Str\$(N).

Таблица 3.4.1 Математические функции

Функция	Описание
Abs(N)	Вычисляет абсолютное значение числа N
Atn(N) <sup>3</sup>	Вычисляет арктангенс числа N. Значение аргумента может находиться в интервале от $-\infty$ до $+\infty$
Cos(N) <sup>4</sup>	Вычисляет косинус аргумента
Exp(N)	Возвращает число e, возведенное в указанную степень

<sup>3</sup> N – переменная числового типа, C – переменная символьного типа.

<sup>4</sup> В тригонометрических функциях аргумент измеряется в радианах. Для пересчета радианов в градусы надо умножить значение аргумента на  $180/\pi$ . Для пересчета градусов в радианы следует умножить аргумент на  $\pi/180$ . Точное значение  $\pi$  вычисляется по формуле  $\pi=4 \cdot \arctg(1)$ .

Таблица 3.4.1 Математические функции

Функция	Описание
Fix(N)	Возвращает целое число, меньшее или равное N для положительных чисел и большее или равное N для отрицательных чисел
Int(N)	Возвращает целое число, меньшее или равное N как для положительных, так и для отрицательных чисел
IIf(усл., V1,V2)	Условная функция, возвращает результат согласно выражению V1, если условие истинно и согласно выражению V2, если условие ложно
Log(N)	Вычисляет натуральный логарифм аргумента
Rnd(N)	Возвращает случайное число в интервале от 0 до 1. При N<0 возвращает определенное число, зависящее от N; при N=0 – псевдослучайное число; при N>0 – новое случайное число
Round(N[, дес])	Возвращает число, округленное к заданному числу десятичных знаков
Sgn(N)	Возвращает знак числа: 1, если N>0; 0, если N=0; -1, если N<0
Sin(N)	Вычисляет синус угла
Sqr(N)	Вычисляет корень квадратный из аргумента
Tan(N)	Вычисляет тангенс угла
Val(C)	Преобразует аргумент символьного типа в число

В языке программирования Visual Basic нет логарифма по основанию 10, а также нет обратных тригонометрических функций. Поэтому полезно запомнить формулы, по которым можно вычислять логарифм по произвольному основанию и обратные тригонометрические функции:

$$\text{Log}_a(x) = \frac{\text{Log}_b(x)}{\text{Log}_b(a)} \quad (3.4.1)$$

$$\arcsin(x) = \arctg(x / \sqrt{1 - x^2}) \quad (3.4.2)$$

$$\arccos(x) = \pi / 2 - \arctg(x / \sqrt{1 - x^2}) \quad (3.4.3)$$

$$\arctg(x) = \pi / 2 - \arctg(x) \quad (3.4.4)$$

**Ввод данных** осуществляется с помощью оператора *Let*.

Let <имя переменной> = <выражение>

Let a=2

Let b1=Sin(x)+exp(x)

Оператор LET может быть опущен, поэтому выражения

Let A= Sqr(x)+ Log(x) и A= Sqr(x)+ Log(x)

эквивалентны.

**Вывод данных** осуществляется с помощью метода Print.

Оператор Print выводит данные на форму. Синтаксис метода Print:

Print "Комментарий" [; / /,] <список выражений> [; /,]

Комментарии делают понятным пользователю, что выводится на экран методом Print. В качестве выражений могут использоваться константы, переменные, функции и выражения:

Print x, y

Print "x=";x, "y=";y

Print "Значение функции Sin(x) при x = ";x;" равно ";Sin(x)

При использовании в качестве разделителя символа ";" значения выводятся без пробела, при использовании в качестве разделителя символа "," значения выводятся в разных зонах.

В методе Print могут использоваться функции Tab(N) и Spc(N). Здесь N – число зна-комест:

```
Print "x=";x; Tab(20); "y=";y
```

Результат:

1.754      3,564354

```
Print "x=";x; Spc(20); "y=";y
```

Результат:

1.754                      3,564354

При использовании функции Tab отсчет символов ведется от края формы, а при использовании функции Spc – от текущего положения курсора.

Если в конце метода Print стоят символы разделители, то курсор ввода остается в текущей строке и следующий метод Print будет выводить информацию с этой позиции. Если символы-разделители в конце метода Print отсутствуют, то курсор переводится на следующую строку.

Позиционирование точки вставки можно осуществлять также с помощью операторов CurrentX, CurrentY. Оператор CurrentX задает расстояние от края формы до точки вставки по горизонтали, а CurrentY – по вертикали. Расстояния задаются в твипах. Например: CurrentX=500; CurrentY=200.

Результаты вычислений можно вывести также в окно Immediate оператором Debug.Print. Здесь Debug – объект для отладки программы, Print – метод данного объекта. Например: Debug.Print "Объем куба = ",V

Функция InputBox позволяет вводить данные в режиме диалога. Синтаксис функции:

<переменная>=InputBox("Сообщение", "Заголовок окна", Состав кнопок, X, Y, Файл справки, Подсказка)

Здесь X и Y - координаты верхнего левого угла окна диалога. Например:

```
sA= InputBox("Введите Фамилию", "Список студентов", ,1500,500).
```

```
iA = Val(InputBox("Введите значение аргумента", "Вычисление функции"))
```

В первом примере переменная символьная, во втором примере переменная числовая.

Функция MsgBox используется в двух форматах: как функция и как команда. При использовании в качестве функции аргументы заключаются в скобки, при использовании в качестве команды аргументы не заключаются в скобки. Команда MsgBox предназначена для вывода сообщений и результатов вычислений в окно диалога.

Синтаксис команды:

```
MsgBox "Сообщение", <Состав кнопок>, "Заголовок окна" Файл справки, Подсказка
```

### 3.4.2. Задание

1. Запишите выражения по правилам языка Visual Basic. Данные вводите с помощью оператора Let.

2. Определите и запишите область определения функции.

3. Исследуйте функцию на отрезке [0,1]. Напишите текст программы и вычислите значение функции при произвольном значении аргумента из области задания функции. Результаты выведите на форму с использованием функций Tab, Spc или операторов CurrentX, CurrentY.

4. Изобразите схему алгоритма.

**Примечание:** если области определения функции в целом не существует, проанализируйте отдельно числители и знаменатели и решите задачу по частям. Укажите для каждой части область определения функции.

5. Повторите задание по п. 1-3. Данные вводите с помощью окна диалога InputBox, а результаты выведите на экран с помощью команды MsgBox.

**Варианты заданий**

1	$y = \frac{\arccos(qx - 0,6) \cdot \sqrt{\sin(x + e^{x^2}) + h}}{\sqrt{x^2 + 2\operatorname{ctg}(x)} \cdot \ln\left(\frac{x^2 + 2,34x}{\sin(x^2)}\right)} - \frac{e^{\frac{x \cdot \operatorname{tg}^2(x)}{x^2 - 4,7}} - \left  2,87 \cdot x + \frac{q}{\sqrt[3]{x^2 + 1}} \right }{\log_2\left(x^3 - \sqrt[3]{x^2}\right) + \cos^2\left(x^3 - 4,56x - 1\right)}$ <p><math>q = -512,38 \cdot 10^{-3}</math> ; <math>h = 2,941 \cdot 10^5</math></p>
2	$y = \frac{\left  x^2 - \operatorname{ctg}^2(x + 4) \right  - \sqrt[3]{x^2 + 3,29x - q}}{\sin\left(x^2 - 4,51\sqrt{x}\right) \cdot \ln\left(\frac{x \cdot \operatorname{tg}^2(x)}{x^2 + h}\right)} + \frac{\arcsin(x - q) \cdot e^{x^4 + \operatorname{ctg}(2x)}}{\log_4(x + 7,84) + \sqrt{\cos^2(x - 4) + \sqrt[3]{x}}}$ <p><math>q = -0,258 \cdot 10^{-3}</math> ; <math>h = 51,334 \cdot 10^5</math></p>
3	$y = \frac{\log_2\left(x^3 + \sqrt[3]{x^2}\right) + \cos^2\left(x^3 - 4,56x - h\right)}{e^{\frac{x \cdot \operatorname{tg}(2x)}{x - 4,7}} - \left  2,87x + \frac{q}{\sqrt[3]{x^2 - 1}} \right } + \frac{\sqrt{x^2 + 2\operatorname{ctg}(x)} \ln\left(\frac{x^2 - 0,24x}{\sin(x^2)}\right)}{\arccos(q \cdot x - 0,4) \cdot \sqrt{\sin\left(x - e^{x^2}\right) + h}}$ <p><math>q = -512,38 \cdot 10^5</math> ; <math>h = 0,558 \cdot 10^5</math></p>
4	$y = \frac{\sqrt[3]{x^2 + q} - \arccos(2x - 1)}{\ln\left(\frac{x - \sin^2(x)}{\sqrt[3]{x - h}}\right) \cos\left(x^2 - \operatorname{tg}(x)\right)} + \frac{\sqrt{\sin(x - 2) + x^3} \cdot \log_2\left(x^3 + 4,28x^2 - q\right)}{e^{\frac{1}{h \cdot \operatorname{ctg}(x - 1)}} + \operatorname{tg}(x^2) + q}$ <p><math>q = 41,548 \cdot 10^4</math> ; <math>h = -52,333 \cdot 10^6</math></p>
5	$y = \frac{\log_{\frac{1}{3}}\left(q + \frac{x}{2 - x}\right) \cdot \sin^2\left(x^2 - 4,53x + h\right)}{\sqrt{\cos(x^2) + e^{x - 5}} \cdot \ln\left(\frac{\operatorname{ctg}(x)}{\sqrt[3]{x^2 + 1}}\right)} - \frac{\operatorname{arccotg}^2(x - 5) + \left  x^3 - 2^{x - 4} + q \right }{\operatorname{tg}\left(\sqrt[3]{x^2 \cdot h}\right) - \sqrt{x^3 + 7,84 \cdot x}}$ <p><math>q = -325,57 \cdot 10^2</math> ; <math>h = 6,278 \cdot 10^4</math></p>

6	$y = \frac{\log \frac{1}{4}(x+7,84) + \sqrt{\cos^2(x)} + \sqrt[3]{x}}{\sin(x^2 - 4,51\sqrt{x}) \cdot \ln\left(\frac{x \cdot \operatorname{tg}^2(x)}{x^2 \cdot h}\right)} + \frac{\arcsin(x-0,4) \cdot e^{x^4 + \operatorname{ctg}(x)}}{ x^2 - \operatorname{ctg}^2(x+4)  - \sqrt{x^3 + 3,29x + q}}$ <p><math>q = -512,38 \cdot 10^{-5}</math>; <math>h = 0,558 \cdot 10^5</math></p>
7	$y = \frac{\ln\left(\frac{\operatorname{ctg}(x+1)}{\sqrt[3]{x^2 + 4,75}} + 2\right) - \left(x^5 + \sin^2(x) - \frac{2x-3,8}{\sqrt{1,7x^3 + 2}}\right) \cdot \operatorname{tg}(x)}{\cos^2(x-h) \cdot \log \frac{1}{4}(x+5,5)} - \frac{e^{2x^3} + \sin(x)}{\arcsin^2(x^2 + q)}$ <p><math>q = 1,782 \cdot 10^{-3}</math>; <math>h = -25,782 \cdot 10^{-5}</math></p>
8	$y = \frac{ x^2 - \operatorname{ctg}^2(x+h)  - \sqrt[3]{x^2 - 3,29x - q}}{\log \frac{1}{4}(x+7,84) + \sqrt{\cos^2(x-4)} + \sqrt[3]{x}} + \frac{\sin(x^2 - 4,51\sqrt{x}) \cdot \ln\left(\frac{x \cdot \operatorname{tg}^2(x)}{x^2 + h}\right)}{\arcsin(x-0,3) \cdot e^{x^4 + \operatorname{ctg}(2x)}}$ <p><math>q = 7,385 \cdot 10^2</math>; <math>h = 1,78495 \cdot 10^{-8}</math></p>
9	$z = \frac{r \log \frac{3}{7} \frac{x}{y} + \sqrt{\frac{x}{y}} + 2,3}{\operatorname{se}^{-xy} + \sin^5(x^2y)} - \frac{\left(\sqrt[5]{x^2y^2} - e^{-x^2y^2}\right) \cdot 3^{-x} + r 2^{-x^2y^2}}{s \cdot 3^{-x} + r \sqrt[5]{\frac{x}{y}} + 2,3 \cdot \left(\arcsin^3(xy) - x^2y^2\right)}$ <p><math>r = -1,7 \cdot 10^5</math>; <math>s = 2,8 \cdot 10^{-6}</math></p>
10	$z = \frac{2s e^{-x^2} + \arcsin^3\left(\frac{xy}{2,7}\right)}{\operatorname{rctg}^3 \frac{x}{y} - 4\sqrt{1 + \frac{xy}{2,7}}} - \left[ \frac{\arcsin(xy/5) - e^{-2xy}}{\arccos(xy/5) + 2^{-xy}} \sqrt{\frac{xy}{2,7}} \right]$ <p><math>r = 1,2 \cdot 10^4</math>; <math>s = -3,2 \cdot 10^{-5}</math></p>
11	$z = \frac{s \cdot 2^{-xy} + \arccos^3\left(\frac{xy}{2}\right) + r}{\arcsin^2\left(\frac{xy}{2}\right) - \sqrt[5]{(2xy)^3}} - \frac{\operatorname{se}^{-xy} + \sqrt[4]{2,7xy}}{r \left( \frac{\log_5^2(x/y) - \sqrt[7]{x/y}}{\arccos(x/3,7) - e^{-x/y}} \right)}$ <p><math>r = -4,5 \cdot 10^{-8}</math>; <math>s = 1,2 \cdot 10^{-5}</math></p>
12	$z = \frac{\sqrt[5]{1 + rx^3} - \log_5^2( x+2y )}{2,3(1 + rx^3/2y)} - \frac{\log_5^2(x/y) - \sqrt[7]{x/y} - e^{(x,y)}}{re^{-(x,y)} + s \cdot 2^{xy} + \arccos(x/3,7) - e^{-(x/y)}}$ <p><math>r = 1,87 \cdot 10^{-4}</math>; <math>s = 23,4 \cdot 10^5</math></p>

13	$y = \frac{\log_{1/3}(x^2 + 3,764) + \sqrt{\cos(x-h)^3} + \sqrt[3]{x}}{\sin(x^2 - 4,51\sqrt{x}) \cdot \ln\left(\frac{x \cdot \operatorname{tg}^2(x)}{x^2 \cdot h}\right)} + \frac{\arccos(x-h) \cdot e^{x^4 + \operatorname{ctg}(x)}}{ x^2 - \operatorname{ctg}(x+4)  - \sqrt{x^3 + 3,29x + q}}$
	$h = 1,23 \cdot 10^{-3}; \quad q = 14,58 \cdot 10^{-5}$
14	$y = \frac{ x^2 - \operatorname{tg}^3(x+h)  - \sqrt{x^2 - 3,29x - q}}{\lg(x+3,84) + \sqrt{\cos^3(x-4) + \sqrt[3]{x}}} + \frac{\cos(x^2 - 4,51\sqrt{x}) \cdot \log_{0,5}\left(\frac{x \cdot \operatorname{tg}^2(x)}{x^2 + h}\right)}{\operatorname{arctg}\left(x - \frac{4}{x}\right) \cdot e^{x^4 + \operatorname{ctg}(2x)}}$
	$h = 3,76 \cdot 10^{-4}; \quad q = 24,17 \cdot 10^{-5}$
15	$y = \frac{\log_{0,3}\left(\frac{\operatorname{tg}(x-1)}{\sqrt[3]{x^2 + 4,75}} + 2\right) - \left(x^5 + \sin^2(x) - \frac{2x - 3,8}{\sqrt[5]{3,7x^3 + 2}}\right) \cdot \operatorname{tg}(x)}{\cos^2(x-h) \cdot \log_{1/4}(x+1,4)} - \frac{e^{2x^3} + \sin^3(x^5)}{\arccos^2(x^2)}$
	$h = 1,28 \cdot 10^{-3}; \quad q = 57,34 \cdot 10^{-5}$

### 3.4.3. Пример выполнения задания

**Пример 3.4.1.** Вычислить выражение:

$$y = \frac{e^{2x+3} + \lg(3x^2 + 1)}{\sqrt{5\sin(x)} - \operatorname{arctg}(x^2)} + \arcsin(x) \quad (3.4.1)$$

#### Решение

1. Запись выражения по правилам языка Visual Basic:

$$y = (\exp(2*x+3) + \log(3*x^2+1)/\log(10))/(\sqrt{5*\sin(x)} - \operatorname{atn}(x^2)) + \operatorname{atn}(x/\sqrt{1-x^2})$$

2. Определение области допустимых значений аргумента:

Функция  $e^{2x+3}$  не накладывает никаких ограничений на значение аргумента.

У функции  $\lg(3x^2+1)$  значение аргумента должно быть больше 0, то есть,  $3x^2+1 > 0$ . Так как  $x^2$  будет положительным при любом значении  $x$ , то областью определения  $x$  будет вся числовая ось.

Для функции  $\sqrt{5\sin(x)}$ : функция  $\sin(x)$  – периодическая, ее значение должно быть больше или равно нулю, следовательно значение  $x$  должно быть расположено в интервалах  $[k\pi, (k+1)\pi]$ , где  $k$  четные члены натурального ряда и 0.

У функции  $\operatorname{arctg}(x^2)$  нет ограничения на значение аргумента, функция принимает значение 0 при  $x = k2\pi$ , где  $k2$  – любое целое число или 0. Но знаменатель в выражении (3.1.1) не должен быть равен нулю. Следовательно,  $x$  может принимать значения из интервалов  $k\pi < x < (k+1)\pi$ .

Для функции  $\arcsin(x)$  аргумент должен находиться в интервале  $-1 \leq x \leq 1$ .

Окончательно получаем: функция определена на интервале  $0 < x \leq 1$

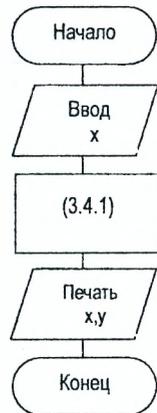


Рис.3.4.1.  
К примеру  
3.4.1

3. Текст программы:

```
REM Линейная программа
Private Sub Form_Click()
x=0.5
y=(EXP(2*x+3)+LOG(3*x^2+1)/LOG(10))/(SQR(5*sin(x))-
ATN(x^2))+ATN(x/sqr(1-x^2))
PRINT "x="x, "y="y
Debug.Print "x=";x;Spc(20);"y=";y
End Sub
Результат: x=0.5 y=42.60272
```

```
Private Sub Form_Click()
x=Val(InputBox("Введите значение x ", "Вычисление функций"))
y=(EXP(2*x+3)+LOG(3*x^2+1)/LOG(10))/(SQR(5*sin(x))-
ATN(x^2))+ATN(x/sqr(1-x^2))
MsgBox "x=" & Str(x) & VbCr & "y=" & Str(Round(y,2))
End Sub
Результат: x=0.5 y=42.60272
```

### Контрольные вопросы

1. Что такое линейная программа? Изобразите схему алгоритма линейной программы.
2. Напишите арифметические функции языка Visual Basic.
3. Напишите тригонометрические и обратные тригонометрические функции языка Visual Basic.
4. Как вычисляется логарифм функции по произвольному основанию?
5. Как вычисляются обратные тригонометрические функции  $\arccos(x)$ ,  $\arcsin(x)$ ,  $\text{arcctg}(x)$ ?
6. Какие ограничения на значение аргумента накладываются при использовании встроенных функций языка Visual Basic?
7. Приведите формат оператора Let и дайте пояснение.
8. Приведите формат метода Print и дайте пояснение.
9. Назовите функции и методы, используемые для управления выводом информации на форму.
10. Изобразите схему алгоритма программы и поясните принцип работы программы по этой схеме.

## 3.5. Программы с ветвлениями

**Цель занятия:** изучить операторы IF/THEN, IF/THEN/ELSE, IF/END IF. SELECT CASE.  
Приобрести практические навыки в разработке программ с ветвлениями.

**Время:** 2 часа

**Литература:** Л1 с.182 – 187, Л2 с. 178 – 182, 203 - 204

### 3.5.1. Общие сведения

#### Условный оператор

Различают однострочный и многострочный условные операторы If:

а) однострочный оператор

If <условие> Then <операторы> [Else <операторы>]

При выполнении оператора If проверяется условие и, если оно истинно, то выполняется действие, указанное после опции Then. Если выражение ложно, то выполняется действие, указанное после опции Else. Затем управление передается на оператор, следующий за оператором If.

После опций Then и Else может быть указано несколько операторов, разделенных двоеточием. Однако число операторов ограничено длиной строки.

б) многострочный оператор If:

а) простой

```
If <условие> Then
    <первая группа операторов>
[Else
    <вторая группа операторов>]
End If
```

б) расширенный

```
If <условие> Then
    <первая группа операторов>
[Elseif <условие> Then
    <вторая группа операторов> ]
[Else
    <третья группа операторов> ]
End If
```

При записи операторов следует обращать внимание на структуру записи, структура должна соответствовать той, что указана в примере.

Достоинство многострочного оператора If состоит в том, что число операторов в группах не ограничено.

### Оператор выбора

Оператор Select Case является оператором выбора. Он обеспечивает переход на ту или иную подпрограмму в зависимости от результатов вычислений. Так же как и многострочный If, оператор Select Case позволяет сформировать хорошо читаемую программу. Очень удобен для обработки пунктов меню.

**Пример** использования оператора Select Case.

```
Select Case nkl      ' nkl - переключающее выражение
Case 1              ' условие выборки
    <блок операторов 1>
Case 2
    <блок операторов 2>
...
Case Else
MsgBox "Неправильный ввод. Введите пункт меню"
Beep ' Выдача звукового сигнала
End Select
```

В опции Case могут использоваться следующие формы:

- число, или список значений, например: Case 1, 3, 5, 8;
- выражение [, выражение]. *Выражение* – любое числовое или строковое выражение, которое должно совпадать с типом переключающего выражения;
- выражение To выражение, например ( Case 1 To 10);
- Is выражение. Ключевое слово Is представляет собой условие, например (Case Is <5).

### 3.5.2. Задание

1. Вычислите значение разветвляющейся функции согласно варианту задания. Определите область допустимых значений аргумента и выберите значения коэффициентов  $a$  и  $b$  таким образом, чтобы границы разрыва функции находились в этой области. Значения переменных  $a$  и  $b$ , а также независимой переменной вводите с клавиатуры.

2. Изобразите условия задачи графически согласно варианту задания (см. пример 3.5.1).

3. Составьте схему алгоритма решения задачи.

4. Решите задачу двумя способами:

– с помощью простого однострочного оператора IF;

– с помощью многострочного оператора IF.

Программа должна сопровождаться текстовыми сообщениями, поясняющими получаемые результаты.

5. Решите задачу с помощью оператора Select Case.

6. Сделайте распечатку одного из вариантов решения задачи.

#### Указания к решению задачи:

– установите на форму три кнопки. Каждую кнопку используйте для запуска одного из вариантов решения задачи;

– введите значения переменных  $a$  и  $b$ ;

– вычислите значения границ изменения функций, например  $d$  и  $c$  и выведите их на форму;

– определите область определения функции на каждом из отрезков;

– введите значение независимой переменной  $x$  на первом отрезке и запишите результат;

– повторите вычисления при значениях аргументов на других отрезках области определения функции;

– для удобства анализа результаты вычислений целесообразно выводить в окно Непосредственного наблюдения (Immediate);

– для очистки формы перед выводом информации, включите в программу оператор Cls.

#### Варианты заданий

1	$y = \begin{cases} 5x^3 + 2\ln( 2x ), & x < \sqrt{a-b} \\ \sqrt{3 + \sin(2x)} - e^{3 x }, & x \geq \sqrt{a-b} \end{cases}$	2	$y = \begin{cases} \ln(x^2) - e^x + \lg(3x), & x < 2a + b \\ \arctg(2x - 0.5) + \sqrt{4x + 1}, & x \geq 2a + b \end{cases}$
3	$y = \begin{cases} e^{\sin(x)} + \sqrt{2 \cos(6x - 0.3)}, & x < b^2 \\ e^{-x} + \sqrt{\lg(3x + 0.6)}, & x \geq b^2 \end{cases}$	4	$y = \begin{cases} 2^x - e^x + x^3 \cos(4x - 0.2), & x <  a^2 - b^2  \\ \lg(4.5x) + \frac{x}{\sin(0.5x)}, & x \geq  a^2 - b^2  \end{cases}$
5	$y = \begin{cases} 2\sqrt{ \sin(x)  + \cos^2(x)} + e^{bx}, & x < \sqrt{b^3 + 6} \\ (x^{-3} - 5)\cos(3x - 0.5), & x > \sqrt{b^3 + 6} \end{cases}$	6	$y = \begin{cases} \ln\left \frac{x^{-1}}{3x + 4}\right  - e^{2x}, & x < \sqrt{a^2 + b} \\ x^3 + 2ax - \sqrt{x^2} -  \lg(x) , & x \geq \sqrt{a^2 + b} \end{cases}$

7	$y = \begin{cases} \ln(x^2) - e^{\frac{x}{2}} + \lg(x+1), & x < 2a-b \\ \arctg(2x+0.5) + \sqrt{5+3x}, & x > 2a-b \end{cases}$	8	$y = \begin{cases} e^{2x} + \sqrt{ x^4 + 3x^3 }, & x < b-a^2 \\ \frac{\sqrt{ x^4 + 4x^2 }}{\lg(2+x)}, & x \geq b-a^2 \end{cases}$
9	$y = \begin{cases} \frac{xe^3}{ 3+2x } + \sin^2(x), & x < ba/2 \\ (3x^3 + 6x^2 + 105)\sqrt{\sin(2*x)}, & x \geq ba/2 \end{cases}$	10	$y = \begin{cases} \frac{3x^3 + \sqrt{x^2 + 7 x }}{2\cos(2x)}, & x \leq 2ab \\ \lg\left(\frac{5x^3}{2x+1}\right), & x > 2ab \end{cases}$
11	$y = \begin{cases} x^3 - 2\ln 2x  + \sin^2(x), & x < \sqrt{b-a} \\ \sqrt{ 3 + \lg(x) - e^{3x} }, & x \geq \sqrt{b-a} \end{cases}$	12	$y = \begin{cases} \frac{2}{3}x - x^2 \sin\left(\frac{x}{2} - 0.6\right), & x < a^2 + b^2 \\ 2\cos(x^2 - 1) + \sqrt[3]{x^2}, & x \geq a^2 + b^2 \end{cases}$
13	$y = \begin{cases} \arctg(5x - 0.7), & x \geq a + 2b \\ \ln(x + 1.5) + \sin(5x), & x < a + 2b \end{cases}$	14	$y = \begin{cases} (x^2 - 1)\cos(x) - x\sqrt{2}, & x < a + b \\ x^{-2} - \sqrt{ \lg(2x-1) }, & x \geq a + b \end{cases}$
15	$y = \begin{cases} \arcsin(2x^2 - 0.8), & x < a^2 + 2b \\ \lg(x+2) + \sin(2x^2), & x > a^2 + 2b \end{cases}$	16	$y = \begin{cases} (x^3 - 2)\sin(x^3) - x\sqrt{3x}, & x \geq b - a \\ x^{-2} - \sqrt{ \text{ctg}(2x-1) }, & x < b - a \end{cases}$

### 3.5.3. Пример выполнения задания

**Пример 3.5.1.** Вычислить разветвляющуюся функцию

$$y = \begin{cases} 2x^2 - 4x + 5, & x < a-b & (1) \\ 10\sin(x^2) + 5\cos(x), & a-b \leq x \leq a+b & (2) \\ e^x - 17, & x > a+b & (3) \end{cases}$$

при  $a=1, b=2$

**Решение**

1. Функция определена на всех отрезках. Группа событий полная, следовательно, для решения задачи достаточно сделать две проверки.

2. Условия решения задачи представим графически (рис. 3.5.1).

3. Составим схему алгоритма решения задачи (рис. 3.5.2).

4. Текст программы

Option Explicit

Dim a As Single, b As Single, c As Single, d As Single

Dim x As Single, y As Single

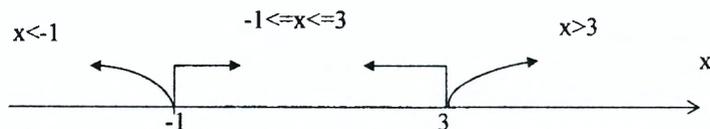


Рис. 3.5.1. Условия решения задачи 3.5.1., представленные графически

```

Private Sub cmdIfPrima_Click()
Rem Разветвляющаяся программа
' решение с использованием однострочного If
Cls
a = Val(InputBox("Введите переменную a "))
b = Val(InputBox("Введите переменную b "))
c = a - b: d = a + b
Print "C="; c, "D="; d
x = Val(InputBox("Введите переменную x "))

```

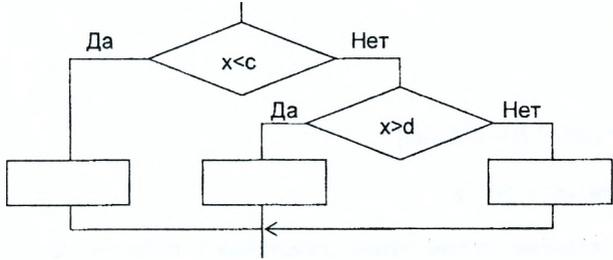


Рис.3.5.2. Схема алгоритма к примеру 3.5.1

```

If x < a Then y = 2 * x ^ 2 - 4 * x + 5: GoTo M1
If x > b Then y = Exp(x) - 17: GoTo M1
y = 10 * Sin(x ^ 2) + 5 * Cos(x)
M1:
Debug.Print "x="; x, "y="; y
End Sub

```

```

Private Sub cmdIfSecond_Click()
Rem Разветвляющаяся программа
' решение с использованием многострочного If
Cls
a = Val(InputBox("Введите переменную a "))
b = Val(InputBox("Введите переменную b "))
c = a - b: d = a + b
Print "C="; c, "D="; d
x = Val(InputBox("Введите переменную x "))
If x < a Then
    y = 2 * x ^ 2 - 4 * x + 5
Elseif x >= a And x <= b Then
    y = 10 * Sin(x ^ 2) + 5 * Cos(x)
Else
    y = Exp(x) - 17
End If
Debug.Print "x="; x, "y="; y
End Sub

```

```

Private Sub cmdSelectCase_Click()

```

```

Rem Разветвляющаяся программа
' решение с использованием оператора Select Case
Cls
a = Val(InputBox("Введите переменную a "))
b = Val(InputBox("Введите переменную b "))
c = a - b: d = a + b
Print "C="; c, "D="; d
x = Val(InputBox("Введите переменную x "))
Select Case x
Case Is < c
    y = 2 * x ^ 2 - 4 * x + 5
Case Is >= d
    y = Exp(x) - 17
Case Else
    y = 10 * Sin(x ^ 2) + 5 * Cos(x)
End Select
Debug.Print "x="; x, "y="; y
End Sub

```

**Примечание:** проверку условий можно производить в любой последовательности. Целесообразно проверять простые условия, как в первом варианте. Для записи сложных условий необходимо использовать логические функции, в данном примере использована функция AND.

#### 5. Проверка алгоритма

При  $a=1$  и  $b=2$   $C=1$ ,  $D=3$

Для проверки правильности работы алгоритма необходимо три раза запустить программу при значениях аргумента принадлежащих всем трем участкам определения функции, например:  $x=-2$ ,  $x=2$ ,  $x=4$ . Для проверки правильности полученных результатов необходимо сравнить их с известными значениями функций или вычислить эти выражения другими средствами, например, с помощью математической системы MathCad.

#### Результаты проверки:

1) $x = -2$	$y = 21$	
2) $x = 2$		$y = -9.648759$
3) $x = 4$		$y = 37.59815$

#### Контрольные вопросы

1. Приведите структуру и поясните принцип работы простого оператора IF.
2. Приведите структуру и поясните принцип работы расширенного оператора IF.
3. Приведите структуру и поясните принцип работы многострочного оператора IF.
4. Изобразите графически условие задачи  $y=\sin(x)$  при  $x < 0$ ,  $y=\cos(x)$  при  $0 <= x <= 2$  и  $y=e^x$  при  $x > 2$ .
5. Что такое полная группа событий применительно к решению задач на выбор? Приведите примеры полной и не полной группы событий.
6. Какое минимальное число проверок необходимо выполнить, если полное число событий в условии задачи равно двум (трем, четырем, ...).
7. Поясните синтаксис оператора Select Case.

## 3.6. Циклы с заданным числом повторений

**Цель занятия:** изучить организацию циклических процедур с помощью оператора If/Then и For/Next. Приобрести практические навыки в табулировании функций, вычислении сумм и произведений, поиске минимума и максимума функции. Закрепить знания в использовании условного оператора.

**Время:** 4 часа.

**Литература:** Л1 с.187 - 204, Л2 с. 182 – 187, с. 204 – 217.

### 3.6.1. Общие сведения

Циклом с заданным числом повторений называется процедура, в которой вычислительные операции выполняются многократно заданное число раз. Циклы этого типа называются циклами типа "ДО". Циклы могут быть организованы как циклы с предусловием, так и как циклы с постусловием. В циклах с предусловием вначале проверяется условие окончания цикла и, если условие окончания цикла не выполняется, то выполняется тело цикла. В таких циклах значение переменной, проверяемой в теле цикла, должно быть вычислено заранее или ей должно быть присвоено некоторое значение, заведомо большее условия окончания цикла, до входа в цикл. В циклах с постусловием условие окончания цикла проверяется в конце цикла. По смыслу решаемых задач циклы типа ДО организуются как циклы с постусловием.

Циклы могут быть организованы с использованием оператора IF или с использованием специального оператора FOR/NEXT:

a) цикл с использованием оператора IF

i=нач: REM заголовок цикла

M1:

<тело цикла>

i=i+di ' приращение прем. цикла

IF i<=N THEN GOTO M1:

REM конец цикла

PRINT "результат";F

b) цикл с использованием оператора FOR/NEXT

FOR i=нач TO N STEP di

<тело цикла>

NEXT i

PRINT "результат";F

### 3.6.2. Задание

1. Изучите описание лабораторной работы, необходимые сведения из рекомендованной литературы, ответьте на контрольные вопросы.
2. Вычислите сумму членов натурального ряда чисел от 1 до 100 с шагом 3.
3. Выполните табулирование функции  $y=f(x)$  согласно варианту задания на отрезке  $[c,d]$  с шагом  $dx$ . Границы отрезка выберите самостоятельно после вычисления точек разрыва функций.
4. Найдите минимум и максимум функции на отрезке  $[a,b]$  с точностью до  $e=0.001$ .
5. Изобразите схему алгоритма для п. 3 задания.
6. Вычислите сумму ряда согласно варианту задания. Число членов ряда выберите самостоятельно.

Варианты заданий к п. 3 и 4 задания

1	$f(x) = \begin{cases} \frac{\cos^3\left(\frac{5x}{2} + 3a\right) - 0,079e^{2-\sin^2(x)}}{\sqrt[4]{x^2 + 4b} \cdot \lg (a+b)x } \\ \sqrt[3]{((2ax - b)/((a - b)/2))^2 + \ln((2ax - b)/(a - b)/2)} \end{cases}$	<p>при <math>x &lt; \frac{a-b}{2}</math>  при <math>x \geq \frac{a+b}{2}</math></p>
2	$f(x) = \begin{cases} \frac{2\ln(3x+1) + e^{3x-1} - \operatorname{ctg}(x)}{\sqrt[3]{ e^{3x-1}  + \operatorname{tg}(ax)}} - \operatorname{ctg}(x) \\ \frac{(1 + 4x^2) \operatorname{tg}^3(\pi x/2) - 4,2b}{\sqrt[4]{x^2 + 7x + \cos(x)}} \end{cases}$	<p>при <math>x \leq a - b</math>  при <math>x \geq a - b</math></p>
3	$f(x) = \begin{cases} \frac{1,2 \sin(1,3x^3) + \operatorname{tg}(1,3x^3)}{\cos(\ln \sqrt{x}) + 2a} \\ \arcsin(1,3x^3) \end{cases}$	<p>при <math>\sqrt[3]{x} &gt; a + b</math>  при <math>\sqrt[3]{x} &lt; a^2 - b</math></p>
4	$f(x) = \begin{cases} \sum_{i=0}^n \frac{(\sqrt{ x-a } - \sqrt[3]{(x-b)^2}i)}{4a \sin(ix)} \\ \lg(3x^2 - (a+b)/2) + e^{x-a} \end{cases}$	<p>при <math>(x-a)^3 &gt; b</math>  при <math>(x-a)^3 &lt; \frac{a+b}{2}</math></p>
5	$f(x) = \begin{cases} 5e^{1-3x} - \cos^3 1 - 3x  \\ \sum_{i=1}^n 2i \sin^2(e^{2ix}) \end{cases}$	<p>при <math>\sqrt[5]{x} \leq \frac{\alpha}{\beta}</math>  при <math>\sqrt[5]{x} &gt; \alpha + \beta</math></p>
6	$f(x) = \begin{cases} \frac{\sin^2(abx) + \cos(abx^2)}{1,75 \ln( x ) + e^{-x}} \\ 5e^{1-3x} - \cos^3( 1 - 3x ) \end{cases}$	<p>при <math>3x &lt; \frac{a+b}{2}</math>  при <math>3x &gt; a + b</math></p>
7	$f(x) = \begin{cases} \sqrt{\sin^3(x)} - \sqrt[3]{\sin^2(x)} \\ \sqrt{\sin(x^2) + \frac{\ln(ix)}{\lg(bx)}} \end{cases}$	<p>при <math>x^3 &lt; a</math>  при <math>x^3 &lt; a + b</math></p>
8	$f(x) = \begin{cases} \frac{\sin^2(2x) - \frac{e^{-x}a \ln(x)}{2x \cdot b}}{\lg( \sin^3 x^2 )} \end{cases}$	<p>при <math>\sqrt[3]{x} \leq \frac{a+b}{2}</math>  при <math>\sqrt[3]{x} &gt; a + b</math></p>
9	$f(x) = \begin{cases} \sqrt{(2x+b)} + \arcsin(2x+b) \\ \sqrt{(x+2ax)^2} - \sqrt[5]{x+2ax} \end{cases}$	<p>при <math>\sqrt[5]{x} &lt; 2a - b</math>  при <math>\sqrt[5]{x} &gt; 2a</math></p>
10	$f(x) = \begin{cases} \sqrt{(2x+b)^3} + \arccos(2x+b) \\ \sqrt{(x+2ax)^3} - \sqrt[5]{x+2ax} \end{cases}$	<p>при <math>\sqrt[5]{x} &lt; 2a - b</math>  при <math>\sqrt[5]{x} &gt; 2a</math></p>

11	$f(x) = \begin{cases} \sqrt{2x^2 + b} + \sin(2x + b) \\ \sqrt{(x^3 + 2ax)^2} - \sqrt[5]{x^2 + 2ax} \end{cases}$	при $x < (b-a)/2$ при $x > (b+a)/2$
12	$f(x) = \begin{cases} \cos(2x) - \frac{e^{-x} a \lg(x)}{2x \cdot b} \\ \ln( \sin^3 x^2 ) \end{cases}$	при $\sqrt[3]{x} < a - b$ при $\sqrt[3]{x} > a + b$
13	$f(x) = \begin{cases} \sqrt{\cos^3(x)} - \sqrt[3]{\sin^2(x)} \\ \sqrt{\sin^3(x^2) + \frac{\ln(x)}{\lg(bx)}} \end{cases}$	при $\sqrt[5]{x} < (a - b)/2$ при $\sqrt[5]{x} > (a + b)/2$
14	$f(x) = \begin{cases} \frac{\sin^2(bx) + \sin^3(ax^2)}{1,75 \ln( x ) + e^{-x}} \\ 3e^{1-3x} - \sin^3( 1 - 2x ) \end{cases}$	при $\sqrt[5]{x} < (2a - b)/3$ при $\sqrt[5]{x} > (2a + b)/3$
15	$f(x) = \begin{cases} 3e^{1-2x} - \cos^3 1 - 5x  \\ \sqrt[3]{2 \sin^3(e^{2x})} \end{cases}$	при $\sqrt[3]{x} < 2a - b$ при $\sqrt[3]{x} > 2a$

#### Варианты заданий к п. 6 задания

1	$\sum_{i=1}^n (2i+1) \arccos(2ix)$	2	$\sum_{i=1}^n \sqrt{(1-ix)^3 + \frac{\ln(\sin(2x))}{\lg(\lg(\beta x))}}$
3	$\sum_{i=0}^n (i+1) \frac{e^{\cos(x)} + 2,53b}{\sin(x \cdot i)}$	4	$\sum_{i=1}^n 2^{\sqrt{x+a}} + 3,5i \cos^2(x - b)^3$
5	$\sum_{i=0}^n (e^{-2x} - i \lg( x ))$	6	$\sum_{i=0}^n i \cos^2(i + x)$
7	$\sum_{i=0}^n (e^{-2x} - i \log_{0,5}( x ))$	8	$\sum_{k=1}^{m-2} 2,7ak \cdot \sin^2\left(\frac{(a-b)x}{(a+b)k}\right)$
9	$\sum_{i=0}^n \frac{(\sqrt{ x-a } - \sqrt[3]{(x-e)^2} \cdot i)}{4a \sin(ix)}$	10	$\sum_{k=0}^{n-2} \frac{k + \sin^2(ab - x^2)}{k \cdot \lg(ab - x^2)}$
11	$\sum_{i=0}^n (i+5) \frac{e^{\sin(x)} + 2,53b}{\sin(2x \cdot i)}$	12	$\sum_{i=1}^n 3^{\sqrt{x^3}} + 3,5i \cos^2(x - b)^3$
13	$\sum_{i=0}^n i \sin^2(2i + x)$	14	$\sum_{i=0}^n (5e^{-3x} - 2i \lg( x ))$
15	$\sum_{k=1}^{m-2} 2,7ak \cdot \sin^2\left(\frac{a}{(a+b)k}\right)$	16	$\sum_{i=0}^n (e^{-2x} - i \log_{0,5}( x ))$

### 3.6.3. Пример выполнения задания

**Пример 3.3.** Вычислить значение выражения

$b = (1 + 1/2 + 1/2^2 + 1/2^3 + \dots + 1/2^n)$  при  $N = 10$

<pre>Rem вычисление суммы с использованием оператора FOR/NEXT N=Val(Input"введите значение N") B=1: y=1 For i=0 To N     y=y*1/2     B=B+y Next i Print "B="B</pre>	<pre>Rem вычисление суммы с использованием оператора IF N=Val(Input "введите значение N")) B=1: y=1: i=1 M: y=y*1/2 B=B+y i=i+1 IF i&lt;=N Then Goto M Print "B="B</pre>
---	--

B=1.999

Структура программы с использованием оператора IF отражает внутреннюю структуру оператора FOR/NEXT.

1. Выполнить табулирование функции  $y = \sin(x)$  на отрезке  $[0, \pi]$  с шагом 5 градусов.

Примечание: границы отрезка табулирования задаются в радианах.

Option Explicit

Dim a As Single, b As Single, h As Single, hr As Single

Dim N As Integer, x As Single, y As Single, i As Integer

---

```
Private Sub Form_Click()
```

```
Rem табулирование функции
```

```
Cls
```

```
Const Pi = 3.141592
```

```
a = Val(InputBox("Введите границу a отрезка табулирования"))
```

```
b = Val(InputBox("Введите границу b отрезка табулирования"))
```

```
h = Val(InputBox("Введите шаг табулирования h в градусах"))
```

```
' шаг табулирования задан в градусах, переведем его в радианы
```

```
hr = h / 180 * Pi      ' hr - шаг в радианах
```

```
' определим число шагов
```

```
N = Int((b - a) / hr) + 1
```

```
x = a
```

```
For i = 0 To N
```

```
    y = Sin(x)
```

```
    Print "x="; x, "y="; y
```

```
    Debug.Print "x="; x, "y="; y
```

```
    x = x + hr
```

```
Next i
```

```
End Sub
```



## Контрольные вопросы

1. Какие операторы используются для организации цикла?
2. Начертите схему алгоритма цикла с постусловием.
3. Начертите схему алгоритма вычисления суммы конечного ряда чисел.
4. Начертите схему алгоритма табулирования функции.
5. Начертите схему алгоритма поиска максимума (минимума) функции.
6. Приведите синтаксис оператора For/Next.

## 3.7. Операции с массивами

**Цель занятия:** приобрести навыки работы с массивами

**Время:** 2 часа.

**Литература:** Л1 с. 204- 210,266 - 271, Л2 с. 217 – 226, Л3 с. 74-81.

### 3.7.1. Общие сведения

Под массивом понимается совокупность индексированных элементов, имеющих одно имя и отличающихся индексом.

Различают статические и динамические массивы. У статических массивов размерность по умолчанию объявляется с помощью числовых констант, у динамических массивов – с помощью переменных или констант. Статический массив объявляется один раз на уровне формы или модуля оператором Dim или Public:

Dim МассивА(3) одномерный массив;

Dim МассивВ(5,6) – двухмерный массив;

Dim МассивС(1 To 5, 5 To 10) – двухмерный массив.

Public А(5) As Integer, В(1 To 10) As Single

Динамические массивы объявляются дважды: первый раз на уровне формы или модуля без указания размерности, например:

Option Explicit

Dim А() As Single, В() As Variant

Dim А1() As String, А2() As Integer, А3() As Single

- второй раз в процедуре оператором ReDim с указанием размерности, например:

Private Sub Form\_Click()

Dim N As Integer, M As Integer, N1 As Integer, M1 As Integer

Dim N2 As Integer, M2 As Integer

N = Val(InputBox("Укажите число строк", "Ввод размерности массива"))

M = Val(InputBox("Укажите число столбцов", "Ввод размерности массива"))

N1 = Val(InputBox("Укажите число строк", "Ввод размерности массива"))

M1 = Val(InputBox("Укажите число столбцов", "Ввод размерности массива"))

N2 = Val(InputBox("Укажите число строк", "Ввод размерности массива"))

M2 = Val(InputBox("Укажите число столбцов", "Ввод размерности массива"))

ReDim А1(N) As String '- одномерный массив;

ReDim А2(N, M) As Integer '- двухмерный массив;

ReDim А3(N1 To M1, N2 To M2) As Single '- двухмерный массив

ReDim А(N, M) As Single, В(1 To N1, 5 To 10) As Variant

При объявлении динамического массива в процедуре нельзя изменять его тип, кроме тех случаев, когда массив был объявлен типа Variant.

Нумерация элементов массива начинается с нуля. Для изменения начального номера элементов массива с нуля на единицу используется оператор Option Base 1. Данный оператор записывается в разделе общие формы или модуля до объявления массивов.

Для определения границ массивов могут использоваться функции Lbound(<имя массива>, индекс) и Ubound(<имя массива>, индекс).

Visual Basic позволяет непосредственно присваивать значения одного массива другому: МассивА=МассивВ. При этом, в общем случае, типы данных и размерности массивов должны совпадать.

Удаление динамических массивов:

Erase <имя массива>

Элемент одномерного массива обозначается A(i), где i – текущий индекс массива.

Элемент двумерного массива обозначается A(i, j), где i – номер строки, j – номер столбца.

### 3.7.2. Задание

1. Введите данные в одномерный массив и выведите данные из массива на форму. Изобразите схему алгоритма.

2. Введите данные в двумерный массив и выведите данные на форму. Изобразите схему алгоритма.

3. Введите данные в двумерный массив, определите максимальный элемент массива и сумму элементов массива. Выведите массив на форму используя оператор Print и в Окно непосредственного наблюдения (Immediate), используя метод Print объекта Debug.

4. Отсортируйте одномерный массив по возрастанию (убыванию) значений элементов массива. Выведите на экран параллельно исходный массив и отсортированный массив.

5. Поменяйте местами значения массивов А и В. При вводе данных контроль ввода осуществляйте с помощью окна непосредственного наблюдения. На форму выведите массивы А и В в виде матриц.

**Указание:** объявите вспомогательный массив, используя функции определения границ массивов.

6. Выполните задание согласно варианту.

#### Варианты задания

1. Протабулировать функцию  $y=f(x)$ , где  $f(x)$  – произвольная функция одной переменной. Результаты записать в двумерный массив.

2. Найти сумму и произведение нечетных элементов массива.

3. Найти сумму и произведение четных элементов массива.

4. Найти сумму элементов массива, расположенных на главной диагонали двумерного массива.

5. Создать одномерный массив, каждый элемент которого равен сумме элементов строки двумерного массива.

6. Выполнить транспонирование двумерного массива и найти сумму элементов этого массива.

7. Дан двумерный массив A(n,m). Создать массив В, каждый элемент которого не превосходит единицы массива А.

**Указание.** Найти максимальный по модулю элемент  $A_{max}$  массива  $A$ . Элементам массива  $B$  присвойте значение  $A(i,j)/A_{max}$

8. Найти скалярное произведение двух векторов  $C = \sum A(i)B(i)$ . Вывести на экран исходные данные и результат.

9. Дан массив  $A(n,m)$ . Найти максимальный элемент массива. Удалить из массива строку и столбец, содержащий этот элемент. Результаты записать в массив  $B$ .

10. Дан одномерный массив  $A(n)$ . Преобразовать массив  $A$  в двумерный массив  $B(m,L)$ , в котором  $L$  столбцов. Недостающие элементы массива  $B$  заполнить нулями.

**Указание.** Определить число строк в массиве  $B$ :  $m = \text{Int}(n/L) + 1$ . Ввести вспомогательную переменную  $k$  – счетчик для определения текущего элемента массива  $A$ . Организовать два вложенных цикла. В циклах элементам массива  $B(i,j)$  присваивать значение элемента массива  $A(k)$ . Когда  $k$  станет больше  $n$ , элементам массива  $B(i,j)$  присваивать значение 0.

11. Вычислить определитель двумерного массива.

12. Найти матрицу  $C(M,P)$  равную произведению матрицы  $A(M,N)$  и матрицы  $B(N,P)$ .

**Указание.**  $C(i, k) = \sum A(i, j)B(j, k), \quad i = \overline{1, M}, j = \overline{1, N}, k = \overline{1, P}$

13. Найти максимальный элемент массива. Поменять значения двух строк: строке, номер которой равен номеру строки, содержащей максимальный элемент, присвоить значения строки, номер которой равен номеру столбца, содержащего максимальный элемент, и наоборот. (Если номер строки, в которой находится максимальный элемент массива, равен, например, 5, а номер столбца, в котором находится максимальный элемент равен 3, тогда надо поменять местами строки 3 и 5).

**Указание.** Найдите максимальный элемент массива  $A(N,M)$ . Пусть это будет элемент  $A(p,q)$ . Введите вспомогательный одномерный массив  $B(M)$ . Присвойте ему значение элементов строки  $q$ , строке  $q$  присвойте значения элементов строки  $p$ . Строке  $p$  присвойте значения элементов массива  $B$ .

14. Создайте массив  $B$ , каждый элемент которого равен среднему значению строки массива  $A$ .

15. Четным элементам массива  $B$  присвойте значения четных элементов массива  $A$ , а нечетным элементам массива  $B$  присвойте значение 1. Подсчитайте сумму четных и нечетных элементов массива  $B$ .

### 3.7.3. Примеры выполнения задания

Фрагмент программы реализации задания по п.5.

Option Explicit

Dim A() As Single, B() As Single, T() As Single

Private Sub Form\_Click()

Cls

Dim N As Integer, M As Integer, i As Integer, j As Integer

N = Val(InputBox("Укажите число строк N массива A(N,M)"))

M = Val(InputBox("Укажите число столбцов M массива A(N,M)"))

ReDim A(N, M) As Single, B(N, M) As Single

ReDim T(LBound(A, 1), UBound(A, 2)) As Single ' Вспомогательный массив

' Ввод данных в двумерный массив A

For i = 0 To N

For j = 0 To M

```

        A(i, j) = Val(InputBox("Введите значение элемента A(" & Str(i) & "," & Str(j) & ")"))
        Debug.Print A(i, j);
    Next j
    Debug. Print
Next i

' Ввод данных в двумерный массив B
For i = 0 To N
    For j = 0 To M
        B(i, j) = Val(InputBox("Введите значение элемента B(" & Str(i) & "," & Str(j) & ")"))
        Debug.Print B(i, j);
    Next j
    Debug.Print
Next i

' Обмен данными массивов
T = A: A = B: B = T

' Вывод массива A на печать в виде матрицы
For i = 0 To N
    For j = 0 To M
        Print A(i, j);
    Next j
    Print
Next i

' Вывод массива B на печать в виде матрицы
For i = 0 To N
    For j = 0 To M
        Print B(i, j);
    Next j
    Print
Next i
End Sub

```

## Контрольные вопросы

1. Что такое статические и динамические массивы? Чем они отличаются?
2. Какие операторы используются для объявления массивов?
3. Как объявить статический массив?
3. Как объявить динамический массив?
4. Как выполнить переопределение динамического массива?
5. Напишите текст программы для ввода данных в одномерный массив.
6. Начертите схему алгоритма ввода данных в одномерный массив.
7. Какие требования предъявляются к организации вложенных циклов?
8. Как вывести на печать двумерный массив в виде матрицы?
9. Какой элемент массива называется четным (нечетным)?
10. Напишите программу сортировки массива? Изобразите схему алгоритма сортировки массива.
11. Перечислите функции для определения границ массивов и поясните их синтаксис.
12. Как поменять значения у двух массивов?
13. Изобразите схему алгоритма для ввода данных в двумерный массив.

### 3.8. Функции пользователя

**Цель работы:** приобрести практические навыки в создании и использовании функций пользователя.

**Время:** 2 часа.

**Литература:** Л1 с. 259 - 265, Л2 с. 228 – 232.

#### 3.8.1. Общие сведения

Часто при вычислениях приходится вычислять одни и те же выражения при различных исходных данных. С целью экономии времени, ресурсов, а также для обеспечения возможности использования известных алгоритмов в других программах, Visual Basic позволяет создавать процедуры и функции пользователя.

Синтаксис функции имеет следующий вид:

```
Function <имя функции>(параметры)
    <тело функции>
    <имя функции>=<результат>
End Function
```

Обращение к функциям пользователя осуществляется так же, как и к встроенным функциям языка программирования. Функция должна быть объявлена до ее использования.

#### 3.8.2. Задание

1. Изучите описание лабораторной работы и рекомендованную литературу.
2. Протестируйте разветвляющуюся функцию согласно варианту задания. Функции оформите с помощью функции пользователя.
3. Вычислите факториал произвольного числа с использованием функции пользователя.

#### Варианты заданий

1	$y = \begin{cases} 5e^x + 3^3 \cos(4x - 0.2), & x < a \\ \operatorname{ctg}(4.5x) + \frac{\sin(x)}{\cos(0.5x)}, & x > a \end{cases}$	2	$y = \begin{cases} 3\sqrt{ \cos(x) } + e^{2x}, & x < b \\ (x^{-3} - 2)/\cos(3x - 0.5), & x \geq b \end{cases}$
3	$y = \begin{cases} e^{2x} + \sqrt{3 \cos(x - 0.3) }, & x < b^2/a \\ e^{-2x} + \sqrt{ \sin(2x + 0.6) }, & x > b^2/a \end{cases}$	4	$y = \begin{cases} \ln(x^2) - e^x + \lg( 2x ), & x < 2a + b \\ \sin(2x) - 3x + 1, & x > 2a + b \end{cases}$
5	$y = \begin{cases} \frac{2e^x + \sqrt{x^3 + 5}}{3 \cos(3x^2)}, & x \geq 2ab \\ \lg( 1.5x^3 ), & x < 2ab \end{cases}$	6	$y = \begin{cases} x^3 - \ln 3x^2  + \sin^3(x), & (b - a) < x \\ \sqrt{ b + \operatorname{ctg}(x) - e^{3x} }, & (b - a) \geq x \end{cases}$
7	$y = \begin{cases} \ln\left(\frac{x-1}{x+1}\right) - e^{3x}, & \sin(x) < a^2 + b \\ 4x^3 + 6x - \sqrt{x^3} -  \operatorname{tg}(x) , & \sin(x) \geq a^2 + b \end{cases}$	8	$y = \begin{cases} 5e^{2x} + \sqrt{ x^4 + 2x^3 }, & b < a^2 + x \\ \frac{\sqrt{ x^4 + 4x }}{\operatorname{ctg}(2 + x)}, & b \geq a^2 + x \end{cases}$

9	$y = \begin{cases} \frac{3xe^3}{ 1.7+2x } + \sin^2(x), & ba < 2x \\ (e^x + 6x^2 + 10.5)^{\sqrt[3]{\cos(x^2)}}, & ba \geq 2x \end{cases}$	10	$y = \begin{cases} \frac{-x}{\ln(x^2) - e^{\frac{1}{3}} + \lg( x-1.8 )}, & x \leq (2a-b) \\ \lg(2x+5.7) + \sqrt{3x+5}, & x > (2a-b) \end{cases}$
11	$y = \begin{cases} e^{1.4x} + 3\sqrt{ x^4 + 3 }, & b^2 < a^2 + x \\ \operatorname{ctg}^2(x^3) + \arccos(x), & b^2 \geq a^2 + x \end{cases}$	12	$y = \begin{cases} 1.3x^2 + \sin^3(x), & ba < 2x \\ (2\ln(x^2) + 6x^2)\sqrt{\sin(2x)}, & ba \geq 2x \end{cases}$
13	$y = \begin{cases} x^{1.4} - x^2 \sin(2x - 0.6), & a^2 + b^2 < 3x \\ \sin(x^3 - 4.3) + \sqrt[3]{x^2}, & a^2 + b^2 \geq 3x \end{cases}$	14	$y = \begin{cases} (1.9x^2 - 4.3) \sin(x^3) - \sqrt{2x}, & a + x \geq b \\ x^3 - \sqrt{ \operatorname{tg}(2x^3 - 1) }, & a + x < b \end{cases}$
15	$y = \begin{cases} e^x + \cos( 1.2x ), & x \leq a + b \\ \sqrt{5.8 + \sin(3x)} - 2^{2x}, & x < a + b \end{cases}$	16	$y = \begin{cases} \operatorname{arctg}(5x - 0.7), & a + 2b \geq x \\ \lg( x + 3.4 ) + \cos(2.1x), & a + 2b < x \end{cases}$

### 3.8.3. Примеры выполнения задания

Найти экстремум функции согласно заданию 3.6.3 с использованием функций пользователя.

Option Explicit

Dim a As Single, b As Single, h As Single, hr As Single

Dim N As Integer, x As Single, y As Single, i As Integer

Dim Ymax As Single, Xmax As Single, Pi As Single

-----  
Private Function FuncY1(x)

y=10 \* Sin(x) + x ^ 2 - 5

FuncY1=y

End Function

-----  
Private Sub Form\_Click()

Rem Поиск экстремумов функции на отрезке табулирования

Pi = 4 \* Atn(1)

a = Val(InputBox("Введите границу a отрезка табулирования"))

b = Val(InputBox("Введите границу b отрезка табулирования"))

h = Val(InputBox("Введите шаг табулирования h"))

' определим число шагов

N = Int((b - a) / hr) + 1

x = a

Ymax = FuncY1(x)

For i = 0 To N

y = FuncY1(x)

If y > Ymax Then Ymax = y: Xmax = x

x = x + hr

Next i

Print "Xmax="; Xmax, "Ymax="; Ymax

End Sub

## Контрольные вопросы

1. Приведите синтаксис функции пользователя.
2. Как используется функция пользователя?
3. Напишите функцию пользователя для вычисления суммы конечного числа членов ряда.
4. Напишите функцию пользователя для вычисления пути, пройденного автомобилем при равноускоренном движении.

## 3.9. Циклы с параметром

**Цель работы:** познакомиться с операторами для работы с циклами с параметром.

**Время:** 2 часа.

**Литература:** Л1 с.187 - 204, Л2 с. 204 – 217.

### 3.9.1. Общие сведения

Циклы с параметром (циклы типа "Пока") отличаются от циклов с заданным числом повторений (циклы типа "До") тем, что в этих циклах число шагов заранее неизвестно. Условием окончания цикла является достижение некоторой переменной, вычисляемой в теле цикла, некоторого, наперед заданного значения. Эти циклы используются для вычисления рядов и других подобных задач. Примеры организации циклов с параметром приведены в табл.3.9.1.

Организируются циклы с параметром как циклы с предусловием и как циклы с постусловием. Для их организации могут использоваться операторы If/Then, While/Wend, Do/Loop. Циклы с оператором Do/Loop могут быть организованы как циклы с предусловием, так и как циклы с постусловием. Особенностью циклов с предусловием является то, что значение контролируемого параметра должно быть задано или вычислено до входа в цикл. Иначе может случиться, что программа никогда не войдет в цикл и будет получаться ложный результат.

Таблица 3.9.1 Примеры организации циклов с параметром

Цикл с предусловием F=<выражение> i=iнач M1: IF F<e THEN GOTO M2 <тело цикла> i=i+1 GOTO M1 M2: PRINT "результат";F	F=<выражение> i=iнач WHILE F>e <тело цикла> i=i+1 WEND PRINT "результат";F	F=<выражение> i=iнач DO [WHILE/UNTIL] <тело цикла> i=i+1 LOOP [WHILE/UNTIL] PRINT "результат";F
При использовании опции WHILE реализуется положительная логика, т. е. цикл выполняется в том случае, когда условие истинно. При использовании опции UNTIL реализуется отрицательная логика, т. е. цикл выполняется в том случае, когда условие ложно		

### 3.9.2. Задание

1. Изучите описание лабораторной работы, необходимые сведения из рекомендованной литературы, ответьте на контрольные вопросы.

2. Вычислите сумму ряда чисел  $S = \sum_{i=1}^n 1/i$  с точностью до 0.001.

Примечание: для знакочередующихся рядов абсолютная погрешность его частичной суммы не превосходит абсолютной величины первого из отброшенных членов ряда. Для положительных рядов эта погрешность может на один-два порядка превышать величину первого из отброшенных членов ряда. Поэтому для практических целей можно полагать, что точность вычисления суммы сходящегося ряда равна величине первого из отброшенных членов ряда.

3. Вычислите сумму ряда в соответствии с вариантом задания.

4. Начертите схему алгоритма для задачи по п. 3

### Указания к выполнению задания

Для вычисления факториала создайте функцию пользователя.

### Варианты заданий

В задании А вычислить сумму членов ряда с точностью до  $\Pi$

В задании Б определить число членов ряда, при котором достигается заданная точность  $\epsilon$ .

Аргумент  $x$  при вычислениях суммы имеет постоянное значение.

	А		Б	
	Выражение	Точность	Выражение	Точность
1	$\sum_{n=1}^{\infty} \frac{2^n}{1+2^{2n}}$	0,001	$\sum_{n=1}^{\infty} \frac{\cos(5n)}{\sqrt{n^3+1} \cdot n!}$	0,01
2	$\sum_{n=1}^{\infty} \frac{1}{(n+1)2^{n-1}}$	0,001	$\sum_{n=1}^{\infty} \frac{\sin(n+1)}{(n+5)!}$	0,001
3	$\sum_{n=1}^{\infty} \frac{2^{n^2}+1}{n\sqrt{n}+3}$	0,01	$\sum_{n=1}^{\infty} \frac{(-1)^{n-1}n^2}{(3n-1)!}$	0,001
4	$\sum_{n=1}^{\infty} \frac{1}{(n+3)^n}$	0,001	$\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{(n+2)!}$	0,01
5	$\sum_{n=1}^{\infty} \frac{1}{(3n+\sqrt{n})}$	0,01	$\sum_{n=1}^{\infty} \frac{(-1)^n}{(2n)!}$	0,001
6	$\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{(n+5)!}$	0,0001	$\sum_{n=1}^{\infty} \frac{(x+1)^n}{\sqrt[3]{n+2}}$	0,01
7	$\sum_{n=1}^{\infty} \frac{(-1)^n}{2n^2+1}$	0,001	$\sum_{n=1}^{\infty} \frac{(x+2)^n}{(n+1)!(5^n+1)}$	0,001
8	$\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{2n}$	0,001	$\sum_{n=1}^{\infty} \frac{(x-2)^n}{(2n)!}$	0,01

9	$1 + \sum_{n=1}^{\infty} \frac{x2}{(n-1)!}$	0,01	$\sum_{n=1}^{\infty} \frac{(-1)^n}{n^2 2^n}$	0,001
10	$\sum_{n=1}^{\infty} (-1)^n \left( \frac{n}{(n+1)(n+2)} \right)^n$	0,01	$\sum_{n=1}^{\infty} \frac{x^n \sin(nx)}{(n+3)!}$	0,01
11	$\sum_{n=1}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!}$	0,001	$\sum_{n=1}^{\infty} \frac{1}{(2n-1)x^{2n-1}}$	0,01
12	$\sum_{n=1}^{\infty} \frac{2^n}{1+2^{2n}}$	0,001	$\sum_{n=1}^{\infty} \frac{\lg^n(x)}{(n+2)!}$	0,01
13	$\sum_{n=1}^{\infty} \frac{n^2+1}{(n+2)!}$	0,01	$\sum_{n=1}^{\infty} \sin^n\left(\frac{\pi x}{4}\right)$	0,001
14	$\sum_{n=1}^{\infty} \frac{n}{2^n(3n-1)!}$	0,001	$\sum_{n=1}^{\infty} \frac{n}{x^n}$	0,001
15	$\sum_{n=1}^{\infty} \frac{1}{(2n+1)!}$	0,0001	$\sum_{n=1}^{\infty} \frac{2}{x \ln^n(x)}$	0,01

### Контрольные вопросы

1. В чем отличие циклов с параметром от циклов типа "ДО"?
2. Какие операторы используются для организации циклов с параметром?
3. Начертите схему алгоритма цикла с параметром?
4. Приведите синтаксис оператора While/Wend.
5. Приведите синтаксис оператора Do/Loop.
6. Для какой цели используются циклы с заданным параметром? Приведите примеры.
7. Приведите схему алгоритма вычисления функции  $e^x$ , когда функция представлена в виде бесконечного ряда.

## 3.10. Численное интегрирование

**Цель работы:** закрепить навыки в разработке простых приложений, а также разработке циклических программ с использованием операторов For/Next, While/Wend и Select Case. Приобрести навыки в использовании функций пользователя.

**Время** – 4 часа.

**Литература:** Л1 с. 206 - 210, Л3 с. 69 – 74.

### 3.10.1. Общие сведения

#### Вычисление определенного интеграла

Сущность метода численного интегрирования состоит в том, что отрезок интегрирования  $[a; b]$  разбивают на  $n$  элементарных отрезков  $[x_{i-1}, x_i]$  и заменяют интеграл функции  $f(x)$  суммой прямоугольников  $S_i = f(t_i) \Delta x_i$  (рис.3.10.1), где  $f(t_i)$  — значение функции в некоторой точке  $(t_i)$  внутри  $i$ -го отрезка;  $\Delta x_i$  — длина отрезка,  $\Delta x_i = x_i - x_{i-1}$ , то есть

$$S_n = S_1 + S_2 + \dots + S_{n-1} + S_n + R = \sum_{i=1}^n f(t_i) \Delta x_i + R \quad (3.10.1)$$

Здесь  $S_n$  - интегральная сумма,  $R$  - погрешность усечения. Предел этой суммы при неограниченном увеличении числа отрезков разбиения, когда длина наибольшего из элементарных отрезков  $\Delta x_i$  стремится к нулю, есть интеграл от функции  $f(x)$  на отрезке  $[a; b]$ :

$$\int_a^b f(x) dx = \lim_{\max \Delta x_i \rightarrow 0} \sum_{i=1}^n f(t_i) \Delta x_i$$

Простейший метод численного интегрирования — *метод средних прямоугольников*. В нем используется замена интеграла интегральной суммой (3.10.1). В этом случае все отрезки  $\Delta x_i$  равны между собой, а в качестве  $f(t_i)$  принимается значение функции в середине отрезка. Пусть  $h$  — длина элементарного отрезка, тогда формула примет следующий вид

$$\int_a^b f(x) dx = h * \sum_{i=1}^n f(a + h(i - 1/2)) \quad (3.10.2)$$

или после приведения к рекуррентной форме  
 $S_{i+1} = S_i + f(x_i) * h$ ,  $x_0 = a + h/2$ ,  
 $x_{i+1} = x_i + h$ .

$$(3.10.3)$$

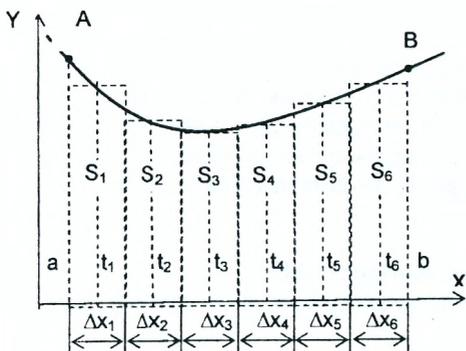


Рис. 3.10.1. Вычисление определенного интеграла методом средних прямоугольников

Здесь  $a$  — нижний предел интегрирования.

Если значение функции вычисляется при  $x$  равном значению аргумента на границах отрезков деления то получают, соответственно, метод левых и метод правых прямоугольников.

Если в качестве интерполирующей функции используется линейная функция, то получаем *метод трапеций*. Формула трапеций после преобразования принимает вид:

$$S = h \left( \frac{y_0 + y_n}{2} + \sum_{i=1}^{n-1} y_i \right) \quad (3.10.4)$$

В методе Симпсона в качестве интерполирующей функции используется многочлен второй степени  $ax^2 + bx + c$ . Эти методы обеспечивают получение более точных результатов при меньшем числе шагов. Формула Симпсона записывается следующим образом:

$$S = \frac{h}{3} (y_0 + y_{2m} + 2(y_2 + y_4 + \dots + y_{2m-2}) + 4(y_1 + y_3 + \dots + y_{2m-1})) \quad (3.10.5)$$

Для достижения заданной точности применяют метод двойного прохода (пересчета). По двум смежным результатам вычисляется ошибка интегрирования. Для оценки ошибки интегрирования применяют приближенные формулы (принцип Рунге):

$$\Delta = |I_n - I_{2n}|/3 \text{ — для формулы трапеций}; \quad (3.10.6)$$

$$\Delta = |I_n - I_{2n}|/15 \text{ – для формулы Симпсона.} \quad (3.10.7)$$

Здесь  $I_n$  – значение интеграла при разбиении интервала интегрирования на  $n$  частей, а  $I_{2n}$  – значение интеграла при разбиении интервала интегрирования на  $2n$  частей.

Для оценки ошибки интегрирования в методах левых, средних и правых прямоугольников можно также воспользоваться формулой (3.10.6), или использовать разность двух смежных значений интеграла.

### Функция Format

Данная функция предназначена для управления выводом информации на форму или печатающее устройство. Функция позволяет указать число десятичных знаков, а также используемые разделители.

Синтаксис функции Format:

Print Format (значение, "шаблон")

Например: Print Format (1124.75786, "#####.###")

Символ # резервирует место под цифру, точка отделяет целую часть от дробной. В данном примере зарезервировано шесть знаков в целой части числа и два знака в дробной части числа. Особенность использования такого пользовательского шаблона состоит в том, что число не должно быть равно нулю. Если число равно нулю, то возникает ошибка периода выполнения и выполнение программы прерывается.

### 3.10.2. Задание

1. Разработайте форму для вычисления определенного интеграла с заданной точностью тремя способами, выбранными из известных способов: методом левых прямоугольников, методом правых прямоугольников, методом средних прямоугольников, методом трапеций, методом Симпсона - согласно варианту задания. Сравните полученные результаты по числу циклов, необходимых для достижения заданной точности.

#### Указания к выполнению задания

1. Исходные данные вводить с помощью окон ввода TextBox.
2. При разработке программы подынтегральные функции задаются в виде функций пользователя.
3. Программа должна позволять вычислять три функции. Результат вычислений и результат сравнения методов вычисления интеграла выводить в метки Label.
4. При загрузке формы на форму должна выводиться подынтегральная функция. Высота шрифта 10 пт, цвет – синий.

Варианты заданий

#### а) для вычисления определенного интеграла:

№ варианта	1	2	3	4	5	6	7	8	9	10
Задания	1,2,3,	1,2,4	1,2,5	1,3,4	1,3,5	1,4,5	2,3,4	2,3,5	2,4,5	3,4,5

Здесь 1 – метод левых прямоугольников, 2 – метод правых прямоугольников, 3 – метод средних прямоугольников, 4 – метод трапеций, 5 – метод Симпсона.

N	1	2	3
1	$\int_{-0.5}^{1.3} \frac{x^2 dx}{\sqrt{x^2 + 1}}$	$\int_{0.6}^{2.4} \frac{(1 + 0.5x^2) dx}{1 + \sqrt{0.8x^2 + 1.4}}$	$\int_{-0.2}^{0.8} \frac{\sin(2x + 0.5) dx}{2 + \cos(x^2 + 1)}$
2	$\int_2^{3.2} \frac{x + 2}{\sqrt{x^2 + 1}} dx$	$\int_{1.2}^{2.64} \frac{(1 + 1.2x^2) dx}{0.8 + \sqrt{x^2 + 1.3}}$	$\int_{0.3}^{0.9} \frac{\cos(0.8x + 1.2) dx}{1.5 + \sin(x^2 + 0.6)}$
3	$\int_{0.5}^{1.6} \frac{x^2 + 0.5}{\sqrt{x^2 + 1}} dx$	$\int_{0.8}^{2.6} \frac{(1 + 1.5x^2) dx}{0.7 + \sqrt{2.2x^2 + 0.5}}$	$\int_{0.4}^{1.0} \frac{\sin(x + 1.4) dx}{0.8 + \cos(2x^2 + 0.5)}$
4	$\int_{2.2}^{3.4} \frac{x^2 dx}{\sqrt{x + 1}}$	$\int_{0.8}^{2.96} \frac{(1 + 0.7x^2) dx}{1.5 + \sqrt{2x^2 + 0.3}}$	$\int_{0.6}^{1.0} \frac{\cos(0.6x^2 + 0.4) dx}{1.4 + \sin^2(x + 0.7)}$
5	$\int_{1.2}^2 \frac{x - 0.5}{\sqrt{x^2 - 1}} dx$	$\int_{0.6}^{2.4} \frac{(1 + 0.6x^2) dx}{0.9 + \sqrt{x^2 + 1.5}}$	$\int_{0.5}^{1.3} \frac{\sin(2x + 0.5) dx}{1.2 + \cos(x^2 + 0.4)}$
6	$\int_{2.2}^{3.8} \frac{x + 1}{\sqrt{x^2 + 2}} dx$	$\int_{0.5}^{2.66} \frac{(1 + 0.3x^2) dx}{1 + \sqrt{0.6x^2 + 1.2}}$	$\int_{0.4}^{0.8} \frac{\cos(x^2 + 0.6) dx}{0.7 + \sin(0.8x + 1)}$
7	$\int_{0.2}^{2.4} \frac{\sqrt{x^2 + 1}}{x + 2} dx$	$\int_{0.7}^{2.5} \frac{(1 + 1.5x^2) dx}{0.5 + \sqrt{x^2 + 0.8}}$	$\int_{0.2}^{1.8} \frac{\sin(0.2x + 1.5) dx}{1.3 + \cos^2(0.5x + 1)}$
8	$\int_1^{2.6} \frac{xdx}{\sqrt{x^2 + 3}}$	$\int_{0.9}^{2.34} \frac{(1 + 0.9x^2) dx}{1.3 + \sqrt{0.5x^2 + 1}}$	$\int_{0.5}^{1.8} \frac{\cos(x^2 + 0.6) dx}{1.2 + \sin(0.7x + 0.2)}$
9	$\int_{0.8}^{1.6} \frac{0.5x + 2}{\sqrt{x^2 + 1}} dx$	$\int_1^{3.16} \frac{(1 + 0.6x^2) dx}{1 + \sqrt{0.4x^2 + 2.5}}$	$\int_{0.4}^{1.2} \frac{\sin(1.5x + 0.3) dx}{2.3 + \cos(0.4x^2 + 1)}$
10	$\int_{-0.4}^{1.6} \frac{x + 1}{\sqrt{x^2 + 1}} dx$	$\int_{1.1}^{2.9} \frac{(1 + 0.7x^2) dx}{0.4 + \sqrt{0.8x^2 + 1.5}}$	$\int_{0.4}^{1.2} \frac{\cos(x^2 + 0.8) dx}{1.5 + \sin(0.6x + 0.5)}$
11	$\int_{-0.8}^{1.4} \frac{x^2 dx}{\sqrt{x^2 + 4}}$	$\int_{1.6}^{2.84} \frac{(1 + 0.3x^2) dx}{1 + \sqrt{1.8x^2 + 1}}$	$\int_{0.5}^{1.3} \frac{\sin(0.7x + 0.4) dx}{2.2 + \cos(0.3x^2 + 0.7)}$
12	$\int_{2.6}^{3.4} \frac{x + 0.5}{\sqrt{x^2 + 1.5}} dx$	$\int_{0.4}^{2.56} \frac{(1 + 0.3x^2) dx}{1 + \sqrt{0.6x^2 + 1.3}}$	$\int_{0.4}^{1.4} \frac{\cos(0.8x^2 + 1) dx}{1.4 + \sin(0.3x + 0.5)}$
13	$\int_{0.8}^2 \frac{xdx}{\sqrt{x^2 + 2}}$	$\int_{1.2}^{2.64} \frac{(1 + 0.5x^2) dx}{0.7 + \sqrt{0.5x^2 + 1.2}}$	$\int_{0.2}^1 \frac{\sin(0.8x^2 + 0.3) dx}{0.7 + \cos(1.2x + 0.3)}$
14	$\int_{2.4}^{3.2} \frac{x^2 dx}{\sqrt{x + 2}}$	$\int_{1.3}^{3.46} \frac{(1 + 0.9x^2) dx}{1.5 + \sqrt{0.4x^2 + 0.7}}$	$\int_{0.3}^{1.1} \frac{\cos(0.3x + 0.5) dx}{1.8 + \sin(x^2 + 0.8)}$
15	$\int_{0.2}^2 \frac{x + 0.5}{\sqrt{x^2 + 1}} dx$	$\int_{0.5}^{2.4} \frac{(1 + 1.2x^2) dx}{1 + \sqrt{0.6x^2 + 1.3}}$	$\int_{0.3}^{1.1} \frac{\sin(0.6x^2 + 0.3) dx}{2.4 + \cos(x + 0.5)}$

### 3.10.3. Пример выполнения задания

Вычислить определенный интеграл тремя способами: методом левых, средних прямоугольников и методом Симпсона с заданной точностью (рис. 3.10.2). Для ввода данных используется окно ввода (TextBox), а для вывода результатов – метки (Label). Меткам, используемым для вывода результатов, свойству BorderStyle присваивается значение Fixed Single.

#### Описание переменных:

Пример 3-10 (Имя программы)	Вычисление определенного интеграла (Назначение программы)		Лист	1
			Листов	1
Переменная			Комментарий	
обозначение в формуле	имя переменной	тип переменной		
a	a	вещественная оди- нарной точности	нижний предел интегри- рования	
b	b	вещественная од. точ.	верхний предел интег- рирования	
n	n	целая одинарной длины	число отрезков разби- ения	
-	e	вещественная од. точ.	точность вычисления интеграла	
h	h	вещественная од. точ.	шаг табуляции	
x	x	вещественная од. точ.	аргумент	
f(x)	y	вещественная од. точ.	подынтегральная функ- ция	
S	S	вещественная од. точ.	значение интеграла	
-	S1	вещественная од. точ.	значение интеграла на предыдущем шаге	
-	S11	вещественная од. точ.	вспомогательная пере- менная	
-	S12	вещественная од. точ.	вспомогательная пере- менная	
R	R	вещественная од. точ.	ошибка интегрирования	
-	i	целая од. длины	переменная цикла	
-	k	целая од. длины	счетчик числа проходов	

#### Текст программы:

```
Option Explicit
Dim a As Single, b As Single, n As Integer, h As Single
Dim S As Single, S1 As Single, R As Single, e As Single
Dim x As Single, y As Single
Dim S11 As Single, S12 As Single, i As Integer, k As Integer
```

Вычисление определенного интеграла

Подынтегральная функция:  $f(x) = x^2 + 5x + 7$

Исходные данные		Решение			
		Результат	Число проходов	Ошибка	
Нижний предел	1	Метод левых прямоугольников	129.3136	12	2.543131E-04
Верхний предел	5	Метод средних прямоугольников	129.3325	4	8.290609E-04
Число отрезков	10	Метод Симпсона	129.3334	2	1.017253E-06
Требуемая точность	0.001				

Вычислить      Выход

Рис. 3.10.2. Форма для вычисления определенного интеграла

```
Private Function Integ1(x)
' подынтегральная функция
  y = x ^ 2 + 5 * x + 7
  Integ1 = y
End Function
```

```
Private Sub cmdCalc_Click()
  a = Val(Text1(0).Text)
  b = Val(Text1(1).Text)
  e = Val(Text1(3).Text)
' метод средних прямоугольников
  n = Val(Text1(2).Text)
  S = 0: R = 1 + e: k = 0
  While Abs(R) > e
    S1 = S: S = 0
    h = (b - a) / n
    x = a + h / 2
    For i = 1 To n
      y = Integ1(x)
      S = S + y * h
      x = x + h
    Next i
    R = (S - S1) / 3
    n = n * 2: k = k + 1
  Wend
  Label5(1) = Str(S): Label6(1) = Str(k): Label8(1) = Str(R)
' метод левых прямоугольников
  n = Val(Text1(2).Text)
  S = 0: R = 1 + e: k = 0
  While Abs(R) > e
    S1 = S: S = 0:
```

```

h = (b - a) / n
x = a
For i = 1 To n
    y = Integ1(x)
    S = S + y * h
    x = x + h
Next i
R = (S - S1) / 3
n = n * 2: k = k + 1
Wend
Label5(0) = Str(S): Label6(0) = Str(k): Label8(0) = Str(R)
' метод Симпсона
n = Val(Text1(2).Text)
S = 0: R = 1 + e: k = 0
While Abs(R) > e
    S1 = S: S = 0
    h = (b - a) / n
    x = a + h: S11 = 0: S12 = 0
    For i = 1 To n - 1
        If i Mod 2 = 0 Then S11 = S11 + Integ1(x)
        If i Mod 2 <> 0 Then S12 = S12 + Integ1(x)
        x = x + h
    Next i
    S = (Integ1(a) + Integ1(b) + 2 * S11 + 4 * S12) * h / 3
    R = (S - S1) / 15
    n = n * 2: k = k + 1
Wend
Label5(2) = Str(S): Label6(2) = Str(k): Label8(2) = Str(R)
End Sub
-----
Private Sub cmdExit_Click()
    Unload Me
End Sub
-----
Private Sub Form_Load()
    Label9(1).Caption = "f(x)= x ^ 2 + 5 * x + 7"
End Sub

```

### Контрольные вопросы

1. В чем заключается метод численного интегрирования?
2. Изобразите схему алгоритма вычисления определенного интеграла.
3. Поясните принцип работы программы по схеме алгоритма.
4. В чем заключается метод двойного прохода при вычислении определенного интеграла с заданной точностью?
5. В чем отличие методов левых, правых и средних прямоугольников?
6. Поясните формулу Симпсона для вычисления определенного интеграла.

### 3.11. Решение алгебраических и трансцендентных уравнений

**Цель работы:** приобрести навыки решения алгебраических и трансцендентных уравнений численными методами.

**Время:** 2 часа.

**Литература:** Л1 с.187 - 204, Л2 с. 204 – 217.

#### 3.11.1. Общие сведения

Задача решения линейных и нелинейных уравнений численными методами разделяется на две самостоятельных задачи: отделение корней и уточнение значений корней на отрезках отделения. При программировании эти две задачи можно решать одновременно.

Алгоритм отделения корней уравнения сводится к табулированию функции на заданном отрезке области определения функции с достаточно большим шагом. Выбор шага зависит от самой функции: монотонно изменяющаяся функция или периодическая функция. На каждом шаге табулирования значение функции вычисляется при текущем значении  $x$  и при  $x+dx$  и проверяется, меняет функция знак на данном отрезке или нет с помощью условного выражения  $f(x)*f(x+dx)<=0$ . Если условие выполняется, то на отрезке  $[x, dx]$  имеется корень, а если не выполняется, то корня на данном отрезке нет. Так как корней может быть несколько, то их желательно запомнить в массиве, например:

```
N=Int((b-a)/dx)+1: Dim A(N,1) ' N – число шагов при табулировании функции
x=a
For i=1 TO N
  y=f(x)
  y1=f(x+dx)
  k=0 ' k – счетчик числа корней
  If y*y1<=0 Then
    Print x, x+dx:
    A(k,0)=x: A(k,1)=x+dx: k=k+1 ' запоминание границ отрезков отделения
  End If
  x=x+dx
Next i
```

Уточнение значения корней проводится одним из способов одномерной поисковой оптимизации. Простейший из них – метод деления отрезка пополам.

Пусть  $p=x: q=x+dx$

```
y=f(p)
c=(p+q)/2
y1=f(c)
While y1>eps
```

```
  If y*y1<0 Then
    q=c
  Else
    p=c: y=y1
```

```
  End If
Wend
c=(p+q)/2: y1=f(c)
Print "Корень =",c
```

Эту процедуру можно оформить как функцию пользователя и вызывать в цикле отделения корней уравнения каждый раз, когда находится отрезок, содержащий корень.

### 3.11.2. Задание

1. Изучите задание и рекомендованную литературу.
2. Отделите корни уравнения согласно варианту задания, результаты запишите в массив.
3. Напишите программу уточнения значений корня на отрезках отделения, извлекая данные из массива.
4. Оформите процедуру уточнения значения корня в виде внешней функции пользователя.
5. Решите задачу отделения и уточнения значения корней по п.2 без предварительной записи данных в массив.

#### Варианты заданий

N	Первое уравнение	Отрезок	Второе уравнение	Отрезок
1	$2^x - (x+1)^3 = 0$	-0.5, 0.5	$x - 2\sin(0.5+x) = 0$	0, 0.5
2	$2^x - \ln x - 7 = 0$	1, 4.5	$x - \cos x = 0$	0.5, 1
3	$2x - 1, 3^x = 0$	0.5, 1	$2x^{1/2} \cos(\pi x/2) = 0$	0, 0.5
4	$2^x - 2, 2x = 0$	0.5, 1	$x^2 + x e^x - 2 = 0$	-0.5, 0
5	$(x-1)^{1/2} - 1/x = 0$	1, 2	$x^2 - 4\sin x = 0$	1.5, 3
6	$e^x - e^{-3x} - 4x = 0$	1, 0	$3x + \cos x + 1 = 0$	-1.0, 0
7	$x^3 + 2, 2x + 2^x = 0$	-0.5, 0	$2x + \sin x + 1 = 0$	-1, 0
8	$e^x - 6x - 3 = 0$	-0.5, 0	$3x - \cos x - 1 = 0$	2, 0
9	$2^x - 4x = 0$	0, 0.5	$x^2 + 0.5\sin x - 2 = 0$	1, 1.5
10	$x2^x = 1 - \sin x$	0, 0.5	$2 - x - \ln x = 0$	1.5, 2
11	$3 \sin x^{1/2} + 0.35x - 0.5$	2, 3	$0.25x^3 + x - 1.2 = 0$	0.2, -3.8
12	$x + x^{1/2} + x^{1/3} - 2.5 = 0$	0.4, 1	$x - 1/(3 + \sin 3.6x) = 0$	0, 0.85
13	$0.1x^2 - x \ln x = 0$	1, 2	$3x - 4 \ln x - 1 = 0$	2, 4
14	$(1 - 0.4x^2)^{1/2}$	0, 1	$e^x + \ln x - 10x = 0$	3, 4
15	$3x - 14 + x^x - e^x = 0$	1, 3	$(1-x)^{1/2} - \operatorname{tg} x = 0$	0, 1

#### Контрольные вопросы

1. Что понимается под отделением корня?
2. Что понимается под уточнением значения корня?
3. В чем заключается метод дихотомии (деления отрезка пополам)?
4. Как выполнить отделение корней уравнения?
5. Начертите схему алгоритма уточнения значения корня на отрезке отделения методом деления отрезка пополам.

### 3.12. Интерполирование функций

**Цель работы:** ознакомиться с методами интерполяции. Приобрести навыки вычисления многочленов по схеме Горнера.

**Время:** 2 часа.

**Литература:** Л1 с. 210 - 213, Л2 с. 226 – 228.

#### 3.12.1. Общие сведения

Интерполяция часто используется для определения значения функции в точках, отличных от узлов интерполяции. Результат интерполяции зависит от выбора интерполирующей функции.

Наиболее часто используются линейная интерполяция, интерполяция методом Лагранжа и методом Ньютона.

Формула для линейной интерполяции:

$$y(x_0, x_1) = y_0 \frac{x - x_1}{x_0 - x_1} + y_1 \frac{x - x_0}{x_1 - x_0}$$

**Программа для интерполяции методом Лагранжа, метод Эйткена:**

Дано: N — количество интервалов интерполяции;

A — значение аргумента;

X(N), F(N) — массивы значений аргумента и функции;

Результат: значение функции - F1.

**Текст программы**

```
Option Explicit
```

```
Dim N As Integer, A As Single, X() As Single, F() As Single
```

```
Dim i As Integer, j As Integer, F1 As Single
```

```
Private Sub Form_Click()
```

```
N = Val(InputBox("Укажите число интервалов интерполяции "))
```

```
ReDim X(N) As Single, F(N) As Single
```

```
For i = 0 To N
```

```
    X(i) = Val(InputBox("Укажите значения X в " & Str(i) & " узле"))
```

```
    F(i) = Val(InputBox("Укажите значения F в " & Str(i) & " узле"))
```

```
Next i
```

```
A = Val(InputBox("Укажите значение аргумента"))
```

```
Rem интерполяция полиномом Лагранжа по Эйткену
```

```
For j = 0 To N - 1
```

```
    For i = j + 1 To N
```

```
        F(i) = ((A - X(j)) * F(i) - (A - X(i)) * F(j)) / (X(i) - X(j))
```

```
    Next i
```

```
Next j
```

```
F1 = F(N)
```

```
MsgBox "Значение функции " & Str(F1)
```

```
End Sub
```

**Программа для интерполяции методом Ньютона:**

Дано: N - степень полинома;

X(N), F(N) — массивы x, y.

A — значение аргумента

Результат: F1— значение полинома в точке A.

**Текст программы.**

```
Private Sub Form_Click()
```

```
'Интерполяция по Ньютону
```

```
N = Val(InputBox("Укажите степень полинома"))
```

```

ReDim X(N) As Single, F(N) As Single
For i = 0 To N
    X(i) = Val(InputBox("Укажите значения X в " & Str(i) & " узле"))
    F(i) = Val(InputBox("Укажите значения F в " & Str(i) & " узле"))
Next i
A = Val(InputBox("Укажите значение аргумента"))
Rem подпрограмма интерполяции по Ньютону
F1 = F(0): S = 1
For i = N To 1 Step -1
    i1 = N - i
    For k = 0 To i - 1
        F(k) = (F(k + 1) - F(k)) / (X(k + 1 + i1) - X(k))
    Next k
    S = S * (A - X(i1))
    F1 = F1 + F(0) * S
Next i
MsgBox "Значение функции " & Str(F1)
End Sub

```

### Вычисление многочлена по схеме Горнера

Пусть дан многочлен  $P(x) = 2x^5 + 4x^3 - 7x^2 - 8$ . Требуется вычислить значение многочлена при  $x = C = 3,78321$

Алгоритм вычисления многочлена по схеме Горнера реализует рекуррентную формулу

$$B_i = A_i + B_{i-1} * C,$$

где  $B_i$  – элемент вспомогательного массива,  $A_i$  – элемент массива коэффициентов при неизвестных. Коэффициент при старшем члене многочлена имеет индекс 0 –  $A_0$ ,  $B_0 = A_0$ .

Порядок работы:

– объявить массив  $A(n)$ , где  $n$  – степень многочлена и ввести в него коэффициенты многочлена, начиная со старшего ;

– присвоить  $B_0 = A(0)$ ;

– организовать цикл от  $i = 1$  до  $n$ ;

– в теле цикла вычислять выражения:

$$B_n = A(i) + B_0 * C;$$

$$B_0 = B_n$$

– при выходе из цикла получим результат:  $P(c) = B_n$ .

### Текст программы:

```

Private Sub Form_Click()
    n = Val(InputBox("Укажите степень полинома"))
    ReDim a(n) As Single
    For i = 0 To n
        a(i) = Val(InputBox("Введите " & Str(i) & " коэффициент полинома"))
    Next i
    C = Val(InputBox("Укажите значение аргумента"))
    B0 = a(0)

```

```

For i = 1 To n
    Bn = a(i) + B0 * C
    B0=Bn
Next i
MsgBox "Значение полинома P(" & Str(C) & ")=" & Str(Bn)
End Sub

```

### 3.12.2. Задание

1. Изучите рекомендованную литературу.
2. Вычислите значение функции в произвольной точке между узлами интерполяции согласно варианту задания тремя способами: с помощью линейной интерполяции, методом Лагранжа и методом Ньютона. Сравните результаты вычислений.
3. Вычислите многочлен согласно варианту задания по схеме Горнера.
4. Изобразите схему Горнера.

#### Варианты заданий

##### 1. К пункту 2.

Вариант	x	0,000	0,500	1,000	1,500	2,000	2,500	3,000	3,500	4,000	4,500
1	y	0,067	0,079	0,264	0,510	0,692	0,781	1,241	1,439	1,603	1,666
2	y	0,114	0,328	0,333	0,524	0,555	1,462	1,532	1,837	2,044	2,447
3	y	0,101	0,347	0,494	0,918	0,970	1,427	1,576	1,905	2,088	3,612
4	y	0,380	0,429	0,454	0,490	1,118	1,274	1,435	1,697	2,010	2,042
5	y	0,135	0,145	0,158	0,201	0,423	0,426	0,939	1,298	1,358	1,973
6	y	0,045	0,294	0,367	0,418	0,684	1,086	1,415	2,007	2,066	2,266
7	y	0,093	0,350	0,357	0,357	0,517	0,772	1,075	1,339	1,844	2,267
8	y	0,480	0,650	0,756	0,860	1,136	1,141	1,393	1,487	1,533	2,581
9	y	0,167	0,454	0,959	1,014	1,143	1,717	1,799	1,823	1,881	2,041
10	y	0,879	0,992	1,135	1,238	1,660	1,788	1,876	2,094	2,402	2,498
11	y	0,333	0,341	0,648	0,662	1,327	1,356	1,426	1,897	1,933	3,820
12	y	0,307	0,451	0,579	0,693	0,974	1,552	1,778	1,850	1,934	2,307
13	y	0,039	0,442	0,841	0,884	1,088	2,023	2,109	2,206	2,361	2,369
14	y	0,184	0,484	0,549	0,750	0,770	0,955	1,616	1,772	2,253	2,535
15	y	0,078	0,545	0,584	0,817	1,049	1,128	1,728	1,789	2,360	2,860

##### 2. К пункту 3

Вариант	Многочлен	C
1	$P(x)=2x^7+5x^6+3x^5-x^4+5x^3-4x^2+x-8$	1,754
2	$P(x)=3x^6-5x^5+3,75x^3-4,18x^2-8,54$	3,562
3	$P(x)=1,75x^7+2,43x^5+6,75x^4-8,54x^2+2,31x-3,82$	2,439
4	$P(x)=2,56x^6-6,51x^4+5,78x^3+1,45x^2-3,65x$	4,075
5	$P(x)=5,25x^7-1,78x^6-0,97x^4+5,34x^3-x-8$	3,284
6	$P(x)=0,345x^7-3x^5+1,28x^4-5x^3+9,24x^2+1,26x$	1,185
7	$P(x)=-5x^6+2,13x^5+1,39x^3-7,42x^2-5,64$	2,821

8	$P(x)=1,93x^7+5,83x^6+1,23x^5-4,91x^4+4,35x^3+5,49x-4,28$	3.643
9	$P(x)=1,73x^7-3x^5+9,75x^4+5,26x^3-4x^2-5,98$	3.125
10	$P(x)=0,38x^7+1,53x^6-3,57x^5+9,03x^4-4,26x^2+1,29x-1,68$	2.769
11	$P(x)=2,35x^6-3,57x^4+5,57x^3-4,34x^2+2,47x-7,88$	2.594
12	$P(x)=1,32x^7-1,35x^6+3,46x^5-4x^2+3,75x-8,02$	1.861
13	$P(x)=2,49x^7+4,5x^6-3x^5+5x^3+4,18$	4.378
14	$P(x)=1,12x^7-1,25x^6+3,86x^5-4,93x^4+7,85x^3-6,4x^2$	3.549
15	$P(x)=-5x^6+7,52x^4+5,08x^3-5,24x^2+6,09x-8,48$	2.276

### Контрольные вопросы

1. Что такое интерполирование и интерполяция?
2. Какими методами осуществляется поиск значения функции в точках, отличных от узлов интерполяции?
3. Поясните принцип интерполяции функции графически.
4. Как вычислить многочлен при заданном значении  $x$ ?
5. В чем достоинство схемы Горнера по сравнению с вычислением по формуле?

### 3.13. Обработка символьных переменных

**Цель работы:** ознакомиться с основными функциями для обработки строк символов.

**Время:** 4 часа.

**Литература:** Л1 с.213 - 217, Л2 с. 246 – 252.

#### 3.13.1. Общие сведения

Ввод строки символов осуществляется с помощью оператора Let, окна ввода TextBox и функции InputBox. При использовании оператора Let строковая переменная заключается в кавычки.

Переменные символьного типа объявляются явно операторами Dim или Public с опцией String, суффиксом "\$" и неявно – оператором DefStr:

Dim C As String – строка переменной длины: 65535 символов при 16-и разрядном коде и до  $2 \times 10^{32}$  при 32-х разрядном коде.

Dim C As String\*длина – строка постоянной длины.

DefStr C, S – все переменные имя которых начинается с "C" или "S", будут переменными строкового типа.

C\$ - тип переменной объявлен с помощью суффикса. Тип переменной указывается только при первом обращении к переменной.

Основные функции для обработки строк символов:

Len(C) – определение длины строки.

Left\$(C,N) – выделение N символов из строки C слева.

Right\$(C,N) – выделение N символов из строки C справа.

Mid\$(C,N1,N2) – выделение N2 символов с N1 позиции из строки C.

Mid\$(C1,N1,N2) = C2 – замена N2 символов с позиции N1 в строке C1 строкой C2.

InStr([I],C1,C2) – поиск вхождения строки C2 в строку C1. Поиск начинается с позиции I. Если I опущено, то поиск начинается с первой позиции. Функция ищет первое вхождение строки C2 в строку C1 и возвращает номер позиции. Если вхождение не найдено, то результат равен нулю.

LCase\$(C) - переводит прописные буквы в строчные.

UCase\$(C) - переводит строчные буквы в прописные.

При обработке строки символов она рассматривается как одномерный массив. Сравнение символов осуществляется по их номеру в кодовой таблице ASCII.

### 3.13.2. Задание

1. Объявите переменные C, C1 и C2 как переменные символьного типа.

2. Введите строку символов C1 = " В Лесу родилась елочка. Лес рубят - щепки летят".  
C2="Ночевала тучка золотая на груди утеса великана".

3. Определите длину строк C1 и C2.

4. Сформируйте из переменных C1 и C2 фразу "Золотая тучка ночевала в лесу". Алгоритм решения задания:

– переведите строку C1 и C2 в нижний регистр;

– найдите и выделите из переменной C1 слова "в лесу";

– найдите и выделите из строки C2 слова "золотая", "ночевала" и "тучка";

– сформируйте из полученных слов заданную фразу;

– переведите первый символ полученной фразы в верхний регистр.

5. Подсчитайте число гласных в слове (слово выбрать произвольно).

Алгоритм решения задачи:

– введите переменную-счетчик числа гласных;

– введите вспомогательную переменную, содержащую одни гласные;

– в цикле выделять из заданного слова одну переменную и осуществить поиск вхождения ее в вспомогательную переменную, если текущая переменная имеется в вспомогательной переменной, значит она гласная, добавьте единицу к переменной-счетчику.

6. Разделите переменную C1 на строки по точке.

7. Разделите переменную C2 на строки по пробелу.

8. Соедините переменные C1 и C2.

9. Разделите полученный текст на строки длиной не более 30 символов.

### Примеры выполнения заданий

**Пример 3.13.1.** Разделить фразу "во поле береза стояла" на слова по пробелу.

**Решение.**

```
Private Sub Form_Click()
```

```
Cl
```

```
' c - строка символов
```

```
' cd - вспомогательная переменная
```

```
' c1 - символ разделитель
```

```
' p - позиция вхождения символа разделителя
```

```
c = "во поле береза стояла"
```

```
cd = c
```

```
c1 = " "
```

```
m21:
```

```
p = InStr(cd, c1) 'поиск вхождения символа разделителя
```

```

If p = 0 Then
    Print cd: Exit Sub
Else
    c3 = Left$(cd, p): Print c3 'выделение текста слева
    cd = Right$(cd, Len(cd) - p) ' выделение текста справа
    GoTo m21
End Sub

```

**Пример 3.13.2.** Разделить фразу "У лукоморья дуб зеленый, золотая цепь на дубе том." на строки длиной не более 14 символов.

**Решение.**

```

Private Sub Form_Click()
' Деление строки на строки заданной длины
Cls
'c1 = InputBox("Введите строку текста")
c1 = "У Лукоморья дуб зеленый, золотая цепь на дубе том."
'c2 = InputBox("Введите символ-разделитель")
c2 = " "
'lc = Val(InputBox("Укажите длину строки"))
lc = 14
p = 1 ' текущее значение указателя
p1 = 1 ' предыдущее положение указателя
cd = c1
m31:
If Len(cd) <= lc Then Print cd: Exit Sub
m32:
p = InStr(p1 + 1, cd, c2)
If p > 0 And p < lc Then
    p1 = p: GoTo m32
Elseif p = lc Then
    c3 = Left$(cd, p): Print c3
    cd = Right$(cd, Len(cd) - p)
    p1 = 0
    GoTo m31
Elseif p > lc Then
    c3 = Left$(cd, p1): Print c3
    cd = Right$(cd, Len(cd) - p1)
    p1 = 0
    GoTo m31
Elseif p = 0 Then
    c3 = Left$(cd, lc) + "-": Print c3
    cd = Right$(cd, Len(cd) - lc)
    GoTo m31
End If
End Sub

```

### Дополнительные задания

1. Проверить сбалансированность открывающих и закрывающих скобок в формуле  $(3x + \sin(x))/(2x-1)$ .

**Примечание.** Скобки сбалансированы, если число открывающих скобок равно числу закрывающих скобок.

2. Подсчитать число символов "а" в слове.
3. Посчитать число гласных букв в слове.
4. Заменить слово "месяц" на "терем" во фразе "отражался месяц в пруду".
5. Определить частоту повторения символов в заданном предложении.

### Контрольные вопросы

1. Какие группы функций для работы с символьными переменными Вам известны?
2. Как найти вхождение заданного слова в предложение?
3. Каким образом программа сравнивает символьные переменные?
4. Для какой цели используется функция MID\$(C,n,n1)?
5. Как подсчитать сбалансированность открывающих и закрывающих скобок?
6. Как определить, является символ гласной или согласной?
7. Как преобразовать символы из верхнего регистра в нижний регистр и наоборот?
8. Как подсчитать частоту повторения символов в предложении?

## 3.14. Разработка меню пользователя

**Цель работы:** закрепить знания в разработке простых форм. Приобрести навыки в разработке меню пользователя, создании родительских (MDI-форм) и дочерних форм.

**Время** – 4 часа.

**Литература:** Л1 с. 293- 297, Л3 с.50-55.

### 3.14.1. Общие сведения

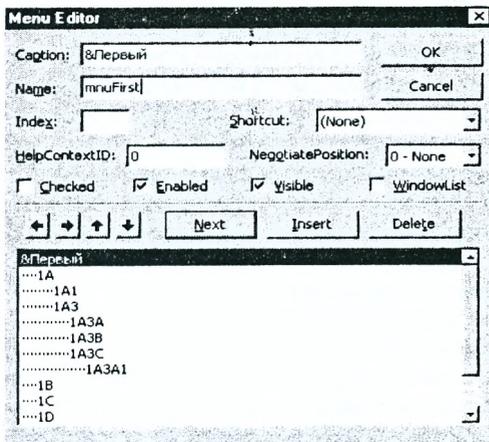


Рис. 3.14.1. Редактор меню Menu Editor

Разработка меню позволяет сделать приложение с более дружелюбным интерфейсом. VB позволяет иметь до шести уровней вложенности меню. Большое количество уровней тоже не совсем удобно. Рекомендуется использовать не более 3-х уровней вложенности. VB позволяет создавать иерархические меню. Первый уровень – горизонтальное меню (строка). Второй и последующие уровни – всплывающие меню.

Visual Basic 6.0 имеет удобное средство для разработки меню – **Menu Editor**, которое вызывается командой **Tools / Menu Editor** или комбинацией клавиш **CTRL – E**.

Диалоговое окно *Menu Editor* приведено на рис. 3.14.1.

Строка ввода ***Caption*** служит для ввода наименования пункта меню, выводимого на экран. После нажатия клавиши *Enter* или щелчка мыши введенное наименование появляется в окне редактора. Если перед одним из символов наименования пункта меню поставить символ "&" - амперсанд, то появится возможность вызывать пункт меню по нажатию комбинации клавиш [*Alt* + клавиша]. Символ, перед которым стоит знак "&", подчеркивается.

Строка ввода ***Name*** служит для ввода имени пункта меню, которое будет использоваться в программе для обработки событий. Перед именем пункта меню рекомендуется ставить префикс *mnu*, например, *mnuFile*. Программа не позволит пользователю выйти из редактора, пока всем пунктам меню не будут присвоены имена.

Окно ***Index*** используется в том случае, если имеется несколько пунктов меню с одинаковыми именами или надо сделать пункты меню частью массива элементов управления.

Окно ***Shortcut*** позволяет назначить каждому пункту меню комбинацию клавиш для быстрого вызова команд меню: *Ctrl* + клавиша, *Shift* + клавиша и др. При открытии списка появится список быстрых клавиш, из которого надо выбрать нужный.

Окно ***HelpContextID*** обеспечивает ввод идентификатора, который используется в электронной справочной системе для выдачи контекстно-зависимой справки по вашему приложению.

Окно ***NegotiatePosition*** – служит для определения способа отображения меню на экране, когда один из связанных объектов приложения активен: не показывать, слева, справа, по центру.

Флажок ***Checked***. Если значение данного свойства равно *True*, то возле соответствующего пункта меню появляется галочка. Это сигнализирует о том, что соответствующий параметр выбран.

Флажок ***Enabled***. Определяет доступность данного пункта меню. Если его значение *False*, то пункт меню недоступен.

Флажок ***Visible***. Данное свойство определяет, будет ли виден на экране соответствующий пункт меню. При разработке приложения можно предусмотреть несколько наборов меню, которые должны появляться на экране в соответствующие моменты времени. Например, если в приложении не открыто ни одно окно, меню *Window* не должно появляться на экране.

Флажок ***WindowList***. Данный флажок служит для автоматического формирования списка открытых окон, являющихся элементами многодокументного интерфейса (*MDI*). Установка этого флажка для элемента управления меню верхнего уровня приведет к тому, что в данном элементе будет автоматически формироваться динамический список всех активных дочерних окон.

Кнопка ***Next*** предназначена для добавления новых пунктов меню.

Кнопка ***Insert*** позволяет вставить поле для ввода нового пункта меню.

Кнопка ***Delete*** служит для удаления выделенного пункта меню.

Пункты меню, близкие по назначению целесообразно группировать, отделяя их от других пунктов меню горизонтальной чертой – разделительная линия (*Separator bar*). Эта черта создается так же, как и другие элементы управления меню, но вместо наименования пункта меню (свойство *Caption*) вводится дефис (-). Имя данному пункту меню можно присвоить произвольно.

Изменение уровня вложенности элементов меню осуществляется с помощью кнопок ➡ и ⬅. Первая кнопка понижает уровень, вторая – повышает.  
Кнопки ⬆ и ⬇ служат для перемещения выделенного пункта меню. Уровень вложенности элемента управления при этом не изменяется.

### Понятие о MDI-форме

*MDI-форма - это многодокументный интерфейс, предназначенный для организации взаимодействия нескольких независимых форм.*

MDI-форма является родительской формой или контейнером для других (дочерних) форм. **Создание MDI-формы** осуществляется командой **Project, Add MDI Form**.

При запуске программы с MDI-формой программа автоматически устанавливает размеры дочерних окон, которые могут оказаться меньше, чем при настройке, и поэтому часть объектов активной формы может быть невидимой. Чтобы избавиться от этого недостатка, необходимо в обработчике события Load каждой формы явно указать размеры и положение формы в окне, например:

```
Private Sub Form_Load()  
    Me.Height = 2745  
    Me.Width = 3090  
    Me.Top = (MDIForm1.ScaleHeight - Me.Height) / 2  
    Me.Left = (MDIForm1.ScaleWidth - Me.Width) / 2  
End Sub
```

Если одна из форм максимизируется, то и все последующие открываемые форму будут развернуты на все окно. Чтобы этого не происходило, необходимо при закрытии максимизированной формы приводить ее размеры в нормальное состояние. Для этого в обработчик события кнопки ВЫХОД формы надо поместить следующий код:

```
Private Sub mnuExit_Click()  
    Me.WindowState = 0  
    Unload Me  
End Sub
```

### Работа с дочерними формами

Чтобы обычная форма стала подчиненной MDI-форме (дочерней формой), значение ее свойства MDIChild необходимо установить в True.

Дочерние формы показываются с помощью метода Show, например:

```
Private Sub mnuVvod_Click()  
    frmVvod.Show  
End Sub
```

### Управление открытыми формами:

- установите флажок WindowList для элемента меню Window (окно);
- введите меню второго уровня в меню Window: каскадом, горизонтально, вертикально. Управление окнами осуществляется с помощью метода **Arrange**. Кроме того, VB имеет четыре константы для управления окнами: VbCascade – размещение окон каскадом, VbTileHorizontal – размещение окон горизонтально, VbTileVertical - размещение окон вертикально, VbArrangeIcons – пиктограммы всех минимизированных окон распола-

гаются по нижнему краю родительской формы. Последнее свойство реализуется с помощью кнопки системного меню Минимизация:

– поместите в обработчики события Click этих пунктов меню объектный код следующего вида:

имяMDI-формы.Arrange константаVisualBasic

Например:

```
Sub mnuCascade_Click()  
    MDIForm1.Arrange VbCascade  
End Sub  
Sub mnuHorizontal_Click()  
    MDIForm1.Arrange VbTileHorizontal  
End Sub  
Sub mnuVertical_Click()  
    MDIForm1.Arrange VbTileVertical  
End Sub
```

Чтобы указать форму, которая должна стартовать первой, щелкните правой кнопкой мыши в окне проекта и выберите из контекстного меню команду Project Properties. На вкладке General в окне Startup Object выберите имя стартовой формы.

### 3.14.2. Задание

1. Разработайте форму для вычисления параметров геометрической фигуры по указанию преподавателя.
2. Добавьте в проект MDI-форму и разработайте меню пользователя для управления формами.

#### Дополнительное задание

Разработайте контекстное меню для управления размером и начертанием шрифта в форме (см. Л1 раздел 4.5.6).

#### Указания к выполнению задания

Загрузите программу лабораторной работы № 3.

Добавьте новую форму командой Project, Add Form.

Разработайте в новой форме пользовательский интерфейс для вычисления параметров другой геометрической фигуры (площади поверхности и объема).

Добавьте в проект MDI-форму командой Project, Add MDI Form.

Присвойте свойству MDIChild дочерних форм значение True.

Разработайте в MDI форме меню пользователя.

Возможная структура меню:

Файл

Создать

Открыть

Сохранить

-----

Печать

-----

Выход

Вычисление

Параллелепипед

Площадь

Объем

Цилиндр

Площадь

Объем

Шар

Площадь

Объем

Помощь

О программе

Автор

-----  
Информация

7. Добавьте в главное меню пункт "Окно", настройте его на автоматическое формирование списка открытых окон, напишите программу для управления окнами;

8. Сделайте MDI-форму стартовой: Project, Properties, выбрать в списке StartUp Objekt MDI-форму.

9. Распечатайте текст программы родительской формы.

### **Контрольные вопросы**

1. Что такое родительская форма?
2. Что такое дочерняя форма?
3. Как добавить в проект простую форму, MDI форму?
4. Как установить клавишу ускоренного вызова программы?
5. Как установить стартовую форму?
6. Как разработать меню пользователя?
7. Чем отличается порядок разработки контекстного меню от порядка разработки обычного меню?

## **3.15. Работа с массивами элементов управления**

**Цель работы:** изучить способы создания массивов элементов управления и их использования для вывода информации.

**Время:** 4 часа.

**Литература.** Л1 с.271 - 274, Л3 с. 83 – 86.

### **3.15.1. Общие сведения**

Массив элементов управления представляет собой группу элементов управления одного типа, имеющих одно и то же имя и отличающихся индексом. Обычно его создают на этапе разработки программы. В большинстве случаев для этого достаточно создать массив из одного элемента следующим образом:

- 1) поместите элемент управления на форму;

- 2) задайте свойство Name;
- 3) свойству Index присвойте значение 0.

Для добавления элемента управления во время выполнения программы используются методы Load и Add. Метод Load позволяет добавлять элементы управления к существующему элементу управления, метод Add позволяет не только добавлять новые элементы управления, но и создавать новые элементы. Метод Add на занятии не исследуется. Синтаксис метода Load:

Load <имя\_элемента управления(индекс)>,

где индекс – номер загружаемого элемента. Для вновь созданного элемента управления свойство Visible равно False, свойства Top и Left имеют такие же значения, что и у родительского элемента управления.

Например, для создания 10 элементов управления Label2 может использоваться следующая программа:

```
For i=1 to 10
    Load Label2(i)
    Label2(i).Visible=True
    Label2(i).Top=Label2(i-1).Top+Label2(0).Height
Next i
```

Доступ к конкретному элементу управления в массиве производится по его имени и индексу, например, Label2(5).Caption="Привет!".

Для удаления элементов управления используется метод Unload:

Unload <имя элемента управления(индекс)>

Нулевой элемент управления не должен удаляться.

### 3.15.2. Задание

1. Разработайте форму для добавления и удаления элементов управления на форму.
2. Составьте таблицу значений функции согласно варианту. Отрезок табулирования и шаг табуляции выберите самостоятельно в области определения функции. Результаты запишите в массив и выведите на экран с помощью массива элементов управления Label.

#### Указания к выполнению задания

##### К пункту 1:

– поместите на форму элемент управления txtText1. Свойству Text присвойте значение Текст0;

- свойству Index присвойте значение 0;
- установите на форму две кнопки cmdAdd (Добавить) и cmdDel (Удалить);
- в обработчик события Click кнопки cmdAdd поместите следующий текст программы:

```
Private Sub cmdAdd_Click()
    Dim i As Single
    i = Text1().Count ' определение номера добавляемого
                    ' элемента управления
    Load Text1(i) ' загрузка элемента управления (ЭУ)
    Text1(i).Text = "Текст" & Str$(i) 'присвоение значения новому ЭУ
    Text1(i).Top = Text1(i - 1).Top + Text1(0).Height
                    ' перемещение нового элемента управления
```

```
Text1(i).Visible = True ' сделать новый ЭУ видимым
End Sub
```

- в обработчик события Click кнопки cmdDel поместите следующий текст программы:

```
Private Sub cmdDel_Click()
    Dim i As Single
    i = Text1().Count - 1 ' определение номера удаляемого ЭУ
    If i > 0 Then
        Unload Text1(i) ' удаление последнего ЭУ
    End If
End Sub
```

## К пункту 2:

- установить на форму метки и окна ввода для ввода начального (Xнач), конечного (Xкон) значений аргумента и шага табуляции (dX) (рис. 3.15.1);

- установить две кнопки: cmdCalc (Вычисление), cmdExit (Выход);

- установите две метки lblArgument (X) и lblFunction (Y) для создания массивов элементов управления. Присвойте свойствам Index этих элементов значение 0. Выравняйте текст по центру;

- введите текст программы:

```
Option Explicit
Dim x As Single, y As Single, dx As Single
```

```
Private Sub cmdCalc_Click()
    Dim Xnach As Single, Xkon As Single, dx As Single
    Dim i As Integer, n As Integer
    Xnach = Val(txtXnach.Text)
    Xkon = Val(txtXkon.Text)
    dx = Val(txtDx.Text)
    n = Int((Xkon - Xnach) / dx) + 1
    x = Xnach
    For i = 1 To n
        ' Загружаем элементы управления
        Load lblArgument(i): Load lblFunction(i)
        ' Вычисляем значение функции
        y = Sin(x)
        ' Настраиваем параметры добавленных элементов управления
        lblArgument(i).Top = lblArgument(i - 1).Top + lblArgument(0).Height
        lblFunction(i).Top = lblFunction(i - 1).Top + lblFunction(0).Height
        lblArgument(i).Caption = Str$(Round(x, 2))
        lblFunction(i).Caption = Str$(Round(y, 2))
        lblArgument(i).Visible = True: lblFunction(i).Visible = True
        x = x + dx
    Next i
End Sub
```



Рис. 3.15.1. Табулирование функции с использованием массивов элементов управления

Для ввода исходных данных (начала и конца отрезка табулирования, числа шагов или шага табулирования) используйте окна ввода или функцию InputBox.

### Варианты заданий

1	$\log_3(3x - 11)$	2	$\frac{\log_7(x^3)}{\arcsin(x^2)}$
3	$\frac{\arcsin(x^2 + 0,5)}{\log_5(x^5)}$	4	$\frac{\sqrt{\log_{15}(x + 12,3)}}{\arcsin(x^4)}$
5	$(\arcsin(x) + \arccos(3x)) \times \log_2(x)$	6	$\sqrt{2 \times \arcsin(x) + \frac{1}{x^2}}$
7	$\arcsin(x^2 + 1)^5$	8	$\cos\left(\frac{x}{2}\right) + 3\operatorname{tg}\left(\frac{3\sqrt{3}}{x}\right)$
9	$2^{x^2} \times \arcsin(5x + 3)$	10	$\log_2\left(\frac{\sin(x)}{1 + 3x}\right)$
11	$\log_3(x)^3 - 5\arcsin(x^2)$	12	$\sqrt{\operatorname{arctg}(x) + \frac{1}{x^2}}$
13	$(x - 4)^2 \operatorname{Log}_{0,5}(x - 3) + 1$	14	$e^{-2x} - 2x + 1$
15	$0,5^x - 1 + (x + 2)^2$	16	$(x - 2)^2 2^x - 1$
17	$x^2 - 20\sin(x)$	18	$2 \operatorname{arctg}(x) - 3x + 2$
19	$2x^3 - 9x^2 - 60x + 1$	20	$(x - 3)^2 \log_{0,5}(x - 2)^2 + 1$

### Контрольные вопросы

1. Как создать массив элементов управления на этапе разработки программы?
2. Как добавить элементы управления в процессе работы программы?
3. Как удалить элементы управления с формы?
4. Какими свойствами обладает вновь созданный элемент массива элементов управления?
5. В чем состоят преимущества в использовании массивов элементов управления по сравнению с набором отдельных элементов управления.

## 3.16. Использование сетки MSFlexGrid

**Цель работы:** изучить основные свойства элемента управления MSFlexGrid (сетки) и способы использования ее для вывода информации.

**Время:** 2 часа.

**Литература.** Л1 с.275 - 280, Л3 с. 86 – 90.

### 4.16.1. Общие сведения

Элемент управления MSFlexGrid предназначен для отображения и обработки табличных данных. Он позволяет сортировать, объединять и форматировать таблицы строковых данных. Для добавления сетки MSFlexGrid на панель элементов управления, необходимо в меню Project, Components установить флажок у элемента "Microsoft FlexGrid" и нажать ОК.

Основные свойства объекта MSFlexGrid: Cols, Rows – количество столбцов, строк в сетке; FixedCols, FixedRows – количество фиксированных столбцов, строк в сетке; TextMatrix (строка, столбец) – содержимое ячейки на пересечении строки (Row) и колонки (Col), то же значение имеет свойство Text; RowSel, ColSel – выделение строк и колонок; Sort – сортировка сетки.

События и методы сетки:

EnterCell, – выделение ячейки при щелчке мыши.

LeaveCell - отмена выделения ячейки при щелчке мыши.

AddItem – вставка строк:

MSFlexGrid1.AddItem Item\$[.номер строки]

#### 4.16.2. Задание

1. Разработайте форму для ввода данных в выделенную ячейку и исследуйте свойства сетки MSFlexGrid.

2. Составьте таблицу значений функции на заданном отрезке. Результаты запишите в массив переменных и выведите на экран с помощью сетки.

3. Протабулировать функцию двух переменных с выводом результатов в сетку.

**Указания к выполнению работы**

**К пункту 1:**

- разработайте форму согласно рис. 3.16.1.

- поместите на панель элементов управления сетку: введите команду Project, Components, установите флажок у элемента Microsoft FlexGrid Control 6.0;

- установите сетку на форму. Присвойте ей имя Grid1;

- установите в окне свойств сетки значения требуемых свойств: Rows = 10, Cols = 10, FixedRows = 1, FixedCols = 1;

- установите окно ввода txtText1 и кнопку Ввод (имя – cmdVvod);

- запишите в обработчики событий кнопки и сетки следующий текст программы:

```
Private Sub MSFlexGrid1_SelChange()  
    'выделенная ячейка закрашивается желтым цветом  
    MSFlexGrid1.CellBackColor = QBColor(14)  
End Sub
```

```
Private Sub cmdVvod_Click()  
    ' в выделенную ячейку вводится текст из объекта TextBox  
    MSFlexGrid1.TextMatrix(MSFlexGrid1.RowSel,MSFlexGrid1.ColSel)=  
        txtText1.Text  
End Sub
```

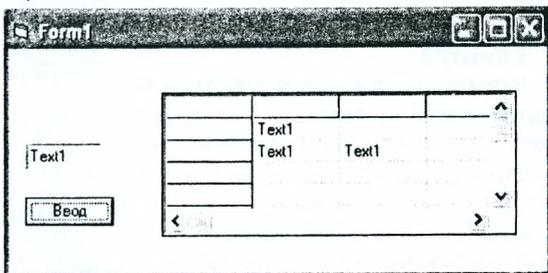


Рис. 3.16.1. Исследование свойств сетки

```
Private Sub MSFlexGrid1_LeaveCell()
    ' обработчик события отмены выделения ячейки, при щелчке мышью
    ' по другой ячейке, ячейка закрашивается белым цветом
    MSFlexGrid1.CellBackColor = QBColor(15)
End Sub
```

```
-----
Private Sub MSFlexGrid1_LostFocus()
    ' очистка ячейки при потере фокуса
    MSFlexGrid1.CellBackColor = QBColor(15)
End Sub
```

## К пункту 2

Данный пункт задания выполняется так же, как и п. 2 в упражнении 3.15, но вывод данных осуществляется в сетку:

- установите на сетку три столбца, определите число шагов табулирования и установите программным путем число строк;
- зафиксируйте первый столбец и первую строку;
- второй столбец используется для вывода значений аргумента, третий – для вывода значений функции.

## К пункту 3

Разработать форму для табулирования функции двух переменных. Данные выводить в сетку и сохранять в массиве для последующего вывода на печать. Поместить на форму строку состояния. В строку состояния выводить динамически число строк и столбцов в массиве и текущее время.

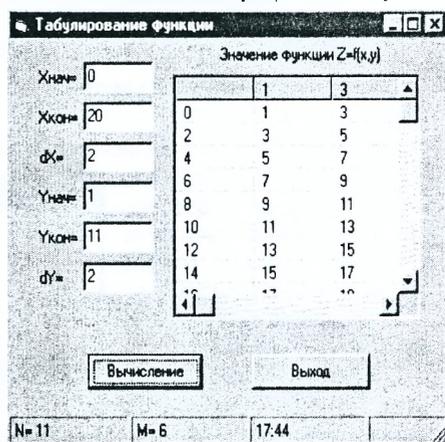


Рис. 3.16.2. Табулирование функции двух переменных

## Порядок работы

Разработайте программу табулирования функции двух переменных.

Разработайте эскиз формы (рис.3.16.2).

Поместите на форму строку состояния:

- введите команду Project, Components и установите флажок у компонента Microsoft Windows Common Controls 6.0(SP3);
- установите на форму элемент управления StatusBar (строка состояния);
- вызовите контекстное меню строки состояния;
- откройте закладку Panels и установите четыре поля: первое и второе текстовые (sbrText), третье – время (sbrTime), четвертое – текстовое (резервное). Все поля в строке состояния имеют индекс (как массивы элементов управления), тип полей устанавливается с помощью свойства Style. Заполнять поля будем программным путем.

Напишите текст программы:

Option Explicit

Dim Xn As Single, Xk As Single, Dx As Single

Dim Yn As Single, Yk As Single, Dy As Single

Dim X As Single, Y As Single

Dim M As Integer, N As Integer

Dim Z() As Single

---

Private Sub Command1\_Click()

Dim i As Integer, j As Integer

Cls

' ввод данных

Xn = Val(Text1(0).Text) : Xk = Val(Text1(1).Text)

Dx = Val(Text1(2).Text) : Yn = Val(Text1(3).Text)

Yk = Val(Text1(4).Text) : Dy = Val(Text1(5).Text)

' определение размерности массива и сетки

$N = \text{Int}((Xk - Xn) / Dx) + 1$

$M = \text{Int}((Yk - Yn) / Dy) + 1$

ReDim Z(N, M)

Grid1.Rows = N + 1

Grid1.Cols = M + 1

' нумерация строк

Grid1.ColAlignment(0) = 3

For i = 1 To N

Grid1.Textmatrix(i, 0) = Str\$(i)

Next i

' нумерация столбцов

For j = 1 To M

Grid1.Col = j

Grid1.ColAlignment(j) = 3

Grid1.TextMatrix(0, j) = Str\$(j)

Next j

' вычисление значения Z

X = Xn

For i = 1 To N

Y = Yn

For j = 1 To M

Z(i, j) = X + Y

Y = Y + Dy

Grid1.TextMatrix(i, j) = Str\$(Round(Z(i, j), "#.##"))

Next j

X = X + Dx

Next i

StatusBar1.Panels(1) = "N=" & Str(N) ' вывод в строку состояния значений

StatusBar1.Panels(2) = "M=" & Str(M) ' числа строк и столбцов

End Sub

---

Private Sub Command2\_Click()

Unload Me

End Sub

Для контроля правильности работы программы составьте исходные данные для тестирования:  $X_{нач}=0$ ,  $X_{кон}=10$ ,  $Dx=2$ ,  $Y_{нач}=1$ ,  $Y_{кон}=11$ ,  $Dy=2$ . Функция  $Z=x+y$ .

Запустите программу и проверьте результаты работы:  $Z(0,0)=1$ ;  $Z(0,1)=3$ ; ...  $Z(6,6)=21$

### Варианты заданий к пункту 2

1	$\ln(x)+(x+1)^3$	2	$x^{2x}-1$
3	$(x+1)^{0.5}-1/x$	4	$x-\cos(x)$
5	$3x+\cos(x)$	6	$x+\ln(x)-0,5$
7	$2-x-\ln(x)$	8	$(x-1)^2-0,5e^x$
9	$(2-x)e^x-0,5$	10	$2^x-0,5(2+x)^{0,5}$
11	$x^2+4\sin(x)$	12	$2x-\lg(x)-7$
13	$5x-8\ln(x)-8$	14	$3x-e^x$
15	$x(x+1)^2-1$	16	$x-(x+1)^3$
17	$2x+\lg(x)+0,5$	18	$\sin(0,5x)+1-x^2$
19	$0,5x+\lg(x-1)-0,5$	20	$2\sin(x-0,6)-1,5+x$

### Варианты заданий к пункту 3

Используйте задания по п. 2, заменив в формуле одну из переменных  $x$  на  $y$ , например, для варианта 1:  $\ln(x)+(y+1)^3$ .

#### Контрольные вопросы

1. Каково назначение элемента управления MSFlexGrid?
2. Как добавить сетку на панель элементов управления?
3. Перечислите основные свойства сетки.
4. Приведите синтаксис команд ввода данных в сетку и вывода данных из сетки в переменные.
5. Как организовать ввод данных в сетку?
6. Какие свойства сетки используются для ввода данных?
7. Как выравнивать данные в столбце по центру (по левому краю, по ширине)?
8. Как выделить ячейку ввода в сетке цветом?
9. Как отменить выделение ячейки сетки?
10. Как зафиксировать строки (столбцы) сетки? Для каких целей это применяется?

## 3.17. Графические объекты

**Цель занятия:** исследовать свойства графических объектов Line и Shape.

**Время:** 2 часа.

**Литература:** Л1 с. 317 - 324, Л3 с. 92 – 99.

### 3.17.1. Общие сведения

В графическом режиме экран видеомонитора представляет собой набор точек, расположенных по строкам. Каждая точка на экране называется пикселем, число точек по горизонтали и вертикали определяет разрешающую способность экрана. Для работы в среде Windows разрешающая способность должна быть не менее 800x600 пикселей.

Для работы с графикой Visual Basic 6.0 имеет графические объекты, графические элементы управления и графические методы.

К **графическим объектам** относятся **форма (Form)** и **графическое окно (Picture Box)**. К этим объектам могут быть применены графические методы.

**Графические элементы управления** – позволяют помещать на графические объекты линии и геометрические фигуры. К ним относятся элементы управления *Line* и *Shape*.

Особо следует выделить элемент управления *Image*. Он не является ни графическим объектом, ни графическим элементом управления, так как не позволяет применять графические методы, но может использоваться для вставки рисунков.

Элемент управления **Line** позволяет рисовать линии различной толщины и стиля.

Этот элемент обладает 15 свойствами. Основными являются:

**X1, Y1** – координаты левого конца линии; **X2, Y2** - координаты правого конца линии.

**BorderStyle** - определяет стиль линии:

0 - невидимая; 1 – сплошная; 2 – пунктирная; 3 – пунктирная с коротким штрихом.

**BorderWidth** - определяет толщину линии и может принимать любые значения кроме нуля.

**BorderColor** - определяет цвет объекта. Существует четыре способа задания цвета:

– непосредственное задание **16-ричной константой**. Например: &H00000000& - черный цвет; &H000080FF& - красный цвет;

– использование **RGB** – функции: RGB (Red, Green, Blue).

RGB – функция формируется из трех цветов: красного, зеленого и синего. Каждый цвет задается числовой константой от 0 до 255. Например:

R=100: G=150: B=75

Line.BorderColor=RGB(R,G,B) 'темно зеленый цвет

– использование **констант Visual Basic**. Имеется 8 констант: *vbBlack* - черный; *vbBlue* - синий; *vbCyan* - голубой; *vbGreen* - зеленый; *vbMagenta* - сиреневый; *vbRed* - красный; *vbWhite* - белый; *vbYellow* – желтый;

– использование функции **QBColor(C)**, где C - номер цвета, принимает значение от 0 до 15 (табл. 3.17.1).

Таблица 3.17.1 Цветовая палитра функции QBColor

Черный	0	Темно – серый	8
Темно-синий	1	Синий	9
Темно-зеленый	2	Зеленый	10
Темно-голубой	3	Голубой	11
Темно-красный	4	Красный	12
Темно-сиреневый	5	Сиреневый	13
Коричневый	6	Желтый	14
Светло-серый	7	Белый	15

**Объект Line** устанавливается на форму во время разработки программы, как и другие объекты управления. Положение объекта Line на форме можно изменить программным путем.

**Элемент управления Shape** служит для изображения геометрических фигур: квадратов, прямоугольников, эллипсов, окружностей.

Элемент Shape обладает практически теми же свойствами, что и элемент Line, но имеет и ряд специфических свойств. Основные свойства *Top*, *Left*, *Height*, *Width*, *Shape*, *BorderStyle*, *BorderWidth*, *FillStyle*, *FillColor*, *BorderColor*.

**Top**, **Left**, **Height**, **Width**, **BorderColor** – эти свойства аналогичны свойствам других элементов управления. Они определяют положение объекта на форме, его размеры и цвет обводки.

**Shape** – определяет форму объекта.

**FillStyle** - обеспечивает автоматическое заполнение фигур, построенных с помощью графических методов.

**BorderStyle, BorderWidth** – определяют стиль контура и толщину линии соответственно. Эти свойства аналогичны соответствующим свойствам объекта Line.

**FillColor** – определяет цвет заполнения объекта, аналогичен свойству **BorderColor** объекта Line.

### 3.17.2. Задание

1. Исследуйте свойства стиль, цвет и толщина объекта Line
2. Исследуйте свойства форма, стиль линии, стиль заполнения, толщина линии, цвет линии объекта Shape.

#### Указания к выполнению задания

К пункту 1:

– поместите на форму три массива элементов объекта по семь элементов в каждом. На одном массиве элементов управления продемонстрируйте свойство стиль линии, на втором – толщину, на третьем - цвет линии.

К пункту 2:

– поместите на форму пять массивов элементов объекта Shape. Число элементов в массиве зависит от числа значений свойства. На первом массиве элементов продемонстрируйте форму объекта, на втором - стиль обводки (бордюра), на третьем – толщину линии, на четвертом – стиль заполнения и на пятом – цвет заполнения.

#### Контрольные вопросы

1. Какие графические объекты и графические элементы управления имеются в VB6?
2. Что такое графический объект, графический элемент управления?
3. Перечислите основные свойства элемента управления Line.
4. Перечислите основные свойства элемента управления Shape.
5. Какие способы управления цветом Вам известны? Приведите примеры.

## 3.18. Исследование методов Line и Circle

**Цель занятия:** Приобрести практические навыки в использовании методов Line и Circle

**Время:** 2 часа.

**Литература:** Л1 с.324 - 331, Л3 с. 100 – 108.

### 3.18.1. Общие сведения

**Графический метод** – это метод, который позволяет изображать на объекте данного класса какой-нибудь геометрический элемент, например точку, линию, окружность и т.д. Графический метод ориентирован на абсолютную или относительную систему координат графического объекта.

К графическим методам относятся методы Line, Circle, Pset.

Метод Line предназначен для изображения линии или прямоугольника.

Синтаксис метода: Line (x1,y1)-(x2,y2),C,BF

где x1, y1, x2, y2 – координаты концов отрезка прямой, C – цвет, B – признак прямоугольника и F – признак закрашенного прямоугольника.

Метод Circle предназначен для изображения овалов, дуг и секторов.

Синтаксис метода: Circle (x,y),R,C,[-]угол\_нач, [-] угол\_кон, Kc

где x,y – координаты центра окружности, R – радиус, C – цвет, угол\_нач – начальный угол сектора или дуги, угол\_кон – конечный угол сектора или дуги. Kc - коэффициент сжатия эллипса. Если перед значением угла стоит знак "-", то конец дуги соединяется с центром окружности.

**Масштаб изображения.** В Visual Basic используется абсолютная система координат ориентированная на верхний левый угол экрана со значениями  $x=0$ ;  $y=0$ , то есть представляет собой IV квадрант прямоугольной декартовой системы координат.

Основной единицей измерения в VB является твип. Твип = 1/1440 логического дюйма. Логический дюйм – это расстояние на форме, которое при печати на принтере будет равно 1 дюйму (1 дюйм = 2,5 см). Используя свойство ScaleMode, можно перейти к другим единицам измерения, например:

Form1.ScaleMode = 3 – установлена единица измерения *пиксель*.

Пиксель – одна точка на экране монитора, число пикселей определяется установленным разрешением экрана Windows.

По умолчанию установлена единица измерения твип.

Для установки другого масштаба, пользовательского, используется метод Scale. Синтаксис метода:

[имяОбъекта]. Scale (x1,y1) – (x2,y2)

где x1,y1 – координаты верхнего левого угла экрана; x2,y2- координаты правого нижнего угла формы (графического объекта).

**Управление пикселем.** Чтобы использовать режим управления пикселем необходимо установить свойство ScaleMode =3

Число твипов, приходящихся на один пиксель, возвращают функции TwipsPerPixelX и TwipsPerPixelY.

Для управления цветом точки используется метод **Pset**. Синтаксис метода: Pset(x,y) [ C].

Метод Pset можно использовать для изображения графиков функций, а также для закрашки фигур произвольной формы.

Для определения цвета точки используется метод **Point**.

Синтаксис метода: Object. Point(x,y)

Свойства объектов, нарисованных с помощью графических методов, зависят также от некоторых свойств графического объекта:

DrawMode – определяет, как цвет рисованного объекта взаимодействует с цветом графического объекта.

DrawWidth – определяет толщину линии рисованного объекта.

FillColor – определяет цвет заполнения рисованного объекта, если контур замкнут.

ForeColor - определяет цвет линии рисованного объекта, если он не был указан у графического метода.

FillStyle – определяет стиль линии рисованного объекта.

### 3.18.2. Задание

1. Исследуйте свойства метода Line.
2. Исследуйте свойства метода Circle.
3. Напишите и отладьте программу “Вспышка звезды”
4. Напишите и отладьте программу “Радар”.

#### Указания к выполнению задания

##### К пункту 1:

Постройте линию, прямоугольник, закрашенный прямоугольник в сочетании с различными значениями свойств графического объекта: DrawStyle, DrawWidth, FillColor. Например:

```
Private Sub Form_Click()  
    Dim i As Integer, k As Integer, Pi As Single  
    Pi = 4 * Atn(1)  
    Me.FillStyle = 0      ' стиль заливки - сплошная заливка  
    Me.DrawStyle = 2     ' стиль линии - штриховая  
    Me.DrawWidth = 1    ' толщина линии 1 твипа  
    Me.FillColor = vbYellow ' цвет заливки желтый  
    'зелёный прямоугольник, закрашен желтым цветом  
    Line (500, 100)-(1000, 3000), vbGreen, B  
End Sub
```

##### К пункту 2:

Постройте сектор, дугу, круг, эллипс в сочетании с различными значениями свойств графического объекта: DrawStyle, DrawWidth, ForeColor, FillStyle, FillColor.

##### К пункту 3:

Текст программы “Вспышка звезды”

Программа запускается при щелчке мышью по форме. Рисование “вспышки звезды” оформлено в виде процедуры StartBurst. В вызывающей процедуре дополнительно рисуется закрашенный прямоугольник, координаты одного из углов которого генерируются по случайному закону.

Option Explicit

```
-----  
Sub StartBurst()  
    ' процедура для рисования “вспышки звезды”  
    Dim i As Integer, ccode As Integer  
    Dim col As Single, row As Single  
    Randomize (1) ' преустановка базы для генерирования  
                  ' псевдослучайных чисел  
    Scale (-245, 134)-(245, -134) ' масштабирование формы  
    For i = 1 To 300  
        col = 245 * Rnd  
        If Rnd < 0.5 Then col = -col  
        row = 134 * Rnd  
        If Rnd < 0.5 Then row = -row  
        ccode = 15 * Rnd
```

```

        Line (0, 0)-(col, row), QBColor(ccode) ' рисование луча
    Next i
End Sub
-----
Private Sub Form_Click()
    Cls
    Static col, row As Single
    Static ccode As Integer
    Scale (-245, 134)-(245, -134)
    DrawMode = 7 ' установка режима взаимодействия цвета
                ' рисуемого объекта с цветом фона
    StartBurst ' вызов процедуры для рисования "вспышки звезды"
    Line (0, 0)-(col, row), QBColor(ccode), BF ' повторное воспроизведение
                ' прямоугольника в старых координатах для стирания

    col = 100 * Rnd
    If Rnd < 0.5 Then col = -col
    row = 50 * Rnd
    If Rnd < 0.5 Then row = -row
    ccode = 15 * Rnd
    Line (0, 0)-(col, row), QBColor(ccode), BF ' воспроизведение прямоугольника
                ' в новых координатах

End Sub

```

#### К пункту 4:

Текст программы "Радар"

```

Private Sub Form_Click()
    Cls
    Const Pi = 3.14159
    Const Alfa = 50
    Dim x As Integer, y As Integer, r As Integer
    Dim xNach As Single, xKon As Single, nCount As Integer
    Dim i As Integer, j As Integer
    'Рисуем окружность в центре формы
    x = Me.ScaleWidth / 2
    y = Me.ScaleHeight / 2
    r = x / 2
    Me.FillStyle = vbSolid ' контур сплошной
    Me.DrawMode = vbCopyPen ' взаимодействие с цветом
    Me.FillStyle = vbFSTransparent ' контур прозрачный
    Me.Circle (x, y), r
    'Рисуем сектор радара
    Me.DrawMode = vbXorPen ' исключающее ИЛИ
    Me.FillColor = vbGreen
    Me.FillStyle = vbSolid ' контур сплошной

```

nach:

```

For i = 0 To 360
  xNach = i
  xKon = i + Alfa
  If xKon > 360 Then xKon = xKon - 360
  Me.Caption = "Дуга" & Alfa & "градусов с началом в" & i
  Me.Circle (x, y), r, , -xNach * Pi / 180, -xKon * Pi / 180
  For j = 1 To 15000: A = A: Next j
  DoEvents
  Me.Circle (x, y), r, , -xNach * Pi / 180, -xKon * Pi / 180
Next i
GoTo nach
Me.Caption = "Конец программы"
End Sub

```

### Контрольные вопросы

1. Для чего предназначен метод Line? Приведите синтаксис метода.
2. Для чего предназначен метод Circle? Приведите синтаксис метода.
3. Какие свойства графического объекта влияют на графические методы?
4. Поясните работу программы "Вспышка звезды".
5. Поясните работу программы "Радар".

## 3.19. Построение графиков и диаграмм

**Цель занятия:** приобрести практические навыки в построении графиков функции одной переменной, столбиковой диаграммы, круговой диаграммы.

**Время:** 4 часа.

**Литература:** Л1 с. 324 - 331, Л3 с. 100 – 116.

### 3.19.1. Общие сведения

Для построения графика функции можно использовать метод Pset или Line. В первом случае график строится точками, поэтому качество графика зависит от шага табулирования функции. При использовании метода Line шаг табуляции может быть достаточно большим, что повышает скорость рисования графика. При построении графика с помощью метода Line рекомендуется использовать сокращенный синтаксис метода: Line – (x2,y2),c. В этом случае в качестве начальной точки берется текущее положение точки вставки на графическом объекте. Поэтому перед построением графика необходимо определить начальную точку. Для этого можно использовать метод Pset или выполнить позиционирование точки ввода с помощью операторов CurrentX, CurrentY.

Для построения диаграмм изучите материал в Л1 раздел 4.9.3.

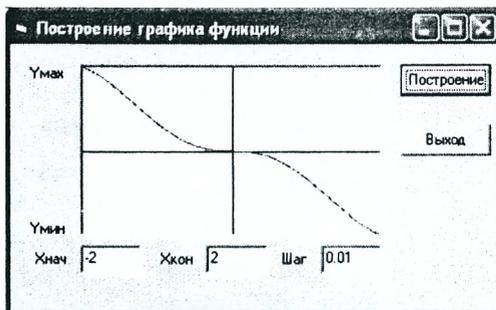


Рис. 3.19.1. Построение графика функций

### 3.19.2. Задание

1. Постройте график функции согласно варианту задания на произвольном отрезке определения функции.
2. Постройте столбиковую диаграмму.
3. Постройте секторную диаграмму.

#### Указания к выполнению задания

Для решения данной задачи необходимо протабулировать функцию на заданном отрезке с некоторым шагом и найти максимальное и минимальное значение функции на отрезке табулирования. Затем выполнить масштабирование формы в соответствии с вычисленными параметрами. Построить график функции, используя метод Pset. Для построения графика функции необходимо повторить операцию табулирования функции, и на каждом шаге строить точку. Чтобы график функции был плотным, шаг табулирования следует выбирать достаточно малым, например, 0.01.

Разработайте форму согласно рис. 3.19.1.

– установите на форму элемент управления PictureBox. Присвойте ему имя picGraph1;

- установите кнопки cmdPlot (Построение) и cmdExit (Выход);
- установите метки для обозначения Xнач, Xкон, Шаг, Ymax, Yмин;
- установите три окна ввода txtXnach, txtXkon, txtDx для ввода начальных значений;

Запишите в обработчик события Click кнопки Построение текст программы:

```
Option Explicit
```

```
Dim x As Single, y As Single, m xn As Single, ymin As Single
```

```
Dim xk As Single, ymax As Single, dx As Single
```

```
-----  
Function FNy(x As Single) ' функция пользователя
```

```
    y = 2 * Sqr(Abs(x ^ 3)) * Sin(x)
```

```
    FNy = y
```

```
End Function
```

```
-----  
Private Sub cmdgraf_Click()
```

```
    Picture1.Cls
```

```
    Picture1.ScaleMode = 3
```

```
    xn = Val(txtXnach.Text): xk = Val(txtXkon.Text): dx = Val(txtDx.Text)
```

```
    ymax = FNy(xn): ymin = ymax
```

```
    For x = xn To xk + dx / 2 Step dx
```

```
        y = FNy(x)
```

```
        If y > ymax Then ymax = y
```

```
        If y < ymin Then ymin = y
```

```
    Next x
```

```
    Picture1.Scale (xn, ymax)-(xk, ymin) ' масштабирование формы
```

```
    Picture1.Line (xn, 0)-(xk, 0) ' ось X-ов
```

```
    Picture1.Line (0, ymin)-(0, ymax) ' ось Y-ов
```

```
    For x = xn To xk + dx / 2 Step dx
```

```

y = FNy(x)
Picture1.PSet (x, y), vbRed
Next x
End Sub
-----
Private Sub cmdExit_Click()
End
End Sub

```

### Варианты заданий

1	2	3	4	5	6	7	8	9	10
$\sin(x)$	$e^x$	$\cos(x)$	$\sin(x+1)$	$\sin(2x)$	$e^{3x}$	$\cos(3x-2)$	$\sin(3x-1)$	$e^{2x}$	$\cos(2x)$
11	12	13	14	15	16	17	18	19	20
$\sin^3(x)$	$1/x^2$	$x/(x^2-5)$	$x^3/(x-4)$	$\text{tg}(x^2-2)$	$5/(x-1)$	$2e^x - 1/x$	$\text{Sin}(x)e^x$	$1/e^x$	$\lg(x)/x^2$

### Контрольные вопросы

1. Что такое графический объект? Назовите графические объекты.
2. Что такое графический метод? Назовите графические методы.
3. Какие единицы измерения применяются в графических объектах?
4. Расскажите назначение метода Scale. Приведите синтаксис метода.
5. Приведите основные свойства метода Line.
6. Приведите формат RGB – функции.
7. Приведите синтаксис метода PSet.
8. Для чего предназначен метод Point?

## 3.20. Анимация

**Цель занятия:** приобрести практические навыки в использовании графических окон для вывода графической информации и создания анимационных эффектов.

**Время** - 4 часа.

**Литература:** Л1 с.340 - 351, Л3 с. 116 – 125.

### 3.20.1. Общие сведения

К графическим объектам относятся элементы управления PictureBox и Image.

**Элемент управления PictureBox** является элементом-контейнером. В него можно помещать другие объекты, а также рисунки. Основными его свойствами являются: *Align*, *Autosize*, *Picture*.

**Align** - определяет положение PictureBox в форме, **Autosize** - определяет, будут ли размеры элемента управления автоматически изменяться для отображения рисунков различного размера, **Picture** - позволяет загружать графические объекты и сохранять их, содержит отображаемый графический объект.

Для загрузки графического объекта на этапе разработки проекта щелкните мышью по кнопке *троеточие* в поле свойства *Picture* – появится диалоговое окно *Load Picture*, которое позволяет осуществить поиск и загрузку нужного файла с диска. Например: C:\Windows\облака.bmp

В процессе работы программы загрузку изображений можно осуществлять с помощью функции **LoadPicture**:

```
Объект.Picture = LoadPicture("спецификация_файла")
```

Выгрузка рисунка в процессе работы осуществляется этой же функцией без указания имени файла:

```
Объект.Picture = LoadPicture()
```

Элемент управления Image также создан для отображения рисунков. Но в отличие от PictureBox, он не является элементом контейнером. Элемент управления Image не позволяет ни рисовать, ни группировать объекты.

Основными свойствами элемента Image является свойство *Stretch* и *Picture*. Свойство *Picture* аналогично соответствующему свойству элемента PictureBox. Свойство *Stretch* позволяет устанавливать соответствие между размерами элемента управления и размером рисунка.

#### **Способы создания анимационных эффектов:**

- пересчет координат объекта и использование свойств *Top* и *Left* объекта или операторов *CurrentX*, *CurrentY* для переопределения координат объекта;
- использование метода *Move*, например: `Command1.Move x, y`
- использование буфера обмена;
- прямое присвоение значений свойств одного графического объекта другому.

#### **Режим DrawMode**

Для перерисовки графических объектов используется режим *DrawMode*. Существует 15 возможных установок *DrawMode*. Во всех случаях VB сравнивает значение цвета пикселя на экране со значением цвета пикселя объекта, который рисуется на экране.

Если *DrawMode* = 7, то результатом его работы будет оператор **Xor**. *DrawMode* = 6 – соответствует оператору **Not**, а *DrawMode* = 4 определяет работу VB с оператором **Not** над значениями цветов переднего плана и использует значения этих цветов для рисования.

Если оператор **Xor** применяется дважды, то происходит восстановление первоначального цвета. Таким образом, повторное воспроизведение графического объекта при установленном режиме *DrawMode* = 7 для формы или окна позволяет стереть построенное ранее изображение без потери другой информации.

#### **Организация пауз**

Для организации пауз можно использовать различные приемы: использование пустых циклов, использование системных часов, использование таймера.

Использование пустых циклов:

```
For i=1 To 1000: DoEvents: Next i
```

Использование системных часов:

```
T=Time() или T=Timere()
```

```
While Time() – T < 2: Wend
```

#### **Использование таймера**

Visual Basic позволяет устанавливать до 36 таймеров. Основные свойства таймера: *Interval* и *Enabled*. Основное событие – *Timer*.

Рассмотрим пример. На форму установлен таймер. При щелчке мышью по форме

свойству Interval таймера присваивается значение 1000, а свойству Enabled – True и вызывается обработчик события Timer таймера. На экран выводятся значения счетчика TimerTimes. Когда значение счетчика TimerTimes станет равным 5, ему присваивается значение 0. Эта процедура повторяется несколько раз и контролируется счетчиком TimeOut. Когда значение счетчика TimeOut достигнет заданного значения (в примере оно равно 10), то таймер выключается, так как свойству Enabled присвоено значение False.

Пример использования таймера

Option Explicit

Dim Msg1 As String

---

Private Sub Form\_Click()

Msg1 = " "

Timer1.Interval=1000

Timer1.Enabled=True 'Запуск счетчика

Timer1\_Timer ' вызов обработчика события таймера

End Sub

---

Private Sub Timer1\_Timer()

' Процедура обработки события Timer

Static TimerTimes As Integer

Static TimerOut As Integer

Cls

TimerTimes = TimerTimes + 1

If TimerTimes > 5 Then

CurrentX = 200: CurrentY = 200

TimerTimes = 0

TimerOut = TimerOut + 1

If TimerOut = 10 Then

Timer1.Enabled = False

'Выключение счетчика

End If

Print Msg1

Exit Sub

Else

CurrentX = 200: CurrentY = 200

Print TimerTimes

Exit Sub

End If

End Sub

### **Анимация посредством переноса изображений через буфер обмена**

Независимо от способа создания рисунка в VB предусмотрена возможность переноса его в другой элемент управления или другие приложения Windows с помощью буфера обмена. Для этого используются методы *SetData*, *GetData()*, *GetForm*, *Clear*.

Метод **SetData** – перемещает данные в объект.

Объект. SetData [ данные ],[формат]

Здесь: *Объект* - буфер обмена, его идентификатор **Clipboard**;

*Данные* – указывают, откуда переносятся данные (ImageBox, PictureBox и др.);

*Формат* – задает формат исходных данных.

Метод **GetData()** – восстанавливает данные из объекта:  
Объект.GetData [формат]

Метод **GetFormat** – возвращает логическое значение, подтверждающее, хранятся ли в объекте данные указанного формата:

Объект.GetFormat[формат]

**Пример.** Анимационное перемещение изображения, содержащегося в элементе управления Image1.Picture в элемент управления Image2.Picture (рис.3.20.1):

```
Private Sub Form_Load()
    Image1.Left = 0 ' первоначальная установка
    Image1.Top = 0
End Sub
Private Sub Command1_Click()
    Dim i As Integer, j As Long
    Image2.Left = 1000 ' первоначальная установка объекта Image2
    Image2.Top = 1000
    Clipboard.Clear ' очистка буфера
    Clipboard.SetData Image1.Picture, vbCFBitmap ' загрузка в буфер
    ' рисунка из объекта Image1
    For j = 1 To 150000: Next j ' пауза
    Image2.Picture = Clipboard.GetData() ' загрузка в объект Image2
    ' рисунка из буфера
    For i = 1 To 300 ' движение объекта
        Image2.Left = 1000 + i * 10
        Image2.Top = 1000 + i * 10
        For j = 1 To 100000: Next j ' пауза
    Next i
    Image2.Picture = LoadPicture("W:\3dsmax3\ui\atmosApp_24a.bmp")
    ' восстановление изображения из файла
End Sub
```

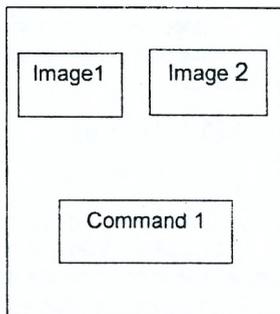


Рис. 3.20.1 Форма для анимационного перемещения объекта

### Анимация посредством присвоения значения свойства одного графического объекта другому

Другой способ анимации состоит в присвоении значения свойства Picture одного графического элемента управления другому.

**Пример.** Установите на форму четыре объекта Image. В первый, второй и третий элементы загрузите рисунки Smiley0.bmp Smiley12.bmp, Smiley3.bmp, соответственно, например:

W:\ICQ2002a\Bitmaps\Chat\Smiley0.bmp

Свойству Visible четвертого объекта Image присвойте значение False. Текст программы поместите в обработчик события Click формы:

```
Private Sub Form_Click()  
    Image4.Picture = Image1.Picture  
    Image1.Picture = Image3.Picture  
    Image3.Picture = Image2.Picture  
    Image2.Picture = Image4.Picture  
End Sub
```

### 3.20.2. Задание

1. Установите на форму элемент PictureBox и исследуйте его свойства *Align*, *Autosize*, *Picture*.

2. Установите на форму элемент Image и исследуйте его свойства *Stretch* и *Picture*.

3. Создайте объект (метку, окно ввода, командную кнопку, картинку, объект Shape), движущийся по заданной траектории.

4. Создайте анимационный эффект посредством присвоения значения свойства одного графического объекта другому объекту.

5. Создайте анимационный эффект посредством переноса изображений через буфер обмена.

#### Указания к выполнению задания

##### 1. Исследование свойств объекта PictureBox:

- установите на форму элемент управления PictureBox;
- исследуйте влияние свойства *Align* на положение и размеры элемента управления, изменяя значение данного свойства. Для возвращения размеров объекта в первоначальное состояние присвойте свойству *Align* значение *None* и установите требуемые значения свойств *Height* и *Width*. Для удаления рисунка из объекта удалите текст *Bitmap* из строки свойства *Picture*. Картинку можно взять из следующих каталогов *W:\3dsmax3\images*; *Windows\ICQ2002a\Bitmaps\Chat*; *S:\PIC*; *S:\PIC\Microsoft\Clipart*. При отсутствии на компьютере указанных каталогов, найдите файлы с расширением *BMP*, *ICO*, *JPG*;

- исследуйте влияние свойства *Autosize* на размеры рисунка, помещаемого в объект.

##### 2. Исследование свойств объекта Image:

- загрузите в форму картинку;
- изменяя значение свойства *Stretch*, установите, каким образом влияет данное свойство на размер объекта и картинки при загрузке рисунка.

##### 3. Движение объекта

- установите на форму объект, например, *Shape*. Установите его свойства: форму, цвет, размеры;

- поместите на форму таймер, установите значение его свойства *Interval* равным, например 100;

- напишите текст программы в обработчик события *Timer* таймера:

```
Option Explicit  
Dim dx As Integer, dy As Integer, X as Single, Y As Single
```

---

```
Private Sub Form_Load()  
    dx = 150: dy = 150
```

```
Shape1.Left=(Me.ScaleWidth - Shape1.Width)/2
Shape1.Top= (Me.ScaleHeight - Shape1.Height)/2
X= Shape1.Left: Y= Shape1.Top
```

```
End Sub
```

```
Private Sub Timer1_Timer()
    Shape1.Left = X + dx
    Shape1.Top = Y + dy
    If Shape1.Top <= 0 Then dy = dy * -1
    If Shape1.Top >= Me.ScaleHeight - Shape1.Height Then dy = dy * -1
    If Shape1.Left <= 0 Then dx = dx * -1
    If Shape1.Left >= Me.ScaleWidth - Shape1.Width Then dx = dx * -1
    X=X+dx: Y=Y+dy
End Sub
```

При программировании движения объекта установите на форму элемент управления PictureBox и загрузите в него картинку, используя свойство Picture, кнопку или другой элемент управления. Для управления перемещением используйте свойства Top и Left объекта или метод Move.

По пунктам 4, 5 задания реализуйте примеры, приведенные в теоретической части данной лабораторной работы. Для обмена рисунками между элементами управления используйте объекты Image.

4. Наберите и отладьте программы имеющиеся в Л1 (с. 309-313): броуновское движение, светофор и обмен рисунками с помощью буфера обмена.

5. Самостоятельно разработайте программу движения рисованного объекта по траектории сложной формы.

6. Разработайте программу обмена изображениями между тремя – четырьмя объектами с использованием буфера обмена или обмена изображениями между объектами Image или PictureBox.

### **Контрольные вопросы**

1. Поясните назначение элементов управления PictureBox и Image? В чем разница между этими элементами управления?
2. Приведите синтаксис команды обращения к свойствам элементов управления PictureBox и Image.
3. Как загрузить графический объект в элементы PictureBox или Image в процессе выполнения программы?
4. Поясните алгоритм создания анимационных эффектов.
5. Какие свойства и методы используются для организации движения объектов?
6. Как организовать паузу при воспроизведении объекта на экране?
7. Какие методы используются для обмена данными через буфер обмена?
8. Как организовать обмен данными между графическими объектами для создания анимационного эффекта?

## **3.21. Дополнительные элементы интерфейса**

**Цель:** приобрести навыки в использовании дополнительных элементов управления для разработки интерфейса пользователя.

**Время** – 4 часа.

**Литература:** Л1 с. 298 - 307, Л3 с.145-154.

### 3.21.1. Задание

1. Разработайте форму для демонстрации графиков элементарных функций. Форма должна позволять выводить на экран график трехфазного тока в каждой фазе отдельно и в любых сочетаниях, а также графики экспоненциальной, гиперболической функций и тангенса.

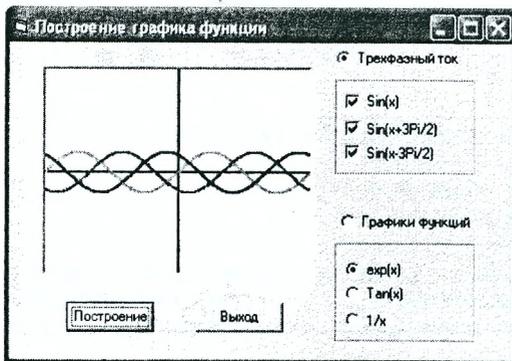


Рис. 3.21.1. Использование элементов управления

2. Разработайте форму для редактирования текста в окне TextBox: выравнивание текста по левому краю, по правому краю, по центру, а также выбора высоты шрифта и начертания: полужирный, курсив, выделенный с использованием элементов управления флажки, переключатели и списки.

3. Разработайте форму для демонстрации управления цветом с использованием линейек прокрутки и счетчиков.

#### Порядок выполнения работы

1. При выполнении задания 1 установите на форму элемент масштабирования элемента управления PictureBox симметрично относительно центра элемента управления, например следующим образом:

```
Picture1.Scale (-2 * pi, 5)-(2 * pi, -5)
```

Постройте на графике оси координат. Для увеличения толщины линии установите значение свойства DrawWidth элемента управления PictureBox равным двум. Элементы управления CheckBox и OptionButton, заключенные в рамку можно объявить как элементы массива.

В обработчик события Click кнопки Построение внесите следующий текст программы.

```
Private Sub cmdGraf_Click()  
    Picture1.Cls  
    Picture1.Scale (-2 * pi, 5)-(2 * pi, -5)  
    Picture1.Line (-2 * pi, 0)-(2 * pi, 0)  
    Picture1.Line (0, -5)-(0, 5)  
    If Option4 Then  
        For x = -2 * pi To 2 * pi Step 0.01  
            If Check1(0).Value Then y1 = Sin(x): Picture1.PSet (x, y1), vbGreen  
            If Check1(1).Value Then y2 = Sin(x + 2 * pi / 3): Picture1.PSet (x, y2), vbBlue  
            If Check1(2).Value Then y3 = Sin(x + 4 * pi / 3): Picture1.PSet (x, y3), vbRed  
        Next x  
    Else  
        For x = -2 * pi To 2 * pi Step 0.1  
            If Option1(0).Value Then y1 = Exp(x): Picture1.PSet (x, y1), vbMagenta  
            If Option1(1).Value Then y2 = Tan(x): Picture1.PSet (x, y2), vbCyan  
            If Option1(2).Value Then y3 = 1 / x: Picture1.PSet (x, y3), vbGreen  
        Next x  
    End If  
End Sub
```

2. При выполнении задания 2 установите на форму элементы управления согласно рис. 3.21.2. Заполните списки элементов ComboBox (см. текст программы). Запишите в обработчик события Click кнопки Применить следующий текст программы:

Dim Ctl As Control

```
Private Sub cmdStart_Click()
    For Each Ctl In Controls
        If TypeOf Ctl Is TextBox Then
            Ctl.Font.Name = Combo1.Text
            Ctl.Font.Bold = False
            Ctl.Font.Italic = False
            Ctl.Font.Underline = False
            Ctl.Font.Strikethrough = False
            If Check1 Then Ctl.Font.Bold = True
            If Check2 Then Ctl.Font.Italic = True
            If Check3 Then Ctl.Font.Underline = True
            If Check4 Then Ctl.Font.Strikethrough = True
            Ctl.FontSize = Combo2.Text
        End If
    Next Ctl
End Sub
```

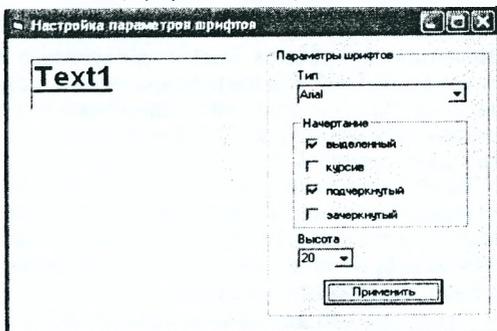


Рис. 3.21.2. Настройка параметров шрифтов

Проверьте работу программы: настройте параметры и щелкните по кнопке Применить – текст в окне Text1 изменит начертание. При вводе нового текста он будет отображаться установленным типом шрифта.

Дополните программу командами выравнивания текста в окне TextBox. Для выравнивания текста используйте свойство Align объекта TextBox.

3. Для выполнения задания по пункту 3 разработайте форму согласно рис. 3.21.3. Форма предназначена для исследования функции управления цветом RGB. Линейки



Рис. 3.21.3. Управление цветом объекта

прокрутки ScrollBar, окна ввода TextBox и счетчики UpDown, предназначены для управления цветом объекта Text1. Вертикальные линейки прокрутки VScroll1, окна ввода Text2 и счетчики UpDown1 объявите как массивы элементов управления. Для демонст-

рации изменения цвета можно использовать любой объект, имеющий свойство BackColor: TextBox, Label, Shape, текущую форму.

При изменении положения ползунка на любой линейке прокрутки синхронно должен изменяться цвет объекта Text1 и одновременно значение кода цвета должно отображаться в окне Text2. Код цвета можно вводить в окно Text2 с клавиатуры или с помощью элемента управления UpDown1. Одновременно с вводом кода цвета в окно Text2 должен меняться цвет объекта Text1 и положение ползунка. Максимальное значение свойства Value элементов управления VScroll1 и UpDown1 должно быть равно 255.

При установке элементов управления Text2 и UpDown1 соблюдайте последовательность установки: установите элемент управления Text2(0), установите элемент управления UpDown1(0) и так далее. В этом случае свойства TabIndex элементов управления Text2(i) и UpDown1(i) будут смежными, что облегчит настройку элемента управления UpDown.

Настройка элемента управления UpDown:

- вызовите контекстное меню объекта UpDown1(0) щелчком правой кнопки мыши по объекту;

- щелкните по пункту меню Properties. Открывается окно диалога Property Pages;

- выберите в окне диалога закладку Buddy;

- установите флажки AutoBuddy и SyncBuddy. При правильной установке свойств TabIndex объектов Text2(0) и UpDown1(0) в окне Buddy Control появится имя объекта Text2(0), а в окне Buddy Property – слово Default – по умолчанию. Таким образом, будут связаны свойства Text элемента управления Text2(0) и свойство Value элемента управления UpDown1(0). Настройте аналогично и другие элементы управления Text2 и UpDown1.

Текст программы запишите в обработчики событий Change элемента управления Text2 и в обработчик события Scroll элемента управления VScroll1. Текст программы:

```
Private Sub Text2_Change(Index As Integer)
    If Val(Text2(Index).Text) <= 255 Then
        VScroll1(Index).Value = Val(Text2(Index).Text)
        Text1.BackColor = RGB(VScroll1(0).Value, VScroll1(1).Value, VScroll1(2).Value)
    End If
End Sub
```

```
-----
Private Sub VScroll1_Change(Index As Integer)
    Text2(Index).Text = Str(VScroll1(Index).Value)
    Text1.BackColor = RGB(VScroll1(0).Value, _
        VScroll1(1).Value, VScroll1(2).Value)
End Sub
```

## Контрольные вопросы

1. Перечислите основные свойства элемента управления CheckBox.
2. Перечислите основные свойства элемента управления Option Button.
3. Перечислите основные свойства элемента управления ScrollBar.
4. Как установить связь между элементом управления ScrollBar и TextBox?
5. Как настроить параметры элемента управления UpDown?
6. Напишите текст программы для управления цветом элемента управления Text1.
7. Перечислите основные свойства раскрывающегося списка (ComboBox).
8. Как внести данные в список на этапе разработки программы и в процессе выполнения программы?

## 3.22. Работа с файлами

**Цель занятия:** приобрести практические навыки в работе с файлами последовательного доступа и использовании стандартных окон Windows.

**Время:** 2 часа.

**Литература:** Л1 с.377 - 390, 357 – 370, Л3 с. 140 – 154.

### 3.22.1. Общие сведения

Текстовые файлы последовательного доступа предназначены для записи и чтения неструктурированных данных. Достоинством файлов последовательного доступа является простота их создания и использования. При необходимости, файл последовательного доступа может быть создан или отредактирован любым текстовым редактором. Разделителем текста в файлах последовательного доступа является символ возврата каретки, который формируется автоматически при нажатии клавиши Enter.

Для работы с файлами данных используются команды открытия файла, закрытия файла, записи и чтения данных из файла, а также ряд функций, облегчающих работу с файлами.

Для **открытия файлов** служит команда **Open**.

Open "спецификация\_файла" For { тип файла} As [#] N [Len=длина]

В приведенном синтаксисе некоторые опции опущены.

Опция "*Спецификация\_файла*", как известно, позволяет указать диск, маршрут, имя и расширение имени файла. Например:

R:\Prognoz\Ucheb\prognoz1.dan

Тип файла указывает на его структуру и способ использования и может принимать следующие значения:

*Input* – файл последовательного доступа, открыт для чтения;

*Output* – файл последовательного доступа, открыт для записи;

*Append* – файл последовательного доступа, открыт для добавления данных;

*Binary* – двоичный файл открыт для записи и чтения данных;

*Random* – файл прямого доступа открыт для записи и чтения данных.

Так как пользователь при написании программы в принципе не может знать, сколько каналов занято и каков номер свободного канала, то для определения номера свободного канала следует использовать функцию **FreeFile**. Функция FreeFile возвращает номер свободного канала;

As – служебное слово, N – номер канала доступа к файлам. Len – длина записи для файлов прямого доступа.

Для **закрытия файлов** используется команда **Close**. Синтаксис команды:

Close [# <номер канала> ]

Команда Close с параметром номера канала закрывает указанный канал. Команда Close без параметров закрывает все открытые файлы. С целью надежного сохранения информации рекомендуется использовать вместо команды Close команду **Reset**. Эта команда, в отличие от команды Close, дает указание операционной системе сбросить содержимое буфера на диск.

Работа с файлами последовательного доступа состоит из двух самостоятельных операций: создания файла и использования файла.

**Создание файла последовательного доступа:**

Открытие файла ' (команда Open с опцией Output или Append)

Запись данных в файл (операторы Write # или Print #).

Закрытие файла ' (команда Close)

### Чтение данных из файла последовательного доступа:

Открытие файла ' (команда Open с опцией Input)

Чтение данных из файла (оператор Input # или *Line Input #*).

Закрытие файла ' (команда Close)

### Запись данных в файл последовательного доступа.

Для записи данных в файл последовательного доступа используются операторы Print # и Write #.

При использовании оператора Print числовые данные, записываемые в файл, необходимо преобразовывать в строку символов, особенно это касается вещественных чисел, так как десятичную точку программа воспринимает как разделитель данных. Поэтому при работе с числами предпочтительнее использовать оператор Write #.

Чтение данных из файла последовательного доступа осуществляется операторами *Input #*.

Оператор *Input #* имеет следующий синтаксис:

Input # <номер канала>[, "текстовое сообщение"], <список переменных>

Переменные в списке разделяются запятыми.

**Пример 3.22.1.** Создание файла последовательного доступа.

Open "R:Test.dan" For Output As #1

A\$ = "Минск – столица Республики Беларусь"

B%=13875

C!=7.58

Print#1, A\$, B%, Str\$(C!)

Close #1

**Пример 3.22.2.** Использование файла последовательного доступа

Open "R:Test.dan" For Input As #1

Input #1, A\$, B%, C\$

Print A\$, B%, Val(C\$)

Close #1

На форме будет строка следующего вида:

Минск – столица Республики Беларусь 13875 7.58

В данном примере 13875 и 7.58 – числа

Оператор Input # целесообразно использовать в сочетании с оператором Write #.

### 3.22.2. Задание

Разработайте и отладьте базу данных "Склад" с использованием файла последовательного доступа.

#### Порядок работы

1. Разработайте структуру БД согласно рис. 3.22.1

N	Наименование	Дата поступления	Номер документа	Количество	Цена	Стоимость

Рис. 3.22.1. Структура базы данных

2. Разработайте форму согласно рис.3.22.2.

Для хранения базы данных в ОЗУ используйте двумерный массив  $BD(n,5)$ . Где  $n$  – число записей в базе данных, а 5 – число полей. Номер записи нужен только на экране или на бумаге, в программе хранить его не требуется. Для отображения базы данных на экране воспользуйтесь сеткой MSFlexGrid. Для ввода данных создайте линейку из массива элементов управления (`txtText2`).

3. Опишите состав элементов управления на форме (табл. 3.22.1).

4. Опишите переменные, используемые в программе (табл.3.22.2)

5. Напишите текст программы:

' Объявление переменных уровня формы:

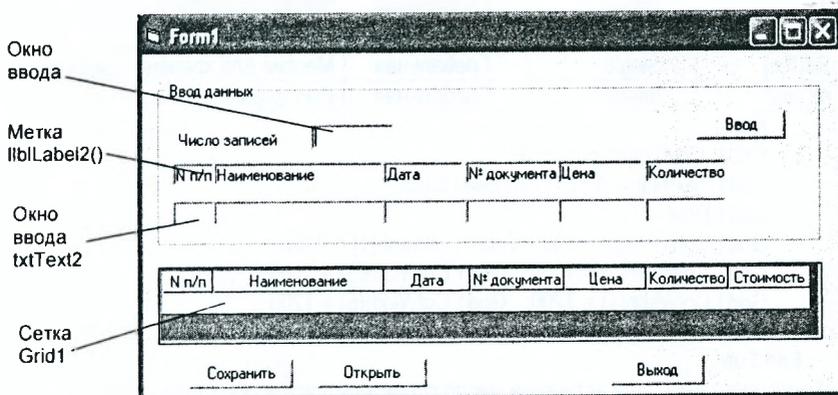


Рис. 3.22.2. Форма базы переданных “Склад”

`Dim i As Integer, j As Integer, Bd() As String`

`Dim n As Integer, Sb As Single, nKanal As Integer`

`Private Sub Form_Load()`

`' Установка начальных параметров при загрузке формы`

`Me.Height = 4275`

Таблица 3.22.1 Описание элементов управления

Тип	Имя	Назначение
Label	lblLabel1 lblLabel2()	текст “Число записей” массив элементов управления. Текст – заголовки шапки таблицы ввода данных
TextBox	txtText1 txtText2( )	ввод числа записей массив элементов управления. Поля для ввода данных
MSFlexGrid	Grid1	таблица для вывода результатов
Command	cmdVvod cmdSave cmdOpen cmdExit	ввод данных в массив сохранение данных чтение данных с диска (открытие файла) выход

```

Me.Width = 8265
For i = 0 To 5
    Grid1.ColAlignment(i)=3
    Grid1.TextMatrix(0,i) = Label2(i).Caption

```

Таблица 3.22.2 Описание переменных

Имя переменной	Тип переменной	Видимость переменной	Комментарий
i, j, k	Integer	Локальная	Используются в качестве переменных цикла
n	Integer	Глобальная	Число студентов в группе
Sb	Single	Глобальная	Средний балл
Bd(5,n)	Single	Глобальная	Массив для хранения данных
nKanal	Integer	Глобальная	Имя файла

```

Next i
Grid1.ColAlignment(6) = 3
Grid1.TextMatrix(0,5) = "Средний балл"
Grid1.Row = 0
Grid1.ColWidth(0) = 600: Grid1.ColWidth(1) = 2000
Grid1.ColWidth(2) = 1000: Grid1.ColWidth(3) = 1100
Grid1.ColWidth(4) = 1200: Grid1.ColWidth(5) = 1200
Grid1.ColWidth(6) = 1200
End Sub

```

Текст программы для установки числа строк запишите в обработчик события *Change* элемента управления *txtText1*, так как первой операцией при вводе данных необходимо указать число записей:

```

Private Sub txtText1_Change()
' Установка числа строк сетки
    n = Val(txtText1.Text)
    Grid1.Rows = n + 1
End Sub

```

Программы ввода данных, сохранения и чтения данных с диска запишите в обработчики событий соответствующих кнопок.

```

Private Sub cmdVvod_Click()
' Процедура ввода данных.
Dim i As Integer, j As Integer
' n = Val(txtText1.Text)
If n = 0 Then
    MsgBox "Укажите число записей"
Exit Sub
End If
ReDim Preserve bd(6, n) As String
If n < Val(txtText2(0).Text) Then
    n = Val(txtText2(0).Text)

```

```

        Grid1.Rows = n + 1
        ReDim Preserve bd(6, n) As String
    End If
    i = Val(txtText2(0).Text)
    Grid1.Row = i:
    For j = 0 To 5
        bd(j, i) = txtText2(j).Text
        Grid1.Col = j
        Grid1.Text = bd(j, i)
        Grid1.Visible = True
    Next j
    st = Val(txtText2(4).Text) * Val(txtText2(5).Text)
    Grid1.Col = j
    Grid1.Text = Str$(Round(st, 2))
    Grid1.Visible = True
    bd(6, i) = Str$(st)
    For j = 1 To 5
        txtText2(j).Text = ""
    Next j
    If i < n Then txtText2(0).Text = i + 1
End Sub

```

---

```

Private Sub cmdSave_Click()
'Процедура сохранения данных на диске
nKanal = FreeFile
Open "f:\Laborat\VisualBasic\file.dan" For Output As #nKanal
Write #nKanal, n
For i = 1 To n
    For j = 0 To 6
        Write #nKanal, bd(j, i)
    Next j
Next i
Close #nKanal
End Sub

```

---

```

Private Sub cmdOpen_Click()
'Процедура чтения данных (открытие файла)
nKanal = FreeFile
Open "f:\Laborat\VisualBasic\file.dan" For Input As #nKanal
Input #nKanal, n
ReDim bd(6, n)
Grid1.Rows = n + 1
For i = 1 To n
    For j = 0 To 6

```

```
        Input #nKanal, bd(j, i)
        Grid1.TextMatrix(i, j) = bd(j, i)
    Next j
Next i
Close #nKanal
End Sub
```

---

```
Private Sub cmdExit_Click()
' Процедура завершения работы с программой
    Unload Me
End Sub
```

### Контрольные вопросы

1. Какие типы файлов данных Вам известны и чем они отличаются?
2. Приведите синтаксис команды Open.
3. Приведите синтаксис команды Close.
4. Какие команды используются для записи данных в файл последовательного доступа?
5. Какие команды используются для чтения данных из файла последовательного доступа?
6. Какая последовательность команд необходима для создания файла последовательного доступа?
7. Какая последовательность команд необходима для чтения данных из файла последовательного доступа?
8. Расскажите алгоритм разработки базы данных на основе файла последовательного доступа.

## 4. РЕДАКТОР ДОКУМЕНТОВ MICROSOFT WORD

### 4.1. Ввод и редактирование текста

**Цель занятия:** ознакомиться с приемами ввода и редактирования текста, форматирования шрифтов, абзацев, вставки рисунков, формул, рисованных объектов, нумерованных списков.

**Время:** 2 часа.

**Литература:** Л1 с. 369 -387.

#### 4.1.1. Общие сведения

Каждый лист документа оформляется определенным образом. Типичными элементами документа, которые могут настраиваться пользователем, являются: размеры полей, ориентация страницы, колонтитулы, параметры абзацев, параметры шрифтов, списки, границы страниц или абзацев, таблицы, рисунки, сноски, закладки, примечания.

Параметры страницы и колонтитулов настраиваются с помощью команды **Параметры страницы** меню **Файл**.

Параметры шрифтов, абзацев, списков, границ страниц оформляются с помощью соответствующих команд меню **Формат**.

Другие элементы, не связанные с редактированием текста, вставляются в документ через меню **Вставка**.

Большинство команд форматирования текста и оформления документа будут рассмотрены в разделе порядок выполнения работы.

#### 4.1.2. Задание

1. Загрузите редактор документов Word.
2. Установите границы текста и параметры страницы.
3. Создайте титульный лист согласно образцу.
3. Наберите 2 – 3 абзаца текста по 5-6 строк, размножьте абзацы, используя операции копирования, отформатируйте абзацы.
4. Сохранить документ на рабочем диске в папке Word.

##### Порядок выполнения задания

1. Установка границ текста и полей.

Для установки границ текста введите команду **Сервис, Параметры**, выберите закладку "Вид" и установите флажок "Границы текста".

Для установки параметров полей введите команду **Файл, Параметры страницы**, и на закладке "Поля" установите их размеры: верхнее, левое и правое – 2 см, нижнее – 2,5 см.

2. Оформление титульного листа:

- Наберите текст титульного листа без учета элементов форматирования.
- Отформатируйте текст согласно образцу (см. раздел 4.1.3.) используя команду **Формат, Шрифт** или кнопки на панели инструментов форматирования. Размер шрифта, тип, начертание и цвет можно выбрать самостоятельно.

Для смещения строк 8 – 11 установите выравнивание текста "по ширине", выделите строки и переместите мышью верхний маркер горизонтальной линейки в требуемое положение. Можно воспользоваться командой **Формат, Абзац** и установить требуемое значение отступа первой строки.

- Установите масштаб "Страница целиком" выбором из списка "Масштаб" панели инструментов **Стандартная** и растяните текст на всю страницу путем вставки пустых строк. Для вставки строки установите курсор в место, где надо вставить строку, и нажми-

те клавишу Enter. Для удаления пустой строки установите курсор выше удаляемой строки и нажмите клавишу Delete.

- Установите рамку на текст:

- выделите всю страницу мышью или командой **Правка, Выделить все**;
- введите команду **Формат, Границы и заливка**. Выберите тип рамки, стиль линии, ее толщину и цвет, выберите стиль заливки используя закладку "Заливка".

### 3. Набор и форматирование шрифта:

- наберите два абзаца текста по 5 – 6 строк.
- установите для первого абзаца шрифт Times New Roman, полужирный, высотой 14 пт, а для второго абзаца - Arial высотой 12 пт, курсив.
- скопируйте первый абзац после 2 - го абзаца с использованием буфера обмена:
- а) с использованием меню;
- б) с использованием кнопок панели инструментов.
- скопируйте второй абзац после 3-го абзаца с использованием мыши.
- измените формат шрифта третьего абзаца на обычный: шрифт Times New Roman, высотой 12 пт, а для четвертого абзаца - Arial Сур , подчеркнутый, высотой 12 пт.
- наберите пятый абзац обычным шрифтом. Измените формат шрифта абзаца на шрифт второго абзаца с использованием кнопки **Формат** по образцу панели инструментов форматирования.

### 4. Форматирование абзацев.

Для настройки параметров абзацев используйте команду **Формат, Абзац** или воспользуйтесь горизонтальной линейкой.

- Установите для второго абзаца отступ слева и справа по одному см, красная строка – 1,25 см от края абзаца, выравнивание влево.
- Установите для третьего абзаца отступ слева и справа 0 см., висячая строка - 1.25 см, выравнивание по ширине.
- Установите для четвертого абзаца отступ слева и справа по одному см, красная строка - 1,25 см, межстрочный интервал - 1,5 строки, интервал между абзацами 12 пт и выравнивание вправо.
- Примените для пятого абзаца стиль второго абзаца:  
выделите второй абзац;  
щелкните мышью по кнопке **Формат по образцу** стандартной панели инструментов;  
протяните мышью по пятому абзацу.

### 5. Создание списков

Создайте в документе список в форме бюллетеня и нумерованный список, например, список студентов, перечень вопросов, список товаров и др.

Для создания списка воспользуйтесь командой **Формат, Список**. Если среди шаблонов нет подходящего списка, выберите в окне диалога один из шаблонов, введите команду **Изменить** и выполните необходимые настройки.

Для оформления списка используйте команду **Формат, Границы и заливка**.

### 6. Вставка рисунка.

Создайте рисунок с помощью графического редактора **Paintbrush Picture** и вставьте его в Ваш документ:

- введите команду **Вставка, Объект**, выберите в списке объект "Paintbrush Picture";
- нарисуйте рисунок;
- для переноса рисунка в документ щелкните мышью по редактору Word. Для редактирования рисунка щелкните дважды мышью по рисунку.

7. Создание колонок.

Преобразуйте второй абзац в две колонки с разделительной линией:

- выделите абзац;
- введите команду **Формат, Колонки**;
- выберите число колонок и установите флажок Разделитель.

8. Сохранение документа на диске.

Введите команду **Файл, Сохранить как**, выберите рабочий диск и папку *Word* (если папки нет, то создайте ее непосредственно в окне сохранения документа соответствующей кнопкой), в строке ввода "Имя файла" укажите имя файла документа, например, *Работа 1*.

9. Печать документа

Выведите документ на печать командой **Файл, Печать**.

#### 4.1.3. Пример выполнения задания

**Примечание.** Титульный лист разместите на первой странице документа и текст титульного листа растяните на всю высоту листа путем вставки пустых строк.

#### **Отчет по лабораторной работе №1**

Выполнил: \_\_\_\_\_

*Программа Far Manager предназначена для работы с файловой системой и позволяет выполнять следующие команды: просмотр каталогов, копирование, переименование и удаление файлов и каталогов методом "подведи курсор и нажми", что значительно проще, чем путем ввода командной строки, а также выполнять ряд других функций*

**Министерство образования Республики Беларусь**

**Учреждение образования**

**«Брестский государственный технический университет»**

**Кафедра информатики и прикладной математики**

**Отчеты**

по лабораторным работам по дисциплине  
*"Информатика"*

**Выполнил**

Студент(ка) группы № \_\_\_\_\_

**Проверил**

Преподаватель \_\_\_\_\_

Брест, 2005

Программа *File Manager* предназначена для работы с файловой системой и позволяет выполнять следующие команды: просмотр каталогов, копирование, переименование и удаление файлов и каталогов методом "подведи курсор и нажми", что значительно проще, чем путем ввода командной строки, а также выполнять ряд других функций.

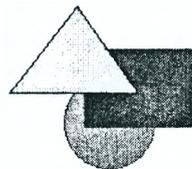
Левая и правая панели - прямоугольные окна, ограниченные двойной рамкой. В верхней рамке указывается имя каталога или подкаталога, выведенного в соответствующую панель. В панели выведено содержание каталогов в полной или краткой формах. При полной форме указывается имя файла, каталога или подкаталога, объем занимаемой памяти в байтах, дата и время создания.

Одна из панелей является текущей. Текущей называется панель, в которой находится курсор панели - прямоугольник, выделенный цветом или инверсным изображением. Смена текущей панели производится нажатием клавиши "Табуляция".

Список маркированный:

#### Ассортимент товаров лотка "ПРИМА"

Наименование	Единица учета	Цена в руб.
✓ Яблоки	кг	1300
✓ Груши	кг	1500
✓ Помидоры свежие	кг	3000
✓ Огурцы свежие	кг	2500
✓ Патисоны	кг	1000
✓ Укроп	100 гр	500



Список нумерованный

1. Петров А. А.
2. Смирнов П. И.
3. Абакумов И. К.
4. Никифоров К. П.

Контрольные вопросы

1. Как установить параметры страницы?
2. Перечислите параметры шрифтов. Как их установить?
3. Перечислите параметры абзацев. Как их установить?
4. Что такое висячая строка?
5. Как выполняется форматирование абзаца по образцу?
6. Как установить рамку на абзац (страницу)?
7. Как создать нумерованный (маркированный список)?
8. Как вставить рисунок, созданный программой Paintbrush Picture, в документ?

## 4.2. Оформление документа

Цель работы: приобрести навыки оформления документа (вставка даты, номеров страниц, символа, буквицы, ссылок, колонтитулов, оглавления и др.).

Время: 2 часа.

Литература: П1 с. 435 - 437

### 4.2.1. Задание.

1. Вставьте в документ текущую дату с возможностью обновления при очередной загрузке.
2. Вставьте в первом абзаце буквицу вместо первого символа первого предложения.
3. Вставьте в документ два рисунка.
4. Вставьте рисованный объект: куб, пирамиду и др.
5. Вставить в документ формулу.
6. Присвойте рисункам и рисованным объектам наименования с помощью соответствующей команды главного меню.
7. Вставьте для второго абзаца в документ сноску.
8. Расположите третий абзац в три колонки.
9. Вставьте колонтитулы:
  - в верхний колонтитул на нечетной странице вставьте наименование Вашего документа;
  - в верхний колонтитул на четной странице вставьте фамилию и инициалы автора документа.
  - на титульном листе колонтитул не печатать.
10. Настройте параметры страницы: установите ширину левого и правого полей, зеркальные поля.
11. Вставьте номера страниц. Номера страниц разместите внизу страницы снаружи. На первой странице номер не печатать.
12. Напишите для каждого абзаца название раздела, подраздела и выделите их с помощью стиля Заголовок1 (Заголовок2 и т. д.) текущего шаблона (шаблон Нормальный) панели инструментов.
13. Просмотрите структуру документа с помощью команды Вид главного меню или с помощью соответствующей кнопки внизу рабочего экрана.
14. Вставьте в документ оглавление.
15. Сохраните документ на диске.
16. Выведите документ на печать.

#### Порядок выполнения задания

1. Загрузка файл работы №1. Введите команду **Файл, Открыть**, выберите в списке "Папка" рабочий диск и папку Word, выберите имя файла.
2. Вставка даты. Вставьте дату 25 марта 2005 г.
  - Установите точку вставки текста в верхний левый угол второй страницы, введите команду **Вставка, Дата и время**, выберите в списке формат даты и установите флажок "Обновлять автоматически".

#### 3. Вставка буквицы.

**В**ыделите первый символ в первом предложении, введите команду **Формат, Буквица**, выберите способ размещения буквицы, тип и размер шрифта и расстояние от текста. Проверьте также положение абзацного отступа у вставленной буквицы, при необходимости уберите абзацный отступ с помощью кнопок горизонтальной линейки.

#### 3. Вставка в документ рисунков:

- введите команду **Вставка, Рисунок, Картинки**, выберите тему, выберите рисунок в теме и щелкните по кнопке **Вставить клип** на всплывающей панели инструментов.

Закройте окно диалога Вставка картинки (вставленный рисунок часто оказывается закрыт окном и не виден пользователю). Выделите вставленный рисунок, установите нужные размеры, используя маркеры, выберите в панели инструментов “Настройка изображения” режим обтекания текстом, например, “вокруг рамки” и переместите рисунок в нужное положение в документе (рис. 4.2.1.) То же можно сделать с помощью панели инструментов Рисование: команда **Действия, Обтекание текстом**;



Рис. 4.2.1  
к п. 4

– найдите на компьютере файлы с расширением **.bmp** и выпишите маршрут для доступа к этим файлам. Введите команду **Вставка, Рисунок, Из файла**, в окне диалога “Добавить рисунок” укажите маршрут для доступа к файлам типа .bmp и выберите нужный файл.

5. Вставка формулы. Напишите формулу:

$$y = \sin(x) + \frac{\sum_{i=1}^n \frac{x}{(i+1)!}}{\sqrt[3]{e^{x+8} + \operatorname{tg}(x)}} - \int_a^b \frac{dx}{1+x^2}$$

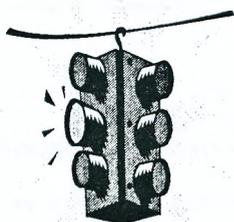


Рис.4.2.2. Светофор  
к п. 6

Введите команду **Файл, Объект**, в списке “Типы Объектов” выберите Microsoft Equation 3.0 – открывается окно для ввода формулы и панель инструментов “Формула”. При этом главное меню программы Microsoft Word заменяется меню программы Microsoft Equation 3.0. Изучите панель инструментов и команды меню. Наберите формулу, используя клавиатуру и специальные символы панели инструментов. Выберите начертание формулы и размер шрифта командами меню. Для окончания набора формулы щелкните мышью по тексту документа.

Для редактирования формулы щелкните по ней дважды мышью – вызывается программа Microsoft Equation 3.0. Для перемещения формулы или изменения размера выделите ее щелчком мыши и работайте так же, как и с другими объектами Word.

6. Присвоение названий рисункам и составление списка иллюстраций.

**Рисункам и рисованным объектам** можно присвоить наименование с помощью команд **Вставка, Название**. Наименования можно присваивать рисункам, таблицам, формулам. Формат названий Рисунок, Таблица, Формула присваивается объектам по умолчанию, нумерация осуществляется автоматически. Можно создать свой формат надписи к объектам. Созданные форматы сохраняются в списке (Рис. 4.2.2.).

Если названия рисункам, таблицам и формулам присвоены с помощью указанной команды, то можно составить список этих объектов командой **Вставка, Оглавление и указатели**, выбрать закладку **Список иллюстраций**.

7. Вставка сноски. Установите курсор после второго абзаца. Введите команду **Вставка, Сноска**, выберите режим Обычная, Нумерация автоматическая. В конце абзаца вставляется метка (номер), а текст сноски будет вставлен в конце страницы.

8. Вставка колонтитулов:

– проверьте число страниц в документе. Если число страниц меньше трех, то добавьте страницы в документ командой **Вставка, Разрыв, Начать новую страницу**;

– введите команду **Вид, Колонтитулы** - открывается слой для ввода колонтитулов и одноименная панель инструментов;

– изучите кнопки панели инструментов;

- опробуйте кнопки перехода от верхнего колонтитула к нижнему колонтитулу;
- опробуйте кнопки перехода между колонтитулами “Переход к следующему”, “Переход к предыдущему”. По умолчанию, в документе будет только один колонтитул;
- щелкните по кнопке **Параметры страниц** – откроется одноименное окно диалога. Установите флажки “Различать колонтитулы четных и нечетных страниц” и “Различать колонтитул первой страницы”. Закройте окно диалога и проверьте наличие разных колонтитулов кнопками “Переход к следующему”, “Переход к предыдущему”;
- в верхнем колонтитуле четной страницы введите свою фамилию, имя и отчество, а в верхнем колонтитуле нечетной страницы запишите тему выполняемой работы;
- закройте режим диалога Колонтитулы щелчком по кнопке **Заккрыть** панели инструментов Колонтитулы.

#### 9. Вставка номера страниц.

Введите команду **Вставка, Номера страниц**. Установите положение номеров страницы “внизу” и выравнивание “снаружи”. На первой странице номер не ставить.

#### 10. Вставка оглавления (содержания) документа.

- напишите для каждого абзаца название раздела, подраздела и выделите их с помощью стиля Заголовок1 текущего шаблона (шаблон Нормальный) панели инструментов;
- просмотрите структуру документа с помощью команды **Вид, Структура** главного меню или с помощью соответствующей кнопки внизу рабочего экрана. Измените уровень просмотра кнопками 1, 2, 3 и т. д.;
- установите курсор в место вставки оглавления (в конце документа) и введите команду **Вставка, Оглавление и указатели, Оглавление** выберите стиль оформления и число уровней.

11. Сохранение документа на диске: Введите команду **Файл, Сохранить как**, укажите маршрут и имя файла.

12. Печать документа. Введите команду **Файл, Печать**, укажите, что печатать (все, текущая страница, номера страниц или выделенный фрагмент, четные или нечетные страницы), установите число копий.

### Контрольные вопросы

1. Какие объекты можно вставлять в документы Word?
2. Расскажите порядок установки параметров страниц.
3. Как вписать математическую формулу в документ?
4. Сколько видов колонтитулов можно вставить в документ?
5. Как установить разные колонтитулы на четной и нечетной странице?
6. Можно ли устанавливать колонтитулы на первой странице?
7. Как вставить оглавление в документ средствами редактора Word?
8. Как просмотреть структуру документа?
9. Как можно составить перечень рисунков (таблиц, формул) в документе средствами редактора Word?
10. Как изменить стиль оформления заголовка (абзаца)?

### 4.3. Создание и редактирование рисунков, объектов WordArt

**Цель занятия:** приобрести навыки использования панели инструментов рисования, использования программы Word Art.

**Время:** 2 часа.

**Литература:** Л1 с. 377 – 387.

### 4.3.1. Общие сведения

Программа Microsoft Word имеет достаточно разнообразных средств для красочного оформления документа: вставка картинок из набора тематических рисунков, вставка рисунков из файлов, вставка рисунков, нарисованных средствами Word, оформление заголовков различными стилями и с помощью коллекции Word Art. Для вставки рисунков используется команды: Вставка, Рисунок; Вставка, Объект, Paintbrush Picture. Для оформления текста с помощью коллекции Word Art необходимо ввести команду Вставка, Рисунок, объект Word Art. Указанные объекты можно вставить также с помощью кнопок панели инструментов Рисование.

Панель Рисование позволяет создавать достаточно сложные рисунки с помощью таких графических примитивов, как линия, прямоугольник, овал. Используя кнопки Объем и Тень можно создавать из прямоугольников и овалов объемные фигуры. Для создания подписей к рисункам, различных заставок можно использовать кнопку Надпись. Если требуется изменить стиль, цвет линии, заливки или сделать линии невидимыми можно воспользоваться кнопками Заливка, Цвет линии, Тип линии, Тип штриха. Положение объекта можно изменить с помощью кнопки Свободное вращение.

**Напоминание:** чтобы узнать назначение кнопки на панели Рисование, установите на соответствующую кнопку указатель мыши и задержите его на кнопке до появления всплывающей подсказки.

Технология работы с графическими примитивами достаточно простая: выделите кнопку мышью, перенесите указатель мыши в точку вставки, нажмите левую клавишу мыши и протяните в нужном направлении. На выделенных рисованных объектах имеются маркеры выделения. Зацепив мышью за эти маркеры, можно изменить размеры объекта.

Для перемещения рисованного объекта установите указатель мыши на объект так, чтобы он превратился в крестик, зацепите его мышью и перенесите в нужное положение.

Для удаления объекта выделите его и нажмите клавишу Del.

Объекты можно объединять в группы. Для этого необходимо:

- активизировать кнопку Выбор объектов;
- обвести выделяемые объекты мышью, начиная с верхнего левого угла. Все выделенные объекты отмечаются маркерами выделения. Если какой-то объект оказался не выделенным, это означает, что он не попал полностью в область выделения. Отмените операцию и повторите попытку;
- выберите команду Действия, Группировать.

Дополнительные возможности по управлению объектами содержатся в меню кнопки Действия: группировать и разгруппировать объекты, порядок размещения объектов относительно друг друга, обтекание объекта текстом.

Для обеспечения возможности свободного перемещения объекта по документу введите один из режимов обтекания объекта текстом.

Другие возможности для вставки объектов предоставляет кнопка автофигуры. Можно вставлять линии, фигуры, выноски, звезды, ленты, блок-схемы и т. п.

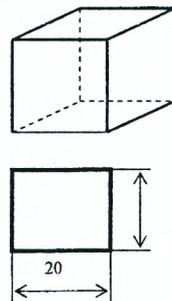


Рис. 4.3.1.

к п. 2 - 5

### 4.3.2. Задание

1. Изучите описание работы и рекомендованную литературу.
2. Выведите на экран панель инструментов Рисование.
3. Изучите назначение кнопок на панели инструментов, команды кнопки Действия, команды кнопки Автофигуры.
4. Изобразите фигуры с помощью инструментов рисования и автофигур.
5. Изобразите схему алгоритма.
6. Вставьте объект Word Art.

#### Порядок выполнения задания

1. Выведите на экран панель инструментов рисования: Введите команду **Вид, Панели инструментов**. Установите флажок напротив панели инструментов Рисование.
2. Используя инструмент рисования Линия, изобразите геометрическую фигуру в аксонометрии, например, куб. Для ускорения рисования используйте копирование линий.
3. Сгруппируйте рисунок. Активизируйте инструмент **Выбор объектов**. Установите указатель мыши выше и левее верхнего левого угла объекта, нажмите левую клавишу мыши и выделите объект протягиванием мыши так, чтобы все элементы рисунка целиком попали в область выделения. Сгруппируйте нарисованную фигуру командой **Действия, Группировать**. Выполните обтекание объекта вокруг рамки.
4. Изобразите проекции фигуры на плоскость и укажите размеры фигуры.
5. Объедините полученные фигуры в один блок, используя команду **Группировать**.

**Внимание:** после выполнения операции группирования отмените выделение инструмента **Выбор Объектов**.

6. Создайте рамку кнопкой Надпись на всю ширину страницы и поместите в нее образцы Автофигур, например рис. 4.3.2.

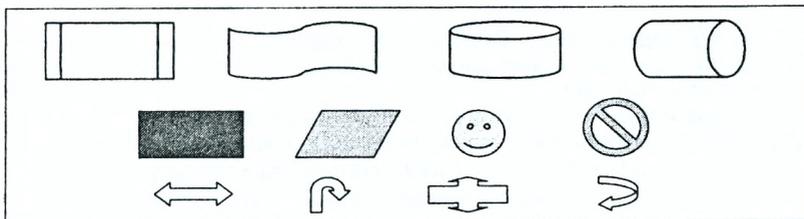


Рис. 4.3.2. Автофигуры, к п. 6 – 7

7. Выполните обтекание рамки сверху и снизу.
8. Вставьте автофигуру прямоугольник (овал) и преобразуйте его с помощью инструмента **Действия, Изменить автофигуру, Основные фигуры** в параллелограмм (ромб, цилиндр). Сделайте копию фигуры и поверните ее на произвольный угол. При выделении автофигуры кроме маркеров выделения появляется дополнительно маркер желтого цвета, с помощью которого можно редактировать форму фигуры.
9. Изобразите несколько фигур с наложением их друг на друга. Измените порядок слоев фигур, используя команду **Действия, Порядок**.

10. Изобразите произвольную фигуру. Установите толщину контура 3 пт, цвет контура черный, цвет заливки зеленый и создайте тень.

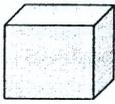
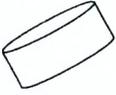
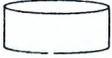


Рис. 4.3.3  
к. п. 8-12

11. Изобразите произвольную фигуру и преобразуйте ее в объемную фигуру.

12. Раскройте фигуры.

13. Изобразите схему алгоритма, используя автофигуры. Для лучшего позиционирования объектов уменьшите шаг сетки: введите команду **Действия, Сетка**, установите флажок **Привязать к сетке** и установите шаг сетки по вертикали и горизонтали 0,1 см.

Надписи выполните с помощью инструмента Надпись:

а) установите объект надпись;

б) уберите оформление рамки: выберите инструмент Цвет линии и выберите команду Нет линий;

с) запишите текст, установите абзац Без отступа, выравнивание по центру, если рамка закрывает другие части рисунка выделите рамку и введите команду **Действия, Порядок, На задний план**.

14. Напишите текст, например, "В добрый путь, студент" и оформите его с помощью коллекции Word Art:

- напишите текст;
- выделите его и выберите инструмент Добавить объект Word Art;
- выберите стиль оформления текста.

## В добрый путь, студент!

15. Создайте "водяной знак". "Водяной знак" создается в режиме колонтитулов и проявляется только в электронных документах. При выводе на печать водяные знаки выводятся как обычный текст.

"Водяной знак" можно создать путем вставки рисунка в колонтитул.

Чтобы поместить водяной знак в любом месте листа выполните следующее:

- перейдите в режим ввода колонтитулов командой Вид, Колонтитулы;
- вставьте в документ рамку с помощью команды Надпись панели инструментов рисование;
- вставьте в рамку рисунок и вернитесь в режим редактирования документа.

### Контрольные вопросы

1. Какие средства оформления документов имеются в редакторе Microsoft Word?
2. Перечислите основные инструменты панели инструментов Рисование.
3. Как сгруппировать рисованные фигуры?
4. Как вставить в документ фигурные скобки?
5. Как сделать надпись к рисунку, подписать блок в схеме алгоритма и т. п.?
6. Как изменить форму автофигуры?
7. Как изменить шаг сетки для привязки графических объектов?
8. Напишите текст "С днем победы, ветераны!" с использованием коллекции Word Art.

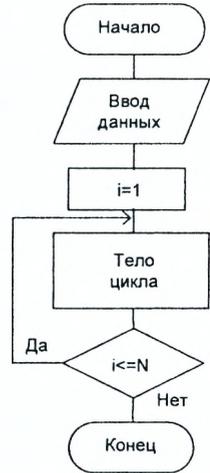


Рис. 4.3.4.  
к. п. 13

## 4.4. Таблицы и диаграммы

**Цель занятия:** приобрести навыки в создании и использовании таблиц, а также построении диаграмм средствами Word.

**Время:** 2 часа.

**Литература:** Л1 с. 387- 391.

### 4.4.1. Общие сведения

Таблицы являются достаточно удобным средством представления данных в табличной форме, создания списков, управления размещением текста, создания логотипов, рекламных объявлений и т. п.

Таблицы могут создаваться средствами редактора или вставляться из других приложений, например Excel.

Для управления таблицами используется команда главного меню Таблица. Для оформления таблиц можно использовать также панель инструментов **Таблицы и границы**.

Каждая ячейка таблицы представляет собой отдельный документ, который редактируется так же, как и любой документ Word. В таблицу можно вставлять практически любые объекты, в том числе и другие таблицы.

Строки и столбцы в таблице по умолчанию пронумерованы. Строки нумеруются цифрами натурального ряда чисел, начиная с единицы, а столбцы нумеруются буквами латинского алфавита.

Ячейки таблицы образуются на пересечении столбцов и строк. Адрес ячейки образуется из номера столбца и строки, на пересечении которых она находится, например, A1, C10, D11.

#### Вставка диаграмм

Редактор Word позволяет создавать простенькие диаграммы собственными средствами (рис. 4.4.1). Алгоритм создания диаграммы:

– создайте таблицу и внесите в нее данные (табл. 4.4.1);

Таблица 4.4.1

Выпуск автомобилей <sup>5</sup>	"1990"	"2000"	"2004"
МАЗ	1200	1500	1600
КАМАЗ	2300	2400	2350
УРАЛ	1000	1500	1700

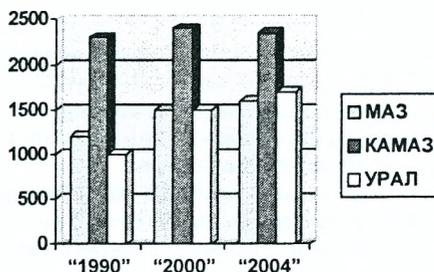


Рис. 4.4.1. Столбчатая диаграмма

– выделите таблицу вместе с верхней и левой шапкой и введите команду **Вставка, Рисунок, Диаграмма**;

– выделите диаграмму и введите команду Действия, Обтекание текстом, Вокруг рамки.

<sup>5</sup> Цифры в таблице условные

В таблице и на диаграмме приведен пример построения диаграммы. Марки автомобилей (левая шапка таблицы) используются программой для создания "легенды", то есть обозначения соответствующих столбцов. Верхняя шапка таблицы используется для создания подписей по оси x-ов. Если в верхней шапке указываются цифры, как в примере, то они должны быть заключены в кавычки, иначе программа воспринимает их как данные.

#### 4.4.2. Задание

1. Создайте таблицу по приведенной форме (табл.4.4.2), заполните ее и подсчитайте выручку.

2. Создайте таблицу произвольной структуры.

3. Постройте столбиковую диаграмму по данным таблицы по п.1.

##### Порядок выполнения задания

1. Создание таблицы.

Введите команду **Таблица, Добавить, Таблицу**. Установите требуемое число строк и столбцов.

Таблица 4.4.2

Наименование деталей	Единица измерения	Цена за единицу, руб.	Количество	Продано	Выручка, руб.
1	2	3	4	5	6
Поршень в сборе	шт.				
Шатун	шт.				
...					
Итого:	XXXXXX	XXXXXX	XXXXX	XXXXXX	

Другой способ создания таблицы: выделите на панели инструментов Стандартная кнопку **Добавить таблицу** и выделите мышью требуемое число строк и столбцов.

2. Вычисления в таблице:

– установите курсор в ячейку F3 и введите команду **Таблица, Формула**. В окне диалога сотрите формулу, предлагаемую по умолчанию и введите формулу " $=C3*E3$ ". Введите аналогичные формулы в другие строки таблицы;

– для вычисления суммы выручки установите курсор в соответствующую ячейку строки "Итого" и введите команду  $=SUM(F3:Fn)$ , где вместо Fn следует указать адрес последней ячейки в суммируемом диапазоне.

3. Редактирование таблицы:

- вставьте строку в середине таблицы;
- вставьте строку в конце таблицы;
- вставьте колонку в середине таблицы;
- вставьте колонку в конце таблицы;
- удалите лишние строки и столбцы.
- отформатируйте таблицу с помощью команды **Автоформат**.

4. Установка режима переноса таблицы на новую страницу и печати шапки таблицы на каждой странице:

– выделите строки в заголовке таблицы, которые надо повторять при переходе больших таблиц на новую страницу и введите команду **Таблица, Заголовки**. Команда действует как переключатель: повторный щелчок мышью по команде Заголовок отменяет команду;

– другой способ: введите команду **Таблица, Свойства таблицы, Строка**. Установите флажки “Разрешить перенос строк на следующую страницу” и “Повторять как заголовков на каждой странице”.

5. Присвоение названий таблицам. Выделите таблицу и введите название таблицы, используя команду **Вставка, Название**. Создайте несколько копий таблицы и присвойте им название. Сформируйте список таблиц: **Вставка, Оглавление и указатели, Список иллюстраций**.

6. Создание таблицы произвольной структуры. Создайте таблицу из требуемого числа строк и столбцов. Для объединения строк, столбцов или ячеек выделите их мышью и введите команду **Объединить ячейки**. Для разбиения ячеек на строки или столбцы выделите их мышью и введите команду **Разбить ячейки**.


7. Преобразование таблиц в текст и текста в таблицу. Преобразуйте одну из таблиц в текст. Выделите таблицу, введите команду **Таблица, Преобразовать, Преобразовать в текст**, укажите символ, который будет использоваться для разделения данных в ячейках.

Сделайте копию полученного текста и преобразуйте его в таблицу командой **Таблица, Преобразовать, Преобразовать в таблицу**.

8. Постройте по таблице диаграмму продажи запасных частей.

9. Выведите документ на печать.

### Контрольные вопросы

1. Как вставить таблицу в документ?
2. Какие средства Word используются для управления таблицами?
3. Как отредактировать текст в таблице?
4. Каким образом осуществляется указание адреса ячейки?
5. Как настроить свойства таблицы?
6. Как создать таблицу сложной формы?
7. Как вставить в документ диаграмму средствами Word?

## 4.5. Работа с формами и макросами

**Цель занятия:** закрепить навыки работы с таблицами, приобрести начальные навыки в создании форм документов и простейших макросов.

**Время:** 2 часа.

**Литература:** Л1 с. 391 - 395, 399 – 403.

### 4.5.1. Общие сведения

На практике часто приходится печатать различные отчеты, справки, списки и т. п. На разработку этих форм уходит достаточно много времени. Гораздо проще создать шаблон и хранить его на диске, вызывая для заполнения по мере необходимости.

Редактор Word содержит значительное число шаблонов и мастеров для создания различных документов от календарей до дипломных проектов. Вызвать их можно командой **Файл, Создать**, выбрать необходимую закладку, документ, установить переключатель **Документ** и щелкнуть по кнопке ОК. Однако эти шаблоны не всегда соответствуют принятым стандартам. Поэтому проще создать собственный шаблон.

Для создания шаблона необходимо ввести команду **Файл, Создать**, выбрать закладку **Общие**, установить переключатель **Шаблон** и щелкнуть по кнопке ОК. Ввести необходимую информацию и вставить поля форм. Защитить форму и сохранить шаблон на диске.

Шаблон содержит постоянный текст и переменный. Постоянный текст вводится при разработке шаблона и не изменяется при оформлении документа, переменный заполняется при оформлении документа. Переменная информация вводится в специальные поля. Эти поля могут быть заполнены текстом - примером, который может заменяться пользователем. В качестве базовых полей используются текстовое поле, раскрывающийся список и флажок.

Вставка полей производится с помощью кнопок панели инструментов **Формы**. Текстовое поле позволяет вводить текст, числа, даты. Флажок позволяет вводить ответы типа "Да" – флажок установлен, и "Нет" – флажок снят. Список создается при разработке формы и заполняется при оформлении документа выбором нужного значения из списка. Настройка свойств полей осуществляется с помощью кнопки Параметры поля формы или командой Свойства контекстного меню поля формы.

#### **Алгоритмы заполнения полей:**

- а) вставка текстового поля:
  - установите курсор в место вставки;
  - щелкните по кнопке *Текстовое поле* панели инструментов Формы – вставляется текстовое поле;
  - щелкните по кнопке *Параметры поля формы* – открывается окно диалога Параметры текстового поля;
  - установите в строке Тип - Обычный текст, в строке Максимальная длина – 20, при необходимости заполните строки Текст по умолчанию и Формат текста;
  - щелкните по кнопке ОК.
- б) вставка числового поля
  - выполните пункты 1-3 п. а);
  - установите в окне Тип – Число, в окне Максимальная длина – 15; в окне Формат числа – 0,00;
  - щелкните по кнопке ОК.
- в) вставка поля даты:
  - выполните пункты 1-3 п. а);
  - установите в строке Тип – Дата.; в строке Формат даты – dd.MM.yyyy; в строке Максимальная длина – 10, в строке дата по умолчанию введите текущую дату, например, 05.01.2005
  - щелкните по кнопке ОК.
- г) вставка флажка:
  - установите курсор в место вставки;
  - щелкните по кнопке *Флажок* панели инструментов Формы – вставляется поле флажка;

- щелкните по кнопке *Параметры поля формы* – открывается окно диалога *Параметры флажка*;
- в группе состояние по умолчанию выберите нужный режим и щелкните по кнопке ОК.
- д) вставка поля со списком:
  - установите курсор в место вставки;
  - щелкните по кнопке *Поле со списком* панели инструментов *Формы* – вставляется поле со списком;
  - щелкните по кнопке *Параметры поля формы* – открывается окно диалога *Параметры поля со списком*;
  - введите в строку *Элемент списка* первый элемент и щелкните по кнопке *Добавить*. Введите второй элемент списка и т. д.;
  - щелкните мышью по кнопке ОК.

Кнопка *Затенение полей* формы должна быть всегда активна.

По окончании ввода данных установите защиту постоянной информации кнопкой *Защита формы*. После этого курсор будет перемещаться только по незащищенным полям, станут активны флажок и список.

Сохраните шаблон на диске. Шаблоны документов имеют расширение .dot и хранятся в папке *Шаблоны программы Windows*.

Значки сохраненных пользовательских шаблонов появляются в окне диалога *Создание документа* на закладке *Общие*.

### **Макросы**

Макросы служат для автоматизации выполнения наиболее часто встречающихся однотипных операций.

**Пример.** Создать макрос для выделения строки и изменения текущего шрифта на курсив с подчеркиванием.

#### **Решение.**

- установите курсор в начало строки;
- введите команду *Начать запись (Сервис, Макрос, Начать запись)* или щелкните дважды мышью по кнопке *ЗАП* в строке состояния);
- введите имя макроса *ТекстКурсивПодчер*, щелкните по кнопке ОК;
- нажмите клавиши *Shift + End* (выделение строки от текущего положения до конца строки);
- щелкните по кнопкам панели инструментов *Форматирование Курсив* и *Подчеркнутый*;
- остановите запись командой *Сервис, Макрос, Остановить запись* или щелкните по кнопке *Остановить Запись* на панели инструментов *Ост* (остановка записи);
- поместите на панель инструментов *Форматирование* кнопку для запуска макроса с текстом *ТКП*, без рисунка:
  - введите команду *Сервис, Настройка*, откройте закладку *Команды* и выберите в списке "Категории" команду *Макросы*;
  - выделите в списке *Команды* макрос *ТекстКурсивПодчерк* и перетащите его мышью на панель инструментов *Форматирование*;
  - щелкните по кнопке *Изменить выделенный объект* в окне диалога *Настройка*;
  - выделите в списке команд команду *Только текст (всегда)* и в строке *Имя* напишите "ТКП";
  - закройте окно диалога.

Проверьте работу макроса: установите курсор в начало форматируемого текста и щелкните по кнопке *ТКП* панели инструментов.

## 4.5.2. Задание

1. Изучите описание работы и рекомендованную литературу.
2. Разработайте Анкету студента по форме рис. 4.5.1.

### Анкета

Фамилия	(текстовое поле)
Имя	(текстовое поле)
Отчество	(текстовое поле)
Дата рождения	23/01/1980 (текстовое поле типа дата с шаблоном)
Социальное положение	Учащийся (поле со списком)
Образование	Среднее техническое (поле со списком)
Членство в профсоюзе	<input checked="" type="checkbox"/> (флажок)
Партийность	(поле со списком)
Домашний адрес	(текстовое поле)
Телефон	56-17-34 (тестовое поле с примером заполнения)

Рис. 4.5.1. Форма шаблона анкеты

4. Разработайте учетную карточку спортсменов по форме рис.4.5.2.
4. Заполните Анкету и Учетную карточку спортсмена в режиме документа и выведите на печать.
5. Создайте простейшие макросы для:
  - удаления текста от положения курсора до конца строки;
  - удаления текста от положения курсора до начала строки;
  - выделения текста красным цветом.

### Контрольные вопросы

1. Для чего предназначены шаблоны? Как они создаются?

### Учетная карточка

Место проведения соревнований:

Дата:

Дистанция 10 км (поле со списком)	Возрастная группа → общая (до 21) (поле со списком)	Лучшее время (текстовое поле)
Вид соревнования	Бег → <input type="checkbox"/>	Спортивная ходьба → <input type="checkbox"/>
ФИО	(текстовое поле)	
Адрес	(текстовое поле)	
Город	(текстовое поле)	Индекс →
Страна	Белоруссия (текстовое поле)	Телефон →

Рис. 4.5.2. Форма шаблона учетной карточки спортсмена

2. Как используются шаблоны?
3. Как открыть панель инструментов Формы?
4. Как вставить в документ поле для ввода даты?
5. Как ввести данные в список?
6. Для чего необходимо защищать форму?
7. Как создать макрос?
8. Как назначить макросу комбинацию клавиш?
9. Как назначить макросу кнопку на панели инструментов?

## 4.6. Слияние документов

**Цель занятия:** изучить порядок слияния документов для рассылки, изготовление марок и почтовых наклеек.

**Время:** 2 часа.

**Литература:** Л1 с. 395 – 399.

### 4.6.1. Общие сведения

Механизм слияния документов является эффективным средством автоматизации ряда рутинных операций: отправка, поиск информации, написание писем и подготовка их к отправке, создание почтовых наклеек, почтовых марок и т. д. Механизм слияния используется не только для подготовки писем, но и других документов, использующих единый шаблон, например, расчетные листки при начислении заработной платы.

При слиянии присутствуют два объекта: шаблон письма или документа и источник данных. Источник данных может быть подготовлен средствами редактора Word или другими приложениями Windows: электронными таблицами, системами управления базами данных.

### 5.6.2. Задание

1. Подготовьте письмо для рассылки (поздравительную открытку, приглашение на конференцию, семинар и т. д.).
2. Оформите конверты для рассылки писем
3. Создайте почтовые наклейки.

#### Порядок выполнения задания

1. Подготовка письма.

- Откройте новый документ и напишите текст письма, оставляя пустые места для таких сведений как обращение (дорогой друг, уважаемый(ая) ..., товарищи, господин, госпожа) фамилия, имя, отчество, адрес и т. д.

- Введите команду **Сервис, Слияние**. Открывается окно диалога Слияние.

- Выберите команду **Основной документ, Создать, Документ на бланке, Активное окно** (т. е. в качестве документа будет использоваться текущее окно редактора Word).

- Введите команду **Источник данных, Получить данные, Новый источник данных** – открывается окно диалога Создание источника данных. С помощью строки ввода "Поле" и кнопки **Добавить поле** внесите в список "Поля в строке заголовка" новые поля, если существующих полей недостаточно, а также удалите ненужные поля с помощью кнопки **Удалить поле**. По окончании редактирования структуры базы данных щелкните по кнопке ОК. Сохраните базу данных.

- Введите команду **Правка источника данных**. Открывается окно диалога для ввода данных. Введите в список сведения о Ваших друзьях, знакомых, родственниках, деловых партнерах и т. д.

Пример базы данных:

Обращение	Фамилия	Имя	Отчество	Адрес1	Город
Уважаемый	Петров	Александр	Николаевич	Улица Минская, 10	Барановичи

- Введите команду **Правка Основного документа**.

- Вставьте поля слияния в нужных местах документа, выбирая их из списка "Добавить поле слияния".

Пример основного документа с полями слияния приведен на рис. 4.6.1.

Введите команду **Объединить**. При необходимости, установите фильтр для отбора нужных записей, например, по обращению, по адресу, по другому ключевому полю.

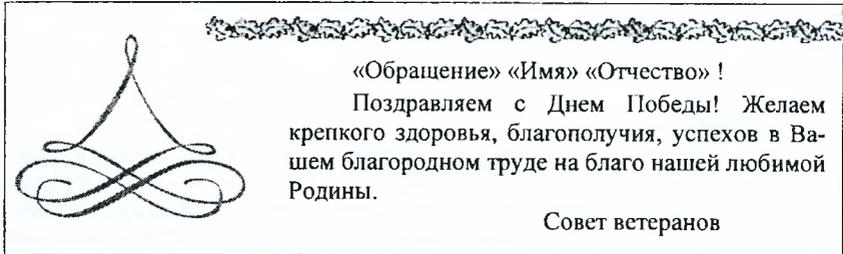


Рис. 4.6.1. Основной документ с полями слияния

- Введите режим просмотра документа Обычный и просмотрите результаты выполнения команды объединить. В поля ввода будут вставлены значения из базы данных.
- 2. Оформление конверта.
  - Введите команду **Сервис, Параметры**, выберите закладку **Пользователь** и в окне "Почтовый адрес" напишите свой домашний адрес по правилам почтовых отправлений. Этот адрес будет впоследствии переноситься на конверт.
    - Откройте новый документ.
    - Введите команду **Сервис, Слияние, Основной документ, Создать, Конверты, Активное окно**.
    - Введите команду **Источник данных, Получить данные, Открыть источник данных**. Выберите файл базы данных.
    - Введите команду **Настройка основного документа**.
    - В окне диалога Параметры конверта установите размеры конверта и настройте параметры печати.
    - В следующем окне диалога – "Адрес на конверте" вставьте в окне "Образец адреса" поля слияния для адреса получателя, выбирая их из списка "Вставить поле слияния".
    - Введите команду **Объединить**. Установите, при необходимости, параметры отбора в окне диалога Слияние и завершите работу щелчком по кнопке **Объединить**.
  - 3. Создание почтовых наклеек.
    - Введите команду **Сервис, Конверты и наклейки**, выберите закладку **Наклейки**.
    - Установите переключатель "Страница с одинаковыми наклейками". Введите, при необходимости, команду Параметры и установите тип наклеек, размеры, а также тип принтера. Щелкните по кнопке Создать – открывается шаблон пустых наклеек.
      - Вставьте в каждое поле картинку.
      - Сохраните документ на диске и выведите на печать.

### Контрольные вопросы

1. Поясните суть технологии слияния документов.
2. Как создать источник данных?
3. Что такое "Основной документ"?
4. Как установить отбор информации для вставки в основной документ?
5. Что надо сделать, чтобы на конвертах автоматически печатался адрес отправителя?
6. Как выполнить автоматически надписи на конвертах? Откуда программа берет данные для заполнения адреса получателя?
7. Как создать наклейки?

## 5. ЭЛЕКТРОННАЯ ТАБЛИЦА EXCEL

### 5.1. Разработка таблиц

**Цель занятия:** приобрести начальные навыки работы в электронной таблице EXCEL.

**Время:** 2 часа.

**Литература:** Л1. с. 404 – 415.

#### 5.1.1. Общие сведения

Табличный процессор Microsoft Excel - интегрированная система. Она предназначена для создания и обработки электронных таблиц, списков (баз данных), представления результатов обработки таблиц и списков в виде диаграмм и графиков функций, подготовки выходных форм документов, сохранения их на дисках и вывода на печать, моделирования процессов.

Рабочее окно Excel представлено на рис.5.1.1. В верхней части окна расположена строка заголовка (1), в которой располагаются кнопка системного меню, заголовок приложения и имя файла редактируемого документа, загруженного в окно (в начале работы по умолчанию выводится имя файла таблицы Книга 1), кнопки свертывания, разворачивания/восстановления и закрытия окна программы. Ниже расположены: строка меню (2), стандартная панель инструментов (5), панель инструментов форматирования (6), строка формул(7), рабочее окно. (8)

Документы, подготовленные в Excel, принято называть книгами. Книги сохраняются на диске в файлах с расширением .xls. Книги состоят из листов. Каждый лист представляет собой двухмерную таблицу с пронумерованными строками и столбцами. Строки нумеруются цифрами, а столбцы буквами латинского алфавита и их комбинациями. На пересечении строк и столбцов образуются ячейки. Ячейка является основным элементом электронной таблицы. Все данные вводятся в ячейки, результаты вычислений также выводятся в ячейки.

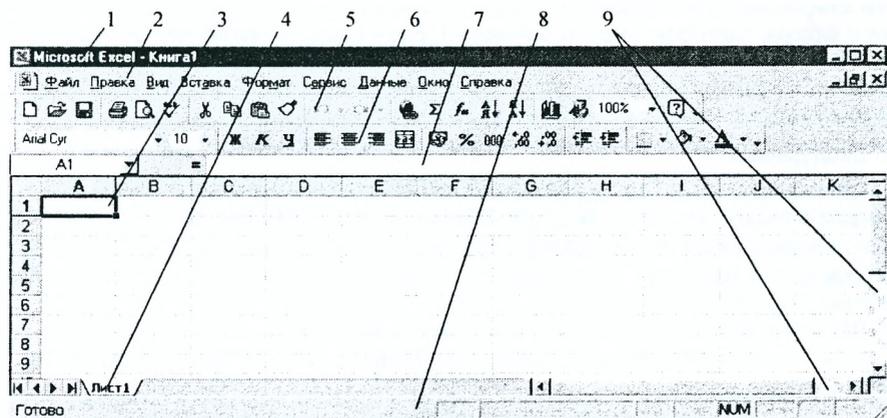


Рис. 5.1.1. Рабочее окно программы Excel: 1- строка заголовка, 2 – меню, 3 – активная ячейка, 4 – ярлычок листа, 5 - панель инструментов стандартная, 6 – панель инструментов форматирования, 7 – строка формул, 8 – строка состояния, 9 – линейки прокрутки.

## Ячейка и ее свойства

**Ячейка** - область, образованная пересечением строки и столбца. Она обозначается номером столбца и строки, на пересечении которых находится. Например, A1, IV9999.

**Диапазон** (группа, блок) - непрерывная область ячеек, обозначенная номерами начальной и конечной ячеек, разделенных двоеточием или точкой, например, A1:C10, D8:H12. Ячейке или диапазону может быть присвоено уникальное имя.

**Адрес ячейки** может быть абсолютным, относительным и смешанным: относительный - A1, E7; абсолютный - \$A\$1, \$E\$7; смешанный - \$A1, A\$1. Абсолютный адрес ячейки не меняется в операциях копирования, вставки или удаления ячеек, строк и столбцов. В формулах возможны ссылки на адрес ячейки или на ее имя. Для присвоения или изменения имени ячейке или диапазону ячеек используется команда **Вставка, Имя**. Для присвоения имени ячейке или диапазону ячеек необходимо:

- выделить ячейку (диапазон ячеек);
- ввести команду **Вставка, Имя, Присвоить**, ввести в строке ввода диалогового окна имя ячейки и щелкнуть по кнопке **OK**.

Для удаления имени ячейки введите команду **Вставка, Имя, Присвоить**, выделите в диалоговом окне удаляемое имя ячейки, щелкните команду **Удалить** и **OK**.

К ячейке, строке, столбцу или диапазону ячеек могут применяться все элементы форматирования: тип данных и способы их представления, выравнивание текста, обрамление ячеек, заливка цветом, установка параметров шрифта, защита ячейки. Форматирование ячеек осуществляется командой **Формат, ячейка** или соответствующей командой контекстного меню, а также командами **Формат, столбец** и **Формат, строка**.

## Курсор таблицы

**Курсор таблицы** (контур выделения) - двойная рамка, окаймляющая всю ячейку. В правом нижнем углу рамки на пересечении сторон располагается маленький квадрат - "маркер заполнения", используемый при заполнении ячеек рядом данных с постоянным шагом или при копировании. Для перемещения курсора используются клавиши управления перемещением курсора, **Ctrl+Home** - перейти в ячейку A1, **End + клавиши управления перемещением курсора** - последняя занятая ячейка в соответствующем направлении, используются также клавиши прокрутки и ряд других комбинаций клавиш. **Непосредственная адресация** осуществляется вводом адреса ячейки в поле Адрес ячейки Строки Формул.

## Ввод данных

Данные вводятся в Строку Формул или непосредственно в ячейку. В первом случае выделите ячейку, в которую вводятся данные, и щелкните мышью по Строке Формул. Для окончания ввода нажмите клавишу **Enter** или щелкните кнопку  $\sqrt{\quad}$  Строки Формул. Во втором случае выделите ячейку и вводите данные прямо в ячейку. По окончании ввода данных нажмите клавишу **Enter**.

При редактировании строки ввода используются клавиши:

- "**←**", "**→**" - сдвиг курсора строки ввода на один символ в соответствующем направлении;
- **Insert** - включение режима вставки символов;
- **Del** - удаление символа в позиции курсора;
- **Home, End, Tab** - переход в начало или конец текста;
- **SpaceBar** (ПРОБЕЛ) - сдвиг вправо с удалением символов или без удаления символов, в зависимости от режима Вставка/Замена;

- *BackSpace* (ВОЗВРАТ НА ШАГ) - удаление символа слева от курсора;
- *Esc* - удаление вводимого текста.

Для очистки ячейки выделите ее и нажмите клавишу *Del* или Пробел и *Enter*, или воспользуйтесь командой **Правка, Очистить**. После ввода этой команды откроется дополнительное меню с запросом, что очищать: Все, Форматы, Содержимое, Примечания.

*Признаком ввода текста* является символ апостроф ('), например: 'Сводная ведомость, 'Электронная таблица. По умолчанию вводимые данные воспринимаются как текст.

*Признаком ввода формулы* является символ "=". Символ "=" должен быть первым символом в строке ввода.

*Дата* вводится в формате ДД.ММ.ГГ или ДД.ММ.ГГГГ.

При вводе вещественных чисел дробная часть числа отделяется запятой или точкой, в зависимости от настройки *Windows* (Пуск, Настройка, Панель управления, Языки и стандарты, закладка Числа, список Разделитель целой и дробной частей числа). Для ввода формул или ознакомления с функциями *Excel* можно использовать Мастера функций. Для этого необходимо щелкнуть кнопку  $f_x$  в стандартной панели инструментов или ввести команду *Вставка, Функция*. Примеры записи формул:

=A2+2 - сложение; =24-12 - вычитание;

=F35/B7\*\$A\$2 - делит значение ячейки F35 на значение ячейки B7 и умножает на значение ячейки A2. У ячейки A2 указан абсолютный адрес;

=СТАВКА\*МЕСЯЦ - перемножаются значения, содержащиеся в ячейках с именами СТАВКА и МЕСЯЦ;

=ЕСЛИ(A2<B2;C3;D2\*E17) - условное выражение. Если значение в ячейке A2 меньше значения в ячейке B2, то результат будет равен значению ячейки C3, иначе произведению значений ячеек D2 и E17.

*При записи формул, для указания адреса ячеек, значения которых не должны изменяться при копировании формул, обязательно используйте абсолютный адрес или имена ячеек.*

### Копирование ячеек

При копировании ячеек можно использовать команды главного меню, кнопки панели инструментов или мышь.

Простейший способ копирования формул – *использование маркера автозаполнения*: выделите ячейку, зацепите мышью за маркер автозаполнения и протащите ее по ячейкам назначения.

*Для копирования данных с помощью мыши* выделите ячейку или блок ячеек, подлежащих копированию, нажмите и удерживайте клавишу *Ctrl*, перетащите выделенные ячейки на новое место, отпустите кнопку мыши, а затем отпустите кнопку *Ctrl*.

*Для копирования с помощью команд меню или кнопок панели инструментов* необходимо выполнить следующее:

- выделить ячейку или блок ячеек, подлежащих копированию;
- выбрать команду *Копировать* из меню *Правка*;
- выделить начальную ячейку или весь блок, куда будут копироваться данные;
- выбрать команду *Вставить* из меню *Правка*;
- отменить выделение блока. Чтобы удалить мерцающий контур у копируемых ячеек, щелкните по *Строка Формул* или нажмите клавишу *Esc*.

## Оформление таблицы, сохранение и вывод на печать

Для оформления таблицы выделите ее и введите команду *Автоформат* из меню *Формат*. Выберите нужный стиль оформления и щелкните ОК.

Для просмотра таблицы перед печатью выберите команду *Просмотр* из меню *Файл*. Для изменения параметров страницы выберите команду *Страница* из меню команды *Просмотр* или команду *Параметры страницы* из меню *Файл*.

Перед печатью сохраните документ на диске, для этого выполните следующее: введите команду *Сохранить как...* из меню *Файл*, укажите в строке ввода "Имя файла" имя файла (расширение имени файла .xls оставьте без изменения), выберите, при необходимости, из раскрывающегося списка "Папка" диск и рабочую папку, щелкните по кнопке ОК.

Для вывода таблицы на печать выберите команду *Печать* из меню *Файл*, укажите, что печатать: Все, Выделенный диапазон или Номера страниц, Число копий и щелкните кнопку ОК.

## Функция Просмотр

Функция Просмотр предназначена для связи двух таблиц. Она имеет два синтаксиса: вектор и массив.

Синтаксис с использованием векторов:

ПРОСМОТР(искомое значение; вектор поиска; вектор результата)

Синтаксис с использованием массива:

ПРОСМОТР(искомое значение; массив)

Здесь: *искомое значение* – значение из таблицы, содержащей данные для поиска: текст, число или логическое значение, имя или ссылка на значение, например, материал из которого изготовлена деталь, марка бетона, наименование детали и т. д.;

*вектор поиска* – диапазон (один столбец или одна строка), в которой осуществляется поиск искомого значения. Вектор может содержать текст, числа или логические значения, расположенные в порядке возрастания;

*вектор результата* – диапазон (строка или столбец), из которого выбирается результат поиска;

*массив* – диапазон, область поиска и результаты. Поиск осуществляется по первому столбцу, а результат поиска выбирается из последнего столбца указанного диапазона.

В обоих случаях программа ищет в области поиска значение, равное или ближайшее меньшее из указанного диапазона, если такое значение не найдено, то выдается сообщение об ошибке.

*Значения в области поиска должны быть отсортированы по возрастанию значения параметра поиска.*

## 5.1.2. Задание

1. Разработайте таблицу для расчета параметров статически определенной системы. Справочные данные разместите в отдельной таблице:

Поршень цилиндра паровой машины (рис. 5.1.2). имеет диаметр 40 см, а шток поршня – диаметр 5,6 см. Давление пара равно 10 ат. Найти наибольшее напряжение в штоке и соответствующее изменение его длины во время одного хода машины. Длина штока равна 75 см, материал штока – сталь.

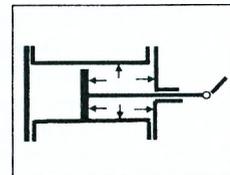


Рис. 5.1.2.  
К листингу 5.1.1

Математическая модель решения задания:

площадь поршня вычисляется по формуле  $F_1 = \pi \cdot D^2 / 4$ ;

усилия в штоке -  $P = F_1 \cdot q$ ;

площадь сечения штока -  $F_2 = \pi \cdot d^2 / 4$ ;

напряжение в штоке -  $c = P / F_2$

укорочение штока -  $dl = PL / (EF_2)$ .

В формулах обозначено:  $F_1$  – площадь поршня;  $D$  – диаметр,  $P$  – усилия в штоке;  $q$  – давление пара,  $F_2$  – площадь сечения штока,  $c$  – напряжение в штоке,  $d$  – диаметр штока, модуль упругости при растяжении и сжатии.

Пример оформления рабочего листа и выполнения задания приведен на Листинге 5.1.1

2. Оформите таблицу в соответствии с Листингом 5.1.1.

3. Просмотрите таблицу и подготовьте к печати.

4. Выведите результаты на печать

Листинг 5.1.1. Пример выполнения задания						
	A	B	C	D	E	F
1	Лабораторная работа № 1					
2	Выполнил: <Фамилия, инициалы>					
3	Тема: Разработка таблиц					
4	Пример выполнения:					
5						
6	Справочные данные:					
7	Значение модулей упругости при растяжении и сжатии	Обозначение	в системе СГС (кг/см <sup>2</sup> )	в системе СИ (н/м <sup>2</sup> )		
8	Алюминий и Дуралюмин	E	7,00E+05	7,00E+10		
9	Дерево вдоль волокон	E	1,00E+05	1,00E+10		
10	Медь	E	1,00E+06	1,00E+11		
11	Сталь	E	2,00E+06	1,96E+11		
12	Чугун	E	1,20E+06	1,20E+11		
13	Значение модулей упругости при растяжении и сжатии	G	8,00E+06	7,00E+10		
14	Объемный вес стали	$\gamma$	7,80E-03	8,80E-04		
15			кг/см <sup>3</sup>	(н/м <sup>3</sup> )		
16	Температурный коэффициент линейного расширения стали $\alpha =$					1,250E-09
17	Температурный коэффициент линейного расширения меди $\alpha =$					1,65E-09
18	Коэффициент поперечной деформации стали $\mu =$					0,3
19						
20						
21	<b>Исходные данные:</b>					
22	Параметры	Обозначение	Значение	Ед. измерения		
23	Диаметр поршня	D	40	см		
24	Диаметр штока	d	5,6	см		

25	Длина штока	L	75	см		
26	Материал штока		Сталь			
27	Давление пара	q	10	ат		
28	Справка: $q=1 \text{ ат} = 1 \text{ кг/см}^2 = 9,81 \cdot 10^4$					
29						
30	<b>Решение:</b>					
31	Наименование параметра	Формула	В системе СГС	В системе СИ		
32	Площадь поршня	$F_1 = \pi \cdot D^2 / 4$	1 256,64	0,1257		
33	Усилие в штоке	$P = F_1 \cdot q$	12 566,37	1,23E+05		
34	Площадь сечения штока	$F_2 = \pi \cdot d^2 / 4$	24,63	2,46E-03		
35	Напряжение в штоке	$c = P / F_2$	510,20	5,01E+07		
36	Укорочение штока	$dl = PL / (EF_2)$	0,0191	1,9152E-04		

#### Указания и пример выполнения задания

1. Внесите текст "Лабораторная работа № 1" в ячейку A1. Установите тип шрифта Arial, 14 пт, полужирный, выравнивание по левому краю, объедините ячейки A1:D1.

2. Внесите в ячейку C2 текст "Выполнил: фамилию и инициалы", тип шрифта Arial, 12 пт, полужирный, выравнивание по левому краю, объедините ячейки C2:F2.

3. Внесите в ячейку A1 текст "Тема: разработка таблиц". Установите тип шрифта Arial, 14 пт, полужирный, выравнивание по левому краю, объедините ячейки A3:C3.

4. Внесите в ячейку A6 текст "Справочные данные:", тип шрифта Arial, 12 пт, полужирный, выравнивание по левому краю, установите ширину столбца A такой, чтобы текст полностью умещался в ячейке.

5. Запишите в ячейки, A7:B7 шапку таблицы. Тип шрифта – Arial, 12 пт, выравнивание по горизонтали – по центру, выравнивание по вертикали – по верхнему краю, установите флажок "Переносить по словам". Для ввода символов в верхнем индексе, например, в тексте  $\text{см}^2$ , выделите цифру 2 и введите команду Формат, Ячейка, Шрифт, установите флажок верхний индекс.

6. Запишите в ячейки, A8:A14 левую шапку таблицы. Тип шрифта – Arial, 12 пт, выравнивание по горизонтали – по левому краю, выравнивание по вертикали – по верхнему краю, установите флажок "Переносить по словам".

7. Запишите в ячейки, B8:B14 обозначение параметров. Тип шрифта – Arial, 12 пт, выравнивание по горизонтали – по центру, по вертикали – по верхнему краю. Для ввода греческих символов  $\alpha$ ,  $\gamma$  установите курсор в ячейку ввода, установите тип шрифта Symbol и при английской раскладке клавиатуры введите латинский символ, соответствующий греческому символу (соответствие латинских символов греческим можно установить путем подбора, например:  $\alpha$  – a  $\beta$  – b  $\gamma$  – g  $\delta$  – m  $\epsilon$  – n  $\zeta$  – z и т. д.)

8. Запишите в ячейки, C8:D14 значения параметров. Тип шрифта – Arial, 12 пт, выравнивание по горизонтали – по правому краю, по вертикали – по верхнему краю, тип чисел – экспоненциальный.

9. Запишите в ячейки, С15:D15 размерность параметров объемного веса стали. Тип шрифта – Arial, 12 пт, выравнивание по горизонтали – по правому краю, по вертикали – по верхнему краю.

10. Аналогично внесите данные в ячейки A16:A18, F16:F18.

11. Аналогично внесите данные в таблицы “Исходные данные” и в шапки таблицы “Решение” (ячейки A31:D31; A32:A36).

12. Внесите в ячейки C32 и D36 расчетные формулы:

– в ячейку C32 -  $\text{ПИ}() \cdot C23^{2/4}$ ; в ячейку D32 -  $\text{ПИ}() \cdot (C23/100)^{2/4}$ ;

– в ячейку C33 -  $C32 \cdot C27$  ; в ячейку D33 -  $D32 \cdot C27 \cdot 9,81 \cdot 10^4$ ;

– в ячейку C34 -  $\text{ПИ}() \cdot C24^{2/4}$ ; в ячейку D34  $\text{ПИ}() \cdot (C24/100)^{2/4}$ ;

– в ячейку C35 -  $C33/C34$ ; в ячейку D35 -  $D33/D34$ ;

– в ячейку C36 -  $C33 \cdot C25 / (\text{ПРОСМОТР}(C26;A8:A12;C8:C12) \cdot C34)$ ;

– ячейку D36 -  $D33 \cdot (C25/100) / (\text{ПРОСМОТР}(C26;A8:A12;D8:D12) \cdot D34)$ .

**Внимание:** ввод адресов ячеек рекомендуется осуществлять щелчком мыши по соответствующей ячейке.

13. ПИ() – функция, возвращает значение константы – числа π.

14. Для ускорения ввода формул можно использовать копирование формул из ячеек C32:C36 в ячейки D32:D36 с последующей корректировкой.

### Контрольные вопросы

1. Назовите основные элементы рабочего окна и их назначение.
2. Что такое абсолютный, смешанный и относительный адреса ячейки, как они записываются?
3. Как вставить строку, группу строк (столбцов)?
4. Как отформатировать шапку таблицы?
5. Как осуществляется ввод данных в ячейки?
6. Как выполняется копирование ячеек?
7. Что такое Мастер функций и для чего он используется?
8. Как сохранить таблицу на диске?
9. Как загрузить таблицу с диска?
10. Как вывести таблицу на печать?

## 5.2. Табулирование функций

**Цель занятия:** ознакомиться с функциями электронной таблицы, приобрести практические навыки в табулировании функций одной и двух переменных.

**Время:** 2 часа.

**Литература:** Л1 с. 417 – 424.

### 5.2.1. Общие сведения

#### Функции электронной таблицы

Excel имеет 11 видов различных функций: математические / тригонометрические, инженерные, логические, текстовые, статистические, функции категории дата/время, функции для работы с базами данных/списками, финансовые, информационные и функ-

ции категории ссылки/массивы, проверки свойств и значений. Кроме того, Excel содержит большое число надстроечных функций (команда Сервис, Надстройки), которые используются для создания компьютерных программ в Excel, а также имеется возможность создания пользовательских функций и программ на Visual Basic for Applications.

Вызов функций осуществляется с помощью кнопки панели инструментов  $f_x$ . Эта кнопка вызывает на экран окно диалога **Мастер функций**, которое обеспечивает выбор функции из списка и пошаговый ввод сложных функций в режиме диалога. При выделении в диалоговом окне Мастера Функций требуемой функции в информационной строке появляется формат (синтаксис) выделенной функции. Можно вызвать подсказку по этой функции, выбрав команду Справка.

### Генерирование данных

Для генерирования данных (чисел, дат, текста) можно использовать механизм автозаполнения. Чтобы заполнить несколько ячеек прогрессией, необходимо записать в смежные ячейки данные, отличающиеся на величину шага, выделить эти ячейки, и перетащить маркер заполнения выделенного диапазона ячеек вниз или вправо. Можно также воспользоваться командой **Прогрессия** из меню **Правка, Заполнить** программы Excel. Для заполнения ячеек часто используемыми текстовыми записями можно создавать пользовательские прогрессии. Для этого используется закладка **Списки** команды **Параметры** в меню **Сервис**.

Ряды чисел часто применяются для табулирования функций переменных. В этих случаях целесообразнее создать собственную программу генерирования ряда чисел с настраиваемым шагом (листинги 5.2.1, 5.2.3).

### Табулирование функций

Под табулированием понимают конструирование, вычисление и составление различных математических таблиц.

Табулирование функции в Excel можно выполнить как с помощью операций копирования, так и с помощью встроенного модуля "Таблица подстановки".

Примеры табулирования функций приведены на листингах 5.2.1 – 5.2.3.

### 5.2.2. Задание

1. Сгенерируйте ряд чисел (дат), используя маркер заполнения.
2. Протабулируйте функцию одной переменной на произвольном отрезке области определения с шагом 0,2, согласно варианту задания.
3. Протабулируйте функцию двух переменных  $z=(2x+3y^2)$  при  $X_{нач}=1$ ,  $Y_{нач}=0.5$ ,  $X_{кон}=2$ ,  $Y_{кон}=2.5$ , шаг табуляции для  $X$  - 0.1, для  $Y$  - 0.2.
4. Протабулируйте функцию одной переменной, указанной в п.2, с использованием модуля Таблица подстановки.
5. Протабулируйте функцию двух переменных согласно п.3 с использованием модуля Таблица подстановки.
6. Сохраните результаты на диске.
7. Выведите результаты на печать.

## Варианты заданий

Протабулировать функции:

1. $e^x \cos^2 x$	2. $(x \ln x)^2$	3. $\arcsin(x/(1+x))$
4. $(x^2 - 1)/((x^2 + 1)(x + 1)^{1/2})$	5. $\sin x \ln(\operatorname{tg} x)$	6. $e^x(1 + \sin x)/(1 + \cos x)$
7. $1/(3 \sin x + 2 \cos x)^2$	8. $\ln x^3/x$	9. $x^3/(3+x)$
10. $x/(x^4 + 3x^2 + 2)$	11. $1/((x+1)(x^2+1)^{1/2})$	12. $x + \ln(x^2 - 4)$
13. $2(x+1)^2/(x-2)$	14. $x \ln^2 x$	15. $(4e^{2x} - 1)/e^{2x}$
16. $x^2 - e^{2x}$	17. $x e^{1/x}$	18. $\ln x/(x+1)^2$
19. $(1-x)^3/(x-2)^2$	20. $x^2 e^{1/x}$	

### 5.2.3. Указания и пример выполнения задания

#### Генерирование данных

Чтобы сгенерировать ряд чисел, например в столбец А, необходимо (Листинг 5.2.1):

- ввести в ячейку В1 начальное значение ряда;
- ввести в ячейку В2 значение шага табуляции (приращение аргумента);
- записать в ячейку А4 начальное значение ряда - ссылку на ячейку В1;
- записать в ячейку А5 формулу  $A4 + \$B\$2$ ;
- определить диапазон ячеек, куда необходимо скопировать формулу (номер начальной и конечной ячеек);
- скопировать формулу из ячейки А5 в остальные ячейки диапазона (А6:А10).

#### Табулирование функций

Порядок табулирования функции двух переменных следующий:

- сгенерировать в столбец, например А, начиная с ячейки А4, значения первого аргумента;
- сгенерировать в строку, например 3, начиная с ячейки В3, значения второго аргумента;
- записать в левую верхнюю ячейку полученной таблицы (ячейка В4) расчетную формулу с использованием смешанных адресов ячеек: у первого аргумента зафиксировать столбец, а у второго аргумента - строку;
- скопировать формулу во все ячейки блока В4:Е10.

Листинг 5.2.1. Табулирование функций двух переменных

	А	В	С	Д	Е
1	Начальное значение x	1	Начальное значение y	0,1	
2	Шаг 1-го аргумента	0,5	Шаг 2-го аргумента	0,2	
3		=D1		=C3+\$D\$2	=D3+\$D\$2
4	=B1	=2*\$A4+B\$3^2	=2*\$A4+C\$3^2	=2*\$A4+D\$3^2	=2*\$A4+E\$3^2
5	=A4+\$B\$2	=2*\$A5+B\$3^2	=2*\$A5+C\$3^2	=2*\$A5+D\$3^2	=2*\$A5+E\$3^2
6	=A5+\$B\$2	=2*\$A6+B\$3^2	=2*\$A6+C\$3^2	=2*\$A6+D\$3^2	=2*\$A6+E\$3^2
7	=A6+\$B\$2	=2*\$A7+B\$3^2	=2*\$A7+C\$3^2	=2*\$A7+D\$3^2	=2*\$A7+E\$3^2
8	=A7+\$B\$2	=2*\$A8+B\$3^2	=2*\$A8+C\$3^2	=2*\$A8+D\$3^2	=2*\$A8+E\$3^2
9	=A8+\$B\$2	=2*\$A9+B\$3^2	=2*\$A9+C\$3^2	=2*\$A9+D\$3^2	=2*\$A9+E\$3^2
10	=A9+\$B\$2	=2*\$A10+B\$3^2	=2*\$A10+C\$3^2	=2*\$A10+D\$3^2	=2*\$A10+E\$3^2

**Пример. 5.2.1.** Протабулировать функцию  $z=2*x+y^2$  при  $x$ , изменяющемся в диапазоне от 1 до 5 с шагом 0,5, и  $y$ , изменяющемся в диапазоне от 0,1 до 1 с шагом 0,2.

**Решение.**

- запишем в ячейку B1 начальное значение первого аргумента - 0,5;
- запишем в ячейку B2 шаг приращения первого аргумента - 0,5;
- запишем в ячейку D1 начальное значение второго аргумента - 0,5;
- запишем в ячейку D2 шаг приращения второго аргумента - 0,2;
- запишем в ячейку A4 начальное значение первого аргумента B1;
- запишем в ячейку A5 формулу  $A4+\$B\$2$ ;
- скопируем формулу из ячейки A5 в ячейки A6:A13;
- запишем в ячейку B3 начальное значение второго аргумента - D1;
- запишем в ячейку C3 формулу  $B3+\$D\$2$ ;
- скопируем формулу из ячейки C3 в ячейки D3:G3;
- запишем в ячейку B4 формулу  $2*\$A4+\$B\$3^2$ ;
- скопируем формулу из ячейки B4 в блок B4:G13.

**Табулирование функции с использованием модуля Таблица подстановки**

**Пример 5.2.2.** Протабулировать функцию  $\sin(x)$  на интервале от  $-\pi/2$  до  $\pi/2$  с шагом 0,5.

**Решение:**

– введите в ячейки A1, A3, A5, C1 (Листинг 5.2.2) текст Ячейка ввода, Начальное значение, Шаг табуляции, Ячейка ввода формулы, соответственно;

– введите в ячейку ввода A2 произвольное число, например 0 (это число не влияет на результат табулирования);

– введите в ячейку A4 начальное значение аргумента  $x$  -  $-\pi/2$ . Для ввода этого числа используем функцию ПИ();

– введите в ячейку A6 значение шага - 0,5;

– введите в ячейку C2 формулу  $\sin(A2)$ ;

– сгенерируйте в столбце B, начиная с ячейки B3, ряд значений аргумента;

– выделите область B2:C10 и введите команду **Данные, Таблица подстановки**;

– в диалоговом окне Таблица подстановки введите в окно ввода **Подставлять значения по строкам в...**: номер Ячейки ввода A2 и щелкните кнопку ОК. (Для ввода номера ячейки достаточно активизировать окно ввода щелчком мыши и щелкнуть по ячейке A2). Работа завершена.

**Пример 5.2.3.** Протабулировать функцию  $2x+y^2$  при  $x$ , изменяющемся от 1 до 2 с шагом 0,2, и  $y$ , изменяющемся от 0,1 до 1,6 с шагом 0,3.

**Решение:**

– обозначьте ячейку A30 как ячейку ввода по строкам, а ячейку A32 как ячейку ввода по столбцам для этого внесите соответствующие записи в ячейки A29 и A32 (см. листинг 5.2.3);

– внесите в столбец B, начиная с ячейки A30 значения аргумента  $x$ ;

– внесите в строку 29, начиная с ячейки C29 значения аргумента  $y$ ;

	A	B	C
1	Ячейка ввода		Ячейка ввода формулы
2	0		=SIN(a2)
3	Нач_Знач	-1,5708	-1
4	-1,5708	-1,0708	-0,87758
5	Шаг таб.	-0,5708	-0,5403
6	0,5	-0,0708	-0,07074
7		0,429204	0,416147
8		0,929204	0,801144
9		1,429204	0,989992
10		1,929204	0,936457

- внесите в ячейку B29 (ячейка на пересечении первого столбца и первой строки будущей таблицы) формулу  $2^*x+y^2$  или, с учетом ссылок на номера ячеек,  $2^*A30+A32^2$ ;
- выделите область B29:H35 и введите команду Данные, Таблица подстановки;
- внесите в строку ввода *Подставлять значения по столбцам в ...*: номер ячейки A32, а в строку ввода *Подставлять значения по строкам в ...*: номер ячейки A30 и щелкните кнопку ОК. Работа завершена.

Листинг 5.2.3. Пример оформления работы

	A	B	C	D	E	F	G	H	I
1			Лабораторная работа № 2						
2	Дата:	10.01.05				Выполнил:			
3									
4	Табулирование функции одной переменной		Табулирование функции двух переменных $Z = X^2 + 2\text{Sin}(y)$						
5									
6	Хнач=	-3,1416	Хнач	-1	Унач=	1			
7	Шаг табуляции:	0,5000	Шаг табуляции:	0,2	Шаг табуляции:	0,5			
8									
9	Число точек:	5	Число точек:	5	Число точек:	5			
10									
11			$\begin{matrix} y \\ x \end{matrix}$	1	1,5	2	2,5	3	
12	Аргумент	Sin(x)	-1	2,68	2,99	2,82	2,20	1,28	
13	-3,14	0,00	-0,8	2,32	2,63	2,46	1,84	0,92	
14	-2,64	-0,48	-0,6	2,04	2,35	2,18	1,56	0,64	
15	-2,14	-0,84	-0,4	1,84	2,15	1,98	1,36	0,44	
16	-1,64	-1,00	-0,2	1,72	2,03	1,86	1,24	0,32	
17	-1,14	-0,91	0	1,68	1,99	1,82	1,20	0,28	
18									
19	Табулирование функции одной переменной с использованием таблицы подстановки								
20	Ячейка ввода				Cos(X)	Sin(x)			
21	0				1	0			
22					-3,14	-1	-0,001593		
23					-2,64	-0,877	-0,480823		
24					-2,14	-0,539	-0,84233		
25					-1,64	-0,069	-0,997606		
26									

27	Табулирование функции двух переменных с использованием таблицы подстановки							
28								
29	Знач.Х	0	0,1	0,4	0,7	1	1,3	1,6
30	0	1	1,001	1,064	1,343	2	3,197	5,096
31		1,2	1,441	1,504	1,783	2,44	3,637	5,536
32	Знач.У	1,4	1,961	2,024	2,303	2,96	4,157	6,056
33	0	1,6	2,561	2,624	2,903	3,56	4,757	6,656
34		1,8	3,241	3,304	3,583	4,24	5,437	7,336
35		2	4,001	4,064	4,343	5	6,197	8,096

### Контрольные вопросы

1. Каким образом можно сгенерировать ряд чисел, используя маркер заполнения курсора таблицы?
2. Как сгенерировать ряд чисел с арифметической или геометрической прогрессией?
3. Как протабулировать функцию одной переменной?
4. Как протабулировать функцию двух переменных?
5. Опишите алгоритм табулирования функции одной переменной с использованием модуля Таблица подстановки.
6. Опишите алгоритм табулирования функции двух переменных с использованием модуля Таблица подстановки.

## 5.3. Графические возможности электронной таблицы

**Цель занятия:** познакомиться с графическими возможностями электронной таблицы, приобрести навыки использования мастера функций для построения графиков функций и диаграмм.

**Время:** 2 часа.

**Литература:** Л1 с. 424 – 428.

### 5.3.1. Общие сведения

Диаграммы позволяют представить числовые данные, результаты обработки таблиц в наглядной форме. При создании диаграммы можно выделить три этапа: создание таблицы, описание графика на бумаге, описание графика в электронной таблице и его использование.

Хорошо разработанная таблица содержит все элементы, необходимые для описания графика. При описании графика на бумаге необходимо установить соответствие между элементами таблицы и графиком (при достаточном навыке этот этап не обязателен). Если на таблице нет каких-то элементов, то их необходимо описать вне пределов таблицы.

График включает, обычно, следующие элементы: заголовок, обозначение осей, разметку по осям, описание меток, числовые данные на графике. Кроме того, необходимо определить тип диаграммы, наиболее подходящий для имеющихся данных.

Для построения графиков и диаграмм в электронной таблице используется Мастер диаграмм, который за четыре шага позволяет описать все элементы графика. Имеется возможность редактировать график после построения.

### 5.3.2. Задание

1. Постройте график функции одной переменной, согласно варианту задания (см. лабораторная работа №2).

2. Постройте диаграмму динамики ввода в строй жилых домов и объектов социально-культурного назначения (содержание таблицы можно выбрать по своему усмотрению) (рис.5.3.1.):

Ввод в строй жилых домов и объектов		(Заголовок)		
Наименование объекта	Единица измерения,	Годы <sup>6</sup>		
		1989	1990	1991
Жилые дома	м кв.	15035	14008	12138
Школы	м кв.	3810	1527	1240
Клубы, библиотеки	м кв.	15328	7630	6589

Рис. 5.3.1. Таблица ввода в строй жилых домов и объектов социально-культурного назначения

3. Выведите графики функций и диаграмму на печать.

#### Дополнительные задания

1. Разработайте таблицу и постройте по ней диаграмму. Тему для таблицы выбрать произвольно: объемы продаж товаров народного потребления; ввод в строй квартир и объектов социального и культурного назначения; запасы сырья и продовольствия; расход топлива на один километр автомобилей разных марок; динамика изменения численности населения и др.

2. Протабулируйте функцию одной переменной и постройте график функции.

3. Протабулируйте произвольную функцию двух переменных и постройте графики функций. Примечание: для построения графика используйте тип диаграммы Поверхность.

4. Рассчитайте и постройте диаграмму реакции опор твердого тела (Сборник задач курсовых работ по теоретической механике, с.5- 11 ).

5. Определите реакции опор и сил в стержнях плоской фермы. (Сборник задач курсовых работ по теоретической механике, с.11- 18 ).

6. Рассчитайте и постройте траекторию полета снаряда при следующих исходных данных: начальная скорость равна 100 м/с, угол стрельбы 45 градусов, начальная высота 100 м над уровнем моря, цель расположена на высоте 200 м над уровнем моря.

### 5.3.3. Указания и пример построения графика

Постройте графики функций  $2x^2-5$  и  $e^x$  на отрезке от -3 до 3. Отрезок разделите на десять равных частей. Пример построения графика приведен на листинге 5.3.1.

#### Порядок работы

1. Протабулируйте функцию на заданном отрезке (см. лабораторную работу №2). Шаг табуляции определяется как отношение длины отрезка табулирования функции к числу отрезков N:  $(\text{правая\_граница} - \text{левая\_граница})/N$ .

1. Выделите область исходных данных для построения графика функции  $2x^2-5$ , включая заголовки: A6:B17. Щелкните в стандартной панели инструментов кнопку Мастер диаграмм.

<sup>6</sup> Данные условные

3. Выполните первый шаг: выберите тип диаграммы - точечная и вид диаграммы - плавная кривая с метками точек на графике функции. Щелкните кнопку Далее.

4. Выполните второй шаг: если диапазон для построения графика не был указан, введите диапазон в строку ввода: Щелкните по строке ввода мышью и выделите требуемый диапазон. Щелкните кнопку Далее.

5. Выполните третий шаг: используя закладки диалогового окна, введите название диаграммы "График функции", название оси X, название оси Y. Установите, при необходимости, линии сетки основные и промежуточные, определите место расположения легенды, режим вывода на график числовых значений или категорий. Щелкните кнопку Далее.

6. Выполните четвертый шаг: укажите место размещения диаграммы: на отдельном листе или в текущем рабочем листе.

#### Листинг 5.3.1. Построение графика функций

	A	B	C	D	E	F	G
1	Построение графика функции						
2							
3	Интервал:	-3,0000	3,0000				
4	Число точек:	10	Шаг табуляции	0,6			
5							
6	Аргумент	$2 \cdot x^2 - 5$	$\exp(x)$				
7	-3	13,00	0,05				
8	-2,4	6,52	0,09				
9	-1,8	1,48	0,17				
10	-1,2	-2,12	0,30				
11	-0,6	-4,28	0,55				
12	0	-5,00	1,00				
13	0,6	-4,28	1,82				
14	1,2	-2,12	3,32				
15	1,8	1,48	6,05				
16	2,4	6,52	11,02				
17	3	13,00	20,09				



7. Добавьте к графику функции  $2x^2-5$  график функции  $e^x$ . Для этого необходимо выполнить следующее:

- щелкните правой кнопкой мыши линию графика функции  $2x^2-5$  и выберите в меню команду Исходные данные;

- в диалоговом окне Исходные данные щелкните кнопку Добавить;
- в строку ввода Имя введите щелчком мыши адрес ячейки - C6;
- в строку ввода Значения X введите диапазон значений аргумента A7:A17;
- в строку ввода Значения Y введите диапазон значений функции C7:C17;
- щелкните кнопку Ок.

**Примечание.** Можно строить одновременно несколько графиков функций. В рассмотренном примере графики функций  $2x^2-5$  и  $e^x$  строятся по очереди в учебных целях.

## Контрольные вопросы

1. Назовите основные элементы диаграммы.
2. Опишите порядок построения графиков функций с использованием мастера диаграмм.
3. Как добавить график функции на диаграмму?
4. Как изменить заголовок диаграммы или ее осей?
5. Как изменить стиль линий сетки?
6. Как поместить диаграмму в документ Word?
7. Каким образом перемещается диаграмма по рабочему листу?
8. Как изменить размеры диаграммы?
9. Какой тип диаграммы больше подходит для построения графиков функций?

## 5.4. Расчет деформации твердого тела

**Цель занятия:** закрепить знания, умения и навыки в использовании математических функций электронной таблицы и построения графиков функций.

**Время:** 2 часа.

**Литература:** Л1 с. 417 – 420, 424 – 428.

### 5.4.1. Общие сведения

При воздействии на тело сил оно деформируется, при этом в плоскостях, параллельных осям координат, возникают напряжения нормальные ( $\sigma$ ), касательные ( $\tau$ ) и результирующие ( $\rho$ ). Направление действия сил  $S_1$ ,  $S_2$ ,  $S_3$  совпадает с осями координат (рис.5.4.1).  $\alpha$ ,  $\beta$ ,  $\gamma$  углы, образованные осями координат, системы координат, связанной с объектом, и осями координат, связанной с площадкой деформации тела: в плоскости, параллельной оси  $X$ , -  $\beta$ , в плоскости, параллельной оси  $Y$ , -  $\alpha$  в плоскости, параллельной оси  $Z$  -  $\gamma$ .

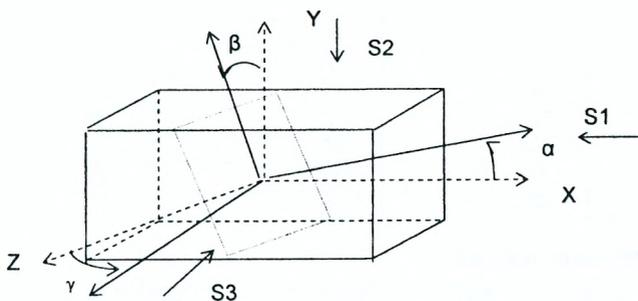


Рис. 5.4.1. К задаче "деформация твердого тела"

### Расчетные формулы:

а) в плоскостях, параллельных оси  $Z$ :

$$\sigma = s_1 \cdot \cos^2(\alpha) + s_2 \cdot \sin^2(\alpha),$$

$$\tau = (s_1 - s_2) / 2 \cdot \sin(2\alpha),$$

$$\rho = (s_1^2 \cos^2(\alpha) + s_2^2 \sin^2(\alpha))^{1/2}.$$

(5.4.1)

б) в плоскостях, параллельных оси Y:

$$\begin{aligned}\sigma &= s_1 \cdot \cos^2(\gamma) + s_3 \cdot \sin^2(\gamma), \\ \tau &= (s_1 - s_3) / 2 \cdot \sin(2\gamma), \\ \rho &= (s_1^2 \cdot \cos^2(\gamma) + s_3^2 \cdot \sin^2(\gamma))^{1/2}.\end{aligned}\quad (5.4.2)$$

в) в плоскостях, параллельных оси X:

$$\begin{aligned}\sigma &= s_2 \cdot \cos^2(\beta) + s_3 \cdot \sin^2(\beta), \\ \tau &= (s_2 - s_3) / 2 \cdot \sin(2\beta), \\ \rho &= (s_2^2 \cdot \cos^2(\beta) + s_3^2 \cdot \sin^2(\beta))^{1/2}.\end{aligned}\quad (5.4.3)$$

г) на площадку равно наклонную к трем главным осям (октаэдрические параметры):

$$\begin{aligned}\sigma_0 &= 1/3(s_1 + s_2 + s_3), \\ \tau_0 &= 1/3 \cdot ((s_1 - s_2)^2 + (s_2 - s_3)^2 + (s_3 - s_1)^2)^{1/2}, \\ \rho_0 &= 1/3 \cdot (s_1^2 + s_2^2 + s_3^2)^{1/2}.\end{aligned}\quad (5.4.4)$$

## 5.4.2. Задание

1. Рассчитать нормальные  $\sigma$  (сигма), касательные  $\tau$  (тау) и результирующие  $\rho$  (ро) напряжения в наклонных плоскостях при заданных значениях углов  $\alpha$  (альфа),  $\beta$  (бета),  $\gamma$  (гамма) согласно варианту задания.

2. Вести на печать Протокол решения и результаты.

3. Построить графики зависимости напряжений от значения угла  $\alpha$  при постоянном значении силы  $S_1$ .

### Варианты заданий

N	S1, МПа	S2, МПа	S3, МПа
1	20	-60	80
2	80	40	20
3	100	40	-60
4	-60	-80	-40
5	40	-50	30
6	-40	35	80
7	100	70	-20
8	100	40	-20
9	-50	-50	50
10	60	75	20

### Дополнительные задания

1. Протабулируйте функцию двух переменных на интервале от 0 до 90 градусов с шагом 10 градусов. Постройте график поверхности.

1.	$\operatorname{tg}(2x + y)$	2	$1 - \cos x \cdot \sin(y)$	3	$\operatorname{arctg}^2 x + \sin(y)$
4	$\sin 3x - \sin y$	5	$\cos 3x - \cos y$	6	$x^2 - \cos 2y$
7	$(1+y)^{1/2} - \cos x$	8	$\sin y + \sin 5x$	9	$3x/(1 - y \operatorname{tg} y)$
10	$3x^2/(2+x) \cos y$	11	$5x^3/(4-x \cos x)$	12	$1 - \cos 4x + 3y$
13	$1 - \cos 2x \cdot \sin(y)$	14	$3\sin^2 4x + y$	15	$\operatorname{tg}(x^2 + 2y)$
16	$x^3 - y \cos x$	17	$(4+x)^{1/2} + \cos y$	18	$\sin(x^2 - 2y)$
19	$2x/(3 - \operatorname{tg} y)$	20	$(x^2 \sin x)/(5+y)$	21	$\sin(x^2 + 5y)$

2. Введите текущую дату.
3. Определите по текущей дате: день недели, число, месяц, год.
4. Вычислите среднеквадратическое отклонение и дисперсию для ряда статистических данных (ряд данных в интервале от 1 до 10 сгенерируйте, используя функцию СЛЧИС()).

### 5.4.3. Указания к выполнению задания

1. Запишите в первой строке наименование работы.
2. Запишите во второй и третьей строках исходные данные, поясняя их текстом. Данные заключите в рамку.
3. Составьте таблицу по приведенной ниже форме и заключите ее в рамку:

A	B	C	D	E
4	значение угла $\alpha$	Сигма	Tau	Po
5				
6				
7				

4. Рассчитайте значение параметров Сигма, Tau и Po по формулам (5.4.1 – 5.4.4):
  - сгенерируйте в столбце B таблицы ряд значений угла  $\alpha$  от 0 до 180 градусов с шагом 15 градусов, другие углы остаются при этом постоянными;
  - запишите расчетные формулы для вычисления параметров в плоскостях, параллельных оси Z, в первую строку таблицы и скопируйте их в последующие строки. При вычислении значений  $\sin(x)$  и  $\cos(x)$  угол измеряется в радианах.
5. Повторите операции для остальных плоскостей.
6. Составьте таблицы зависимости касательных напряжений в плоскости, параллельной оси Z, от угла наклона плоскости (угол  $\alpha$ ) и направления действия силы  $s_1$ . Постройте графики.
7. Выведите результаты на печать.
8. Завершите работу. Выйдите из сети.

### Контрольные вопросы

1. Перечислите основные типы функций, используемые в электронной таблице.
2. Назовите основные математические функции электронной таблицы.
3. Назовите основные тригонометрические функции электронной таблицы.
4. Какие формы представления чисел используются в электронной таблице?
5. Какова точность вычислений и каков диапазон представления чисел в Excel?
6. Как вычислить логарифм числа по произвольному основанию?
7. Как возвести в степень число "e" - основание натурального логарифма?
8. Как вводится в таблицу дата?
9. Как изменить формат представления числа, даты?
10. Приведите формат функции ЕСЛИ.
11. Как вычислить среднее арифметическое по заданному диапазону клеток?
12. Перечислите функции для обработки статистических данных.
13. Как построить график функции одной переменной, двух переменных?
14. Для чего используется модуль Таблица подстановки, как он применяется?
15. Что такое легенда, что необходимо сделать при указании области данных, чтобы автоматизировать формирование легенды?

## 5.5. Работа с матрицами. Решение систем линейных уравнений

**Цель занятия:** приобрести практические навыки работы с матрицами и решения систем линейных алгебраических уравнений, используя возможности электронной таблицы Excel.

**Время:** 2 часа.

**Литература:** Л1 с. 428 – 432.

### 5.5.1. Общие сведения

Электронная таблица позволяет выполнять линейные преобразования матриц: умножение, деление матриц на число, прибавление или вычитание чисел, а также операции над матрицами: сложение, умножение матриц, транспонирование, вычисление определителей. Средствами Excel можно решать и системы линейных алгебраических уравнений. Для этой цели электронная таблица имеет ряд функций для работы с матрицами:

МОПРЕД(массив) – вычисление определителя матрицы;

МОБР(массив) – вычисление обратной матрицы;

МУМНОЖ(массив; массив) – умножение матриц;

ТРАНСП(массив) – транспонирование матриц.

Операции над матрицами в Excel выполняются по правилам математики.

Для решения **систем линейных уравнений** применяют аналитические методы.

Электронная таблица Excel не имеет функций для решения систем уравнений, формулы для вычисления матриц необходимо формировать самостоятельно, используя известные методы, например метод Крамера или метод Гаусса (метод исключения переменных). Однако используя встроенные функции МОБР, МУМНОЖ и МОПРЕД, эти операции выполняются достаточно легко. Например, можно воспользоваться формулой вычисления вектора неизвестных через обратную матрицу и вектор свободных членов:  $\vec{X} = A^{-1} \bullet \vec{B}$ .

### 5.5.2. Задание

1. Выполните следующие преобразования матриц размером 3x3:

$A^*C$ ;  $A^T$ ;  $A^{-1}$ ;  $A+B$ ;  $A^*B$ .

2. Ввести матрицу A размером 4x4. Вычислить определитель матрицы. Вычислить матрицу, обратную матрице A.

3. Выполнить преобразования  $(A^*B)^T C C^{-1}$ , где A и C – матрицы одной размерности, B – вектор, число элементов в котором равно числу столбцов в матрице A.

4. Решить систему линейных алгебраических уравнений согласно варианту задания двумя методами: методом Крамера, матричным методом с использованием обратной матрицы.

#### Варианты заданий к п. 4

N	Система уравнений	N	Система уравнений
1	$\begin{cases} x + 3y - z = 1 \\ 2x - y + 3z = 2 \\ 4x + 5y + z = 4 \end{cases}$	2	$\begin{cases} x + 2y + 3z = 5 \\ 2x + y - z = 14 \\ 4x + 5y + 5z = 14 \end{cases}$

N	Система уравнений	N	Система уравнений
3	$\begin{cases} 2x - y + z = 2 \\ x + y - 2z = 1 \\ 4x + y - z = 1 \end{cases}$	4	$\begin{cases} 7x + 2y + 2z = 6,1 \\ 2x + 4y + 4,1z = 5,5 \\ x - 2y + 3,1z = 2,6 \end{cases}$
5	$\begin{cases} 5x + y + z = 6 \\ x - 4y + z = 2 \\ 3x + 2y + 8z = 10 \end{cases}$	6	$\begin{cases} 7,9x + 5,6y + 5,7z = 6,68 \\ 8,5x - 4,8y + 0,8z = 9,95 \\ 4,3x + 4,2y - 3,2z = 8,6 \end{cases}$
7	$\begin{cases} x + y - 3z = -2 \\ 2x + y - 2z = 2 \\ x + y + z = 6 \end{cases}$	8	$\begin{cases} 7,4x + 2y + 2z = 6,1 \\ 2x + 4y + 4,1z = 5,5 \\ x - 2y + 3,5z = 2,6 \end{cases}$
9	$\begin{cases} 3x + 2y - z = 4 \\ 2x - y + 3z = 9 \\ x - 2y + 2z = 3 \end{cases}$	10	$\begin{cases} 2x + y + 3z = 7 \\ 2x + 3y + z = 1 \\ 3x + 2y + z = 6 \end{cases}$
11	$\begin{cases} 2x - y + 2z = 3 \\ x + y + 2z = -4 \\ 4x + y + 4z = -3 \end{cases}$	12	$\begin{cases} 3x + y + z = 12 \\ x + 2y + 4z = 6 \\ 5x + y + 2z = 3 \end{cases}$
13	$\begin{cases} 3x - 2y + 4z = 12 \\ 3x + 4y - 2z = 6 \\ 2x - y - z = -9 \end{cases}$	14	$\begin{cases} 8x + 3y - 6z = -4 \\ x + y - z = 2 \\ 4x + y - 3z = -5 \end{cases}$
15	$\begin{cases} 10x + y + z = 12 \\ 2x + 10y + z = 13 \\ 2x + 2y + 10z = 14 \end{cases}$	16	$\begin{cases} 4x + 0,24y - 0,08z = 8 \\ 0,09x + 3y - 0,15z = 9 \\ 0,04x + 0,08y + 4z = 20 \end{cases}$

### 5.5.3. Указания и примеры выполнения задания

#### Операции над матрицами

Алгоритм выполнения операций над матрицами сводится к следующей последовательности действий:

- выделить ячейку или область, если результатом выполнения операции будет матрица, куда будет помещаться результат вычисления;
- ввести в строку ввода символ "=";
- ввести в строку ввода первый операнд, например, область матрицы В (B5:C6);
- ввести в строку ввода символ операции, например, оператор сложения "+";
- ввести в строку ввода второй операнд, например, адрес числа а - В3. Например, для первого примера на листинге 6.6. получим выражение {=B5:C6+B3};
- нажать комбинацию клавиш Ctrl+Shift+Enter для вставки формулы в область.

Примеры операций с матрицами приведены на листинге 5.5.1. Обратите внимание на разные результаты, получаемые при умножении матриц с использованием оператора умножения "\*\*", и с использованием функции МУМНОЖ. В первом случае каждый элемент матрицы результата равен произведению соответствующих элементов сомножителей, во втором случае каждый элемент матрицы вычисляется по формуле

$$C(i, k) = \sum_{j=1}^m A(i, j) * b(j, k), \text{ где } m - \text{число столбцов в матрице } A.$$

## Решение систем линейных уравнений

**Пример 5.5.1.** Решить систему линейных алгебраических уравнений матричным способом (Листинг 5.5.2):

Листинг 5.5.1. Операции с матрицами												
	A	B	C	D	E	F	G	H	I	J	K	L
1	Прибавление числа к матрице				Умножение матрицы на число				Сложение матриц			
2												
3	a=	2,543			a1=	7,345			C=B+B1	3	6	
4										8	11	
5	B=	2	4		B1=	1	2					
6		5	7			3	4		Умножение матриц			
7												
8	B+a=	5,086	10,17		B1*a1=	7,345	14,69		D=B*B1	2	8	
9		12,715	17,8			22,035	29,38			15	28	
10												
11	Транспонирование матриц				Использование функции МУМНОЖ							
12												
13	Bt=	2	5					МУМНОЖ(B;B1)=		14	20	
14		4	7							26	38	

$$\begin{cases} 2x - 5y + 6z = 8 \\ 2x - 4y + 7z = 5 \\ 3x - 6y + 8z = -4 \end{cases} \quad (5.5.1)$$

### Решение.

1. Введите в ячейки B6 – D8 значения коэффициентов при неизвестных.
2. Введите в ячейки F6 – F8 значения свободных членов системы уравнений.
3. Выделите диапазон ячеек B12: D14 и введите формулу МОБР(B6:D8), для завершения операции ввода нажмите комбинацию клавиш [Ctrl-Shift-Enter].

Выделите диапазон ячеек F12:F14 и введите формулу МУМНОЖ(B12:D14;F6:F8). Для завершения ввода формулы нажмите комбинацию клавиш [Ctrl-Shift-Enter]. В ячейках F12 – F14 появятся значения корней уравнения.

**Пример 6.5.** Решить систему линейных алгебраических уравнений (5.5.1.) методом Крамера (Листинг 5.5.3).

### Решение.

1. Внесите в таблицу расширенную матрицу, то есть запишите в ячейки A3:D5 электронной таблицы коэффициенты при неизвестных и свободные члены.

2. Запишите в ячейки A8:C10 главный определитель, используя в качестве исходных данных адреса ячеек из расширенной матрицы. Этот метод предпочтительнее простого копирования, так как при изменении данных в ячейках расширенной матрицы автоматически изменяются и значения в ячейках дополнительного определителя;

3. Скопируйте два раза коэффициенты матрицы из ячеек A8:C10 в ячейки A12:C14 и A16:C18;

4. Сформируйте из копий матрицы главного определителя дополнительные определители путем замены коэффициентов при неизвестных на вектор свободных членов. При этом так же как и в пункте 2 ссылаться на адреса ячеек D3 – D5;

**Листинг 5.5.2. Решение системы линейных алгебраических уравнений матричным способом**

	A	B	C	D	E	F	G
1	Лабораторная работа №5						
2							
3	Тема: Решение системы линейных алгебраических уравнений матричным способом: $X=A^{-1}B$						
4							
5	Матрица Коэффициентов				Вектор свободных членов		
6		2	-5	6		8	
7		2	-4	7		5	
8		3	-6	8		-4	
9							
10	Обратная матрица				Результат		
11							
12		-2	-0,8	2,2		-28,8	
13		-1	0,4	0,4		-7,6	
14		0	0,6	-0,4		4,6	

5. Запишите напротив первой строки матрицы коэффициентов главного определителя в ячейку F3 расчетную формулу для его вычисления: МОПРЕД(A3:C5) ;

6. Скопируйте расчетную формулу из ячейки F3 в ячейки F8, скорректируйте ее, а затем скопируйте в ячейки F12 и F16;

7. Запишите формулы для вычислить неизвестных как отношение соответствующих дополнительных определителей к главному определителю в ячейки I3, I4 и I5.

**Листинг 5.5.3. Пример решения системы линейных алгебраических уравнений методом Крамера**

	A	B	C	D	E	F	G	H	I
1	Решение системы уравнений методом Крамера								
2	Расширенная матрица				Определители		Результат		
3	2	-5	6	8		-5		X=	-28,8
4	2	-4	7	5				Y=	-7,6
5	3	-6	8	-4				Z=	4,6
6									
7	1-й дополнительный определитель								
8	8	-5	6			144			
9	5	-4	7						
10	-4	-6	8						

11	2-й дополнительный определитель						
12	2	8	6		38		
13	2	5	7				
14	3	-4	8				
15	3-й дополнительный определитель						
16	2	-5	8		-23		
17	2	-4	5				
18	3	-6	-4				

### Контрольные вопросы

1. Назовите функции электронной таблицы для работы с матрицами.
2. Каким образом выполняются операции над матрицами в электронной таблице?
3. Как решить систему линейных алгебраических уравнений матричным способом?
4. Опишите порядок решения САЛУ методом Крамера.
5. Как выполнить транспонирование матрицы.

## 5.6. Списки (базы данных) в электронной таблице

**Цель занятия:** ознакомиться с принципами создания и использования баз данных в электронных таблицах.

**Время:** 2 часа.

**Литература:** Л1 с. 455 – 465.

### 5.6.1. Общие сведения

#### Создание списка (базы данных)

Список рекомендуется создавать на отдельном листе. В этом случае программа быстрее использует команды сортировки и фильтр, а также исключается возможность испортить другие данные. Списки в электронной таблице представляют собой однотабличную базу данных.

Для создания списка может использоваться любой диапазон ячеек. Тогда каждый столбец диапазона считается полем, которое может содержать строку длиной до 255 символов. Соответственно каждая строка диапазона будет считаться записью. Для создания и работы со списками Excel имеет специальные команды, функции и методы.

Щелкните правой клавишей мыши по ярлычку номера страницы и введите название базы данных (по ее содержанию).

В первую строку диапазона, отведенного для создания списка, записываются имена полей. Имена полей должны быть, по возможности, простыми, краткими и описательными, при этом они не могут занимать более одной строки таблицы. В последующие строки записываются данные. Первая запись не должна ничем отделяться от строки заголовков, в списке не должно быть одинаковых записей, пустых строк и колонок. Итоговые данные, помещаемые внизу таблицы, должны быть отделены от базы данных хотя бы одной строкой.

Для работы со списками используются команды **Сортировка**, **Фильтр** и **Форма**, которые находятся в пункте главного меню **Данные**.

Команда **Форма** используется для создания списка, добавления данных в список, просмотра списка и поиска данных по заданному критерию.

Команда **Сортировка** позволяет отсортировать выделенный диапазон по значениям одного, двух или трех полей. Сортировку можно проводить по возрастанию или убыванию значения соответствующего поля. В случае сортировки базы данных столбцы будут фигурировать под названиями полей, а в случае сортировки простого списка – под названиями столбцов. Сортировка может выполняться по строкам или столбцам, а также по содержанию. Для использования этих возможностей необходимо в диалоговом окне **Сортировка диапазона** щелкнуть кнопку **Параметры**.

Команда **Фильтр**. Команда Фильтр содержит опции **Автофильтр**, **Отобразить все** и **Расширенный фильтр**.

При выборе опции **Автофильтр** на каждом поле появляется кнопка раскрывающегося списка (рис.5.6.1.). Если щелкнуть мышью по кнопке раскрывающегося списка, то открывается список параметров (рис.5.6.2.). Выбор требуемого параметра осуществляется щелчком мыши. Параметр **Условие** выводит на

Температура	Влажность	Давление
23	89	756
14	68	765
20	68	768
13	65	760
20	60	765

Рис.5.6.1. База данных, режим Автофильтр

экран диалоговое окно **Пользовательский автофильтр**, которое позволяет объединить два параметра по логическому условию И или ИЛИ (рис.5.6.3.). В левых раскрывающихся списках выбираются логические условия для выбора числовой или текстовой информации, в правых раскрывающихся списках выбираются из списка или вводятся значения параметров для отбора. При указании значений параметров допускается использование маски: вопросительный знак или звездочка. Символ "?" заменяет один знак в указанной позиции. Например, "ию?ь" совпадает с "июнь" и "июль". Символ \* заменяет все слово или его часть. Например, "авто\*" соответствует "автомобиль", "автокар" и т. д.

(Все)
(Первые десять)
(Условие...)
15
20
34
...
(Пустые)
(Не пустые)

Рис.5.6.2. Список параметров

После применения фильтра все записи, не удовлетворяющие заданным критериям, убираются с экрана. Для отображения всех записей необходимо применить параметр **Все**.

**Расширенный фильтр** предоставляет пользователю дополнительные возможности по выбору критериев и формированию результатов: список можно фильтровать на месте или скопировать результат в указанный диапазон; условия для выбора могут быть заданы в отдельном диапазоне рабочего листа; можно использовать при выборе только уникальные записи.

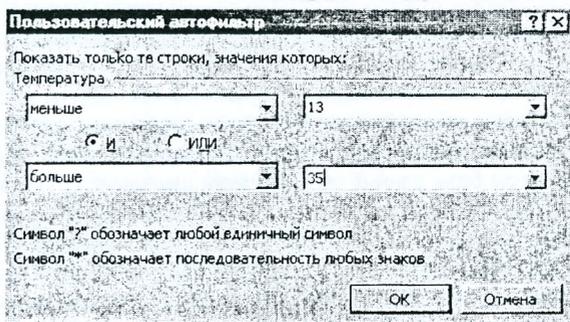


Рис. 5.6.3. Пользовательский автофильтр

Блок критериев для расширенного фильтра содержит условия для поиска и выборки данных. Он может располагаться в любом месте электронной таблицы. Блок критериев состоит из двух или более строк. В первой строке задаются имена полей, а в последующих строках – критерии для поиска. Критерием может быть текстовая или числовая константа, логическая функция или логическое выражение. Если критерий содержит несколько строк, то считается, что эти строки связаны функцией ИЛИ. Если блок критерия содержит несколько полей, то считается, что эти поля связаны функцией И. Примеры блоков критериев для расширенного фильтра приведены на рис. 5.6.4.

В блоке критериев А2:А5 три наименования объединены по схеме ИЛИ. То есть при наличии на складе указанных товаров все они будут включены в выходной список. В блоке критериев С2:Е3 три параметра объединены по схеме И. Это значит, что для выбора товара должны быть выполнены все три условия: в базе данных будет отыски-

	А	В	С	Д	Е
1	Схема ИЛИ		Схема И		
2	<b>Наименование</b>		<b>Наименование</b>	<b>Дата поступления</b>	<b>Цена</b>
3	Доска		Бумага	01.01.00	<10000
4	Бумага				
5	Картон		Схема ИЛИ		
6			<b>Наименование</b>	<b>Поставщик</b>	<b>Цена</b>
7			Картон		
8				Барнаул	
9					<200000

Рис. 5.6.4. Примеры блоков критериев расширенного фильтра

ваться бумага, поступивший на склад первого января 2000 года по цене меньше 10000 рублей. В блоке критериев С6:Е9 три разных критерия объединены по схеме ИЛИ, то есть будут отыскиваться картон, все товары, поступившие из Барнаула и все товары по цене меньше 200000 рублей.

Блок критериев для расширенного фильтра рекомендуется задавать выше базы данных (списка), отделяя его от базы данных одной строкой (листинг 5.6.1).

Работа с "базой данных" состоит из двух этапов: *создание базы данных; ведение и использование базы данных*. Для создания базы данных используется команда **Данные, Форма**, хотя можно создать базу данных как простую таблицу. На втором этапе осуществляются операции редактирования, дополнения, удаления записей, процедуры поиска и выборки данных. При вставке новых записей или удалении существующих записей размеры базы данных автоматически изменяются. Поиск данных осуществляется командой **Форма**, а извлечение записей - командой **Фильтр** пункта меню **Данные**.

В примере на листинге 5.6.1. первый блок критериев А4:А6, объединяет критерии по схеме ИЛИ, второй блок критериев С4:Д6, объединяет два критерия (Город и Температуру) по строке по схеме И и два критерия (Брест и Гродно) по схеме ИЛИ. Третий блок критериев F4:H5 объединяет три критерия по схеме И.

Листинг 5.6.1. Пример базы данных Метеоусловия

	A	B	C	D	E	F	G	H	I
1	<b>База данных Метеоусловия</b>								
2									
3	<b>Блоки критериев 1, 2</b>					<b>Блок критериев 3</b>			
4	Город		Город	Температура		Город	Температура	Влажность	
5	Брест		Брест	>20		Брест	>18	<86	
6	Гродно		Гродно	>18					
7									
8						<b>Выходной блок 1</b>			
9	<b>База данных Метеоусловия</b>					Город	Температура	Влажность	Давление
10	Город	Температура	Влажность	Давление		Брест	23	89	756
11	Брест	23	89	756		Брест	19	85	765
12	Могилев	23	75	758		Брест	18	85	766
13	Минск	13	83	760		Гродно	18	83	767
14	Минск	20	87	760		Гродно	17	89	768
15	Могилев	25	70	760		Гродно	20	76	768
16	Брест	19	85	765					
17	Витебск	14	78	765		<b>Выходной блок 2</b>			
18	Минск	21	75	765		Город	Температура		
19	Могилев	20	79	765		Брест	23		
20	Брест	18	85	766		Гродно	20		
21	Гродно	18	83	767					
22	Витебск	15	89	768		<b>Выходной блок 3</b>			
23	Витебск	15	90	768		Город	Температура	Влажность	Давление
24	Гродно	17	89	768		Брест	19	85	765
25	Гродно	20	76	768					

## 5.6.2. Задание

1. Создать базу данных "Успеваемость" по следующей форме:

№ груп-пы	Фамилия	Дата рождения	Физика	Мате-матика	Выч. техника	Физкуль-тура	Ср. балл	Сти-пендия

Стипендию подсчитать следующим образом: если средний балл меньше 3, то стипендия не начисляется, если средний балл больше или равен 3, то начисляется 3 базовых величины, если средний балл больше или равен 3,5 – четыре базовых величины, а если средний балл больше 4,5 то начисляется 5 базовых величин.

2. Произвести выборку данных по следующим критериям:

- по двум фамилиям, используя команду Автофильтр;
- по номеру группы и среднему баллу, используя команду Автофильтр.

Результаты перекопировать на отдельный лист.

3. Выбрать, используя расширенный фильтр с формированием отдельного выходного блока:

- всех отличников;
  - студентов, которые учатся только на 4 и 5;
  - неуспевающих студентов;
  - студентов, которые имеют право участвовать в выборах (18 лет на день выборов).
4. Вывести результаты на печать.

### Дополнительные задания

1. Создать телефонный справочник с полями: Фамилия, Имя, Отчество, Адрес, номер телефона. Создать блоки критериев для поиска адресата по номеру телефона, по фамилии, по домашнему адресу.

2. Создать базу данных для хранения сведений и поиска необходимой информации о своих друзьях, знакомых, деловых партнерах.

3. Создать базу данных "Склад" со следующими полями: Наименование, Единица измерения, Дата поступления, Количество, Цена, Стоимость, Норма запаса.

4. Создать базу данных для хранения сведений о движении пригородных поездов по станции Брест со следующими полями: Номер поезда, Направление (по конечной станции), Время отправления, Время прибытия на конечную станцию, Стоимость проезда.

5. Создать базу данных для хранения сведений о проектах с полями: Проектная организация, Заказчик, Номер проекта, Наименование проекта, Дата выдачи задания, Разработчик проекта, Дата сдачи готового проекта, Отметка о прохождении вневедомственной экспертизы, Число страниц в пояснительной записке, Стоимость проекта, Место хранения.

## 5.6.3. Пример создания и использования базы данных

Пример использования базы данных приведен на листинге 5.6.2.

Блоки критериев для расширенного фильтра размещаются в верхней части таблицы. Выборка по заданным критериям размещается справа от базы данных. Первый блок критериев A4:D7 на листинге реализует схему ИЛИ: выбираются все студенты группы а6, у которых хотя бы по одному предмету оценка меньше 3. Второй блок критериев F4:G6 реализует схему И-ИЛИ - выбираются студенты групп а7 или а8, у которых средний балл больше 4,5.

Листинг 5.6.2. База данных успеваемость

	A	B	C	D	E	F	G
1	Создание и использование Базы данных						
2	Миним. Зарплата		7600				
3	Блоки критериев для расширенного фильтра						
4	Номер группы	Физика	Математика	Информатика		Номер группы	Средний балл
5	a6	<3				a7	>4,5
6	a6		<3			a8	>4,5
7	a6			<3			
8							
9	Номер группы	Фамилия	Физика	Математика	Информатика	Средний балл	Стипендия
10	a6	Кириянов	5	5	4	4,67	38000
11	a6	Петров	3	4	4	3,67	30400
12	a6	Приходько	4	4	3	3,67	30400
13	a7	Иванов	3	4	3	3,33	22800
14	a7	Измайлов	2	3	2	2,33	0
15	a7	Майский	4	4	3	3,67	30400
16	a7	Перелет	5	2	3	3,33	22800
17	a8	Павлюченя	3	3	2	2,67	0
18	a8	Сибиряк	3	3	4	3,33	22800
19	a8	Сидоревич	4	4	5	4,33	38000
20	a8	Шеремет	4	4	4	4,00	38000
21	a9	Западнев	5	5	4	4,67	38000

### Контрольные вопросы

1. Для чего предназначены базы данных?
2. Что такое поле, запись, критерий?
3. Чем отличается база данных от обычной таблицы или списка?
4. Как создать базу данных?
5. Как объединяются критерии по схеме И?
6. Как объединяются критерии по схеме ИЛИ?
7. Для чего предназначена команда Фильтр?
8. Как задаются условия при использовании автофильтра?
9. Как задаются условия при использовании расширенного фильтра?
10. Как осуществляется сортировка баз данных?

## 5.7. Разработка функций пользователя

**Цель занятия:** приобрести практические навыки в создании и использовании функций пользователя для решения типовых задач с использованием встроенной системы программирования Visual Basic for Applications.

**Время:** 4 часа.

**Литература:** Л1 с. 452 – 454.

### 5.7.1. Общие сведения

В приложение Excel интегрирована неполная версия языка программирования Visual Basic под названием Visual Basic, Application Edition или Visual Basic for Applications, которая обладает частью стандартных функциональных возможностей. Кроме того, эта специальная версия поддерживает объекты, которые позволяют обращаться к содержимому ячеек и управлять приложением Excel. Программа в Visual Basic for Application представляет собой набор событийно управляемых процедур и функций.

Функция представляет собой фрагмент программы, выполняемый как один блок, и имеет обязательно заголовок - **Function** и завершающую инструкцию – **End Function**, между которыми и находится собственно выполняемая программа. Синтаксис функции пользователя:

Function Имя\_функции(аргументы)

<Тело функции>

End Function

Для создания функции пользователя необходимо ввести команду **Сервис, Макрос, Редактор Visual Basic**. В редакторе VB выбрать команду **Вставка, Модуль**, а затем **Вставка, Процедура**. В окне диалога установить переключатель **Function** и установить область видимости **Public**. Написать текст программы, сохранить программу командой **Файл (File), Сохранить (Save)** и выйти из редактора (**Close and Return to Microsoft Excel**).

Для использования функций пользователя необходимо выделить ячейку и ввести функцию, указав аргументы. При вводе функций работы с массивами необходимо выделить область ячеек, куда будет помещаться массив, ввести функцию в первую ячейку массива и нажать комбинацию клавиш **Ctrl+Shift+Enter**. Для ввода функций пользователя можно воспользоваться **Мастером функций**, категория "**Пользовательские функции**".

### 5.7.2. Задание

1. Разработать и отладить функцию пользователя для решения следующих задач:
  - табулирование функции на заданном отрезке и построение ее графика. При построении графика функции использовать точечный тип диаграммы;
  - табулирование разветвляющейся функции и построение ее графика;
  - табулирование функции двух переменных и построение ее графика. При построении графика функции использовать тип диаграммы "поверхность";
2. Разработать функцию пользователя в соответствии с вариантом задания.

#### Варианты заданий для п.2

1. Разработать функцию для решения квадратного уравнения в области действительных чисел.
2. Вычислить сумму и произведение конечного ряда чисел в соответствии с вариантом задания.
3. Вычислить сумму бесконечного ряда в соответствии с вариантом задания.

### Задачи к вариантам заданий

Вычислить сумму конечного ряда

1. $S = \sum_{i=1}^n \cos\left(\frac{\pi x}{i}\right)$	2. $S = \sum_{i=1}^n (-1)^{i+1} \frac{\sin(\pi x)}{i}$	3. $S = \sum_{i=1}^n x^i \sin\left(\frac{i\pi}{3}\right)$
4. $S = \sum_{i=1}^n (x^i \cos\left(\frac{\pi}{3}\right))^i$	5. $S = \sum_{i=1}^n \frac{x^{i+1}}{i+1}$	6. $S = \sum_{i=1}^n \frac{1}{2i+1} \frac{(x-1)^{2i+1}}{x+1}$
7. $S = \sum_{i=1}^n (-1)^i \frac{\cos(ix)}{i^2}$	8. $S = \sum_{i=1}^n (-1)^{i+1} \frac{x^{2i+1}}{i^2-1}$	9. $S = \sum_{i=1}^n \frac{\sin(2i-1)x}{2i-1}$
10. $S = \sum_{i=1}^n \frac{\cos(2ix)}{i^2-1}$	11. $S = \sum_{i=1}^n (-1)^i \frac{x^{2i+1}}{i^2+1}$	12. $S = \sum_{i=1}^n (-1)^i \frac{(1+x)^{2i}}{i}$
13. $S = \sum_{i=1}^n \frac{x^i \cos(2i)}{2i+1}$	14. $S = \sum_{i=1}^n \frac{x^{2i}+1}{3i^2}$	15. $S = \sum_{i=1}^n \frac{\cos(ix/3)}{i^2}$

Вычислить сумму бесконечного ряда с точностью  $E=0.001$ :

1. $S = \sum_{i=1}^{\infty} \frac{(-1)^{i-1} x^{2i-1}}{(2i-1)!}$	2. $S = \sum_{i=1}^{\infty} \frac{\ln^i 3}{i!} x^i$	3. $S = \sum_{i=1}^{\infty} (-1)^i \frac{x^{2i+1}}{(2i+1)!}$
6. $S = \sum_{i=1}^{\infty} \frac{\cos(2ix^2)}{(2i+1)!}$	5. $S = \sum_{i=1}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$	6. $S = \sum_{i=1}^{\infty} \frac{\cos(ix)}{(i+2)!} x^i$
7. $S = \sum_{i=1}^{\infty} \frac{2i+1}{i!} x^{2i}$	8. $S = \sum_{i=1}^{\infty} (-1)^i \frac{2i^2+1}{(2i)!} x^{2i}$	9. $S = \sum_{i=1}^{\infty} \frac{i^2+1}{i!} \frac{x^i}{2}$
10. $S = \sum_{i=1}^{\infty} \frac{i^2}{(2i+1)!} x^i$	11. $S = \sum_{i=1}^{\infty} \frac{x^{2i+1}}{(2i+1)!}$	12. $S = \sum_{i=1}^{\infty} \frac{x^{2i}}{(2i)!}$
13. $S = \sum_{i=1}^{\infty} \frac{x^{i+1} + \sin(xi)}{(3i)!}$	14. $S = \sum_{i=1}^{\infty} \frac{\cos(ix)^2}{(3i+1)!}$	15. $S = \sum_{i=1}^{\infty} \frac{x^{2i} \sin(ix)}{(2i+1)!}$

### 5.7.3. Примеры выполнения задания

1. Создайте функцию пользователя Tab1perem и сохраните ее на диске. Аргументами функции являются число шагов - N, начальное значение аргумента - X и шаг - Dx. Пример текста программы приведен на листинге 5.7.1. После сохранения функция попадает в список Мастера функций, категория Пользовательские.

**Внимание.** Динамический массив объявляется дважды. Первый раз массив объявляется в разделе Общие без указания размерности с помощью оператора Public или Dim, а второй раз – в теле функции с указанием размерности с помощью оператора ReDim (так же как и в Visual Basic 6.0).

2. Пример табулирования функции одной переменной (листинг 5.7.1).

Протабулировать функцию  $y=x^2$  на отрезке [1,4] с шагом 0,5.

– введите в ячейки A2:B4 исходные данные;

– выделите область A6:B12 и выберите функцию Tab1perem из Мастера функций.

Укажите аргументы и нажмите комбинацию клавиш Ctrl+Shift+Enter.

Листинг 5.7.1. Табулирование функции одной переменной

	A	B	C	D	E	F	G
1	Исходные данные			Текст программы			
2	N=	7		Dim A() As Single			
3	X=	1		Private Function Tab1perem(n As Byte, _			
4	Dx=	0,5		x As Single, dx As Single) As Variant			
5	Аргумент	Функция		Dim k As Byte			
6	1	1		ReDim A(n,1) As Single			
7	1,5	2,25		For k = 0 To n - 1			
8	2	4		A(k, 0) = x: A(k, 1) = x * x			
9	2,5	6,25		x = x + dx			
10	3	9		Next k			
11	3,5	12,25		Tab1perem = A			
12	4	16		End Function			

3. Пример табулирования функции двух переменных.

Протабулировать функцию двух переменных  $Z=x*y$  при  $x$  изменяющемся от 1 до 7 с шагом 0,5 и  $y$  изменяющемся от 2 до 4 с шагом 0,5. Текст программы и решение приведены на листинге 5.7.2.

Листинг 5.7.2. Табулирование функции двух переменных

	A	B	C	D	E	F	G	H	I
2	Исходные данные				Текст программы				
3		N=	7,00		Public B() As Single				
4		M=	4,00		Private Function Tab2perem(n As Byte, m As _				
5		X=	1,00		Byte, x As Single, y As Single, dx As Single, _				
6		Y=	2,00		dy As Single) As Variant				
7		Dx=	0,50		Dim i As Byte, j As Byte				
8		Dy=	0,50		ReDim B(n, m) As Single				
9	Таблица функции двух переменных				For i = 0 To n - 1				
10	2,00	3,75	6,00	8,75	For j = 0 To m - 1				
11	12,00	15,75	20,00	24,75	B(i, j) = x * y				
12	30,00	35,75	42,00	48,75	x = x + dx: y = y + dy				
13	56,00	63,75	72,00	80,75	Next j				
14	90	99,75	110,00	120,75	Next i				
15	132	143,75	156,00	168,75	Tab2perem = B				
16	182	195,75	210,00	224,75	End Function				

### Контрольные вопросы

1. Опишите структуру функции пользователя.
2. Как создать функцию пользователя?
3. Как используется функция пользователя?
4. Как ввести функцию пользователя в массив?

## 5.8. Решение алгебраических и трансцендентных уравнений

**Цель занятия:** приобрести практические навыки решения алгебраических уравнений в электронной таблице используя известные математические зависимости и встроены функции электронной таблицы.

**Время:** 2 часа.

**Литература:** Л1 с. 440 – 447.

### 5.8.1. Общие сведения

Нелинейные уравнения вида  $F(x)=0$  принято называть *алгебраическими*, если они содержат только алгебраические функции и *трансцендентными*, если они содержат другие функции (тригонометрические, показательные, логарифмические и т. д.).

*Корнем уравнения*  $f(x)=0$ , где  $f(x)$  - функция непрерывная и дифференцируемая на отрезке  $[a,b]$  и в граничных точках, называется всякое число  $c$ , принадлежащее отрезку  $[a,b]$ , такое, что  $f(c)=0$ .

Процесс нахождения корней состоит из двух этапов: *отделения корней* и *уточнения значения корней* на отрезках отделения с заданной точностью.

*Отделением корней* называется процесс выделения из области определения функции  $f$  отрезков  $[a,b]$ , в каждом из которых содержится один и только один корень уравнения  $f(x)=0$ .

Корни уравнения могут находиться на интервалах, определяемых переменной знака функции, между критическими точками. К критическим относятся точки, в которых производная от функции  $f(x)$  обращается в нуль, а также граничные точки.

В электронной таблице отделение корней можно выполнить путем табулирования функции с некоторым, достаточно малым, шагом. Областями отделения корней будут значения аргументов, между которыми происходит смена знака функции. Можно выполнить отделение корней уравнения также по графику функции.

Под *уточнением значения* корня с заданной точностью  $h$  понимают сужение границ отрезка  $[a,b]$  до длины, не превосходящей  $h$ .

Для уточнения значения корня в Excel могут применяться все известные методы одномерной поисковой оптимизации. В настоящей работе рассмотрены три метода:

- простое табулирование;
- метод простых итераций;
- метод Ньютона (метод касательных).

Метод простого табулирования трудоемкий и малоэффективный.

Метод простых итераций более эффективен. Он сходится, если  $f'(x)<0$ . Для его реализации необходимо функцию  $f(x)=0$  преобразовать к рекуррентному виду

$$x_{i+1}=\varphi(x_i) \quad (5.8.1)$$

Табулировать необходимо правую часть выражения (5.8.1). Для первой формулы в качестве аргумента используется начальное приближение  $X_0$ . Начальное приближение выбирается произвольно (как правило, одна из границ отрезка отделения), для последующих формул - значение корня на предыдущем шаге, т. е.  $f(x_i)$ . Условием окончания процедуры вычисления будет выполнение неравенства  $\varphi(x_{i+1}) - \varphi(x_i) < \epsilon$ .

Метод Ньютона - самый эффективный метод. Он обеспечивает сходимость за минимальное число шагов. Однако этот метод накладывает серьезные ограничения на вид функции. Функция должна быть дважды дифференцируема. Для поиска корня этим ме-

тодом, так же как и в методе простых итераций, составляется рекуррентная формула:

$$x_{i+1} = x_i - f(x_i)/f'(x_i). \quad (5.8.2)$$

Табулировать необходимо правую часть выражения. Начальное приближение выбирается на одной из границ отрезка от деления корня. В качестве начального приближения  $x_0$  выбирается граница  $b$ , если  $f'(x) \cdot f''(x) > 0$ , и граница  $a$ , если  $f'(x) \cdot f''(x) < 0$ . Для первой формулы в качестве аргумента используется начальное приближение  $x_0$ , для последующих формул - значение корня на предыдущем шаге, т. е.  $f(x_i)$ . Условием окончания процедуры уточнения корня является достижение функцией значения  $f(x_{i+1}) < \epsilon$ .

Значение  $\epsilon$  можно принять в интервале от 0.0001 до 0.00001.

Все указанные методы могут быть реализованы с помощью функций пользователя, разработанных с помощью VBA.

## 5.8.2. Задание

1. Отделите корни уравнения на любом отрезке существования функции.
2. Уточните значение корня на одном из отрезков от деления тремя методами: простым табулированием, методом простых итераций, методом Ньютона. Начальное значение шага табулирования в первом методе примите равным 0.5, точность - 0.001.
3. Найдите корень уравнения с помощью команд Подбор параметра и Поиск решения.
4. Сделайте выводы о достоинствах и недостатках методов.

### Варианты заданий

N	Первое уравнение	Отрезок	Второе уравнение	Отрезок
1	$2^x - (x+1)^3 = 0$	-0.5, 0.5	$x - 2\sin(0.5+x) = 0$	0, 0.5
2	$2^x \cdot \ln x - 7 = 0$	1, 4.5	$x - \cos x = 0$	0.5, 1
3	$2x - 1,3^x = 0$	0.5, 1	$2x^{1/2} \cos(\pi x/2)$	0, 0.5
4	$2^x - 2,2x = 0$	0.5, 1	$x^2 + x e^x - 2 = 0$	-0.5, 0
5	$(x-1)^{1/2} - 1/x = 0$	1, 2	$x^2 - 4\sin x = 0$	1.5, 3
6	$e^x - e^{-3x} - 4x = 0$	1, 0	$3x + \cos x + 1 = 0$	-1.0, 0
7	$x^3 + 2,2x + 2^x = 0$	-0.5, 0	$2x + \sin x + 1 = 0$	-1, 0
8	$e^x - 6x - 3 = 0$	-0.5, 0	$3x - \cos x - 1 = 0$	2, 0
9	$2^x - 4x = 0$	0, 0.5	$x^2 + 0,5\sin x - 2 = 0$	1, 1.5
10	$x2^x = 1 - \sin x$	0, 0.5	$2 - x - \ln x = 0$	1.5, 2
11	$3 \sin x^{1/2} + 0,35x - 0,5$	2, 3	$0,25x^3 + x - 1,2 = 0$	0, 2, -3,8
12	$x + x^{1/2} + x^{1/3} - 2,5 = 0$	0.4, 1	$x - 1/(3 + \sin 3,6x) = 0$	0, 0.85
13	$0,1x^2 - x \ln x = 0$	1, 2	$3x - 4 \ln x - 1 = 0$	2, 4
14	$(1 - 0,4x^2)^{1/2}$	0, 1	$e^x + \ln x - 10x = 0$	3, 4
15	$3x - 14 + x^x - e^x = 0$	1, 3	$(1-x)^{1/2} - \operatorname{tg} x = 0$	0, 1
16	$x + \cos(x^{0,52} + 2) = 0$	0.5, 1	$3 \ln 2x + 6 \ln x - 5 = 0$	1, 3
17	$\sin x^2 + \cos x^2 - 10x = 0$	0, 1	$x^2 - \ln(1+x) - 3 = 0$	2, 3
18	$2x \sin x - \cos x = 0$	0.4, 1	$e^x + (1 + e^{2x})^{1/2} - 2 = 0$	-1, 0
19	$\ln x - x + 1,8 = 0$	2, 3	$x \operatorname{tg} x - 1/3 = 0$	0.2, 1
20	$0,4 + \operatorname{arctg}(x^{1/2}) - x = 0$	1, 2	$0,6 3^x - 2,3x - 3 = 0$	2, 3

### 5.8.3. Указания и примеры выполнения задания

1. Загрузите электронную таблицу.

2. Проанализируйте функцию, определите область определения функции. Отделите корни уравнения. Для этой цели протабулируйте функцию на значительном отрезке с большим шагом и зафиксируйте границы смены знака функции или постройте график функции и отделите корни уравнения по графику.

**Внимание:** Если функция имеет точки разрыва, то на графике они не будут видны.

3. Уточните значения корня на отрезке отделения простым табулированием:

– выберите отрезок, на котором функция меняет знак, и протабулируйте функцию на этом отрезке с меньшим шагом;

– повторяйте операцию, пока не будет достигнута заданная точность.

Пример уточнения значения корня методом простого табулирования приведен на листинге 5.8.1.

Листинг 5.8.1. Поиск и уточнение значения корня табулированием функции

	A	B	C	D	E	F	G	H
1	Найти корень функции $y=2^x+2x-5$ с точностью 0,001							
2								
3	1-й шаг		2-й шаг		3-й шаг		4-й шаг	
4	X	f(x)	X	f(x)	X	f(x)	X	f(x)
5	-5	-14,9688	1	-1	1,2	-0,3026	1,28	-0,01161
6	-4	-12,9375	1,1	-0,65645	1,21	-0,26662	1,281	-0,00793
7	-3	-10,875	1,2	-0,3026	1,22	-0,23053	1,282	-0,00424
8	-2	-8,75	1,3	0,062289	1,23	-0,19433	1,283	-0,00056
9	-1	-6,5	1,4	0,439016	1,24	-0,15801	1,284	0,003132
10	0	-4			1,25	-0,12159	1,285	0,006821
11	1	-1			1,26	-0,08504	1,286	0,01051
12	2	3			1,27	-0,04838		
13	3	9			1,28	-0,01161		
14	4	19			1,29	0,025281		
15					1,3	0,062289		

На первом шаге корень находится на отрезке [1;2], на втором шаге – на отрезке [1,2; 1,3], на третьем шаге - на отрезке [1,28; 1,29], на четвертом шаге - на отрезке [1,283; 1,284]. Все операции выполняются путем копирования области данных с последующей корректировкой значений аргумента.

На четвертом шаге шаг табуляции равен 0.001, точность поиска корня равна заданной – 0.001. Значение функции при  $x=0,283$  равно 0,00056 что также меньше заданной точности. Следовательно, в качестве значения корня можно принять  $x=1.283$

4. Уточните значение корня методом простых итераций (листинг 5.8.2):

Листинг 5.8.2. Пример выполнения задания

	A	B	C	D	E	F
1	Решение нелинейных уравнений $y=2^x+2x-5$					
2						
3	1. Простое табулирование		2. Метод итераций		3. Метод Ньютона	
4	Начальное значение -5		Нач. знач.	1.28	Нач. знач.	3
5	Шаг табуляции .1		Шаг таб.	нет	Шаг таб.	нет
6	Точность	.001	Точность	.001	Точность	.001
7	Формула $y=2^x+2x-5$		$y=2^x+2x-5$		$y=2^x+2x-5$	
9			Рекуррентная формула: $x=(5-2^x)/2$		$x = x - (2^x + 2x - 5) / (2^x * LN(2) + 2)$	
10	X	Функция f(x)	Функция φ(x)	Разность =ABS(x <sub>i+1</sub> - x <sub>i</sub> )	Функция x <sub>i</sub> -f(x <sub>i</sub> )/f'(x <sub>i</sub> )	F(x <sub>i</sub> )
11	1.28	-0.11610	1.285805		1,80719	2,11396
12	1.2805	-0.09768	1.280910	.0048955	1,32953	0,17228
13	1.281	-0.07926	1.285039	.004130	1,28349	0,00127
14	1.2815	-0.06084	1.281557	.0034828	1,28315	0,00000
15	1.282	-0.04241	1.284494	.002938		
16	1.2825	-0.02399	1.282017	.0024778		
17	1.283	-0.00555	1.284107	.002090		
18	1.2835	.0012882	1.282344	.0017628		
19	1.284	.0031320	1.283831	.001487		
20	1.2845	.0049761	1.282577	.0012541		
21	1.285	.0068205	1.283635	.001058		
22	Функция плохо сходится					
23						
24	Подбор параметра		Поиск решения			
25	X	f(x)	x	f(x)		
26	1,28314	-3,9 E-5	1,28315	0,00000		

- разделите переменные и запишите рекуррентную формулу  $x=(5-2^x)/2$ ;
  - введите в ячейку C11 правую часть формулы при начальном значении аргумента – 1,28;
  - введите в ячейку C12 эту же формулу, но в качестве аргумента используйте значение из ячейки C11;
  - запишите в ячейку D12 формулу для контроля точности: ABS(C11-C12);
  - скопируйте формулы из ячеек C12 и D12 в нижележащие ячейки до достижения требуемого результата.
5. Уточните значение корня методом Ньютона:
- найдите аналитическое выражение первой производной от заданной функции:  $2^x * LN(2)+2$ ;
  - введите в ячейку E11 правую часть формулы  $x_{i+1} = x_i - f(x_i)/f'(x_i)$ .

- (при начальном значении аргумента – 3);
- введите в ячейку E12 эту же формулу, но в качестве аргумента используйте значение из ячейки E11;
- запишите в ячейку F12 формулу для вычисления значения функции при текущем значении аргумента;
- скопируйте формулы из ячеек E12 и F12 в нижележащие ячейки до достижения требуемого результата.

6. Решите данную задачу с помощью функции Подбор параметра:

- введите в ячейку A26 начальное значение аргумента, например, 5;
- введите в ячейку B26 формулу  $y=2^x+2x-5$  со ссылкой на ячейку A26;
- введите команду **Сервис, Подбор параметра**. Введите в строку "Установить в ячейке" адрес функции – B26; введите в строку "Значение" – 0; введите в строку "Изменяя значение ячейки" адрес аргумента – A26 и щелкните по кнопке ОК.

7. Решите данную задачу с помощью функции Поиск решения:

- введите в ячейку C26 начальное значение аргумента, например –5;
- введите в ячейку D26 формулу  $y=2^x+2x-5$  со ссылкой на ячейку C26;
- введите команду **Сервис, Поиск решения**. Введите в строку "Установить целевую ячейку" адрес функции – D26; установите переключатель "Равной значению" и введите в строку "значению" – 0; введите в строку "Изменяя ячейки" адрес аргумента – C26 и щелкните по кнопке Выполнить. В окне диалога Результаты поиска решения щелкните по кнопке ОК.

8. Сравните результаты решения. Сделайте выводы.

Пример выполнения задания приведен на листинге 5.8.2

### Контрольные вопросы

1. В чем отличие алгебраического уравнения от трансцендентного уравнения?
2. Что называется корнем уравнения?
3. Что такое отделение корня? Для какой цели оно производится?
4. Что такое уточнение корня, какими методами оно осуществляется?
5. Поясните, в чем заключается метод простых итераций?
6. Поясните принцип использования метода касательных (метод Ньютона) для уточнения корня.
7. Что такое рекуррентная формула, как она получается?
8. Поясните порядок решения алгебраических и трансцендентных уравнений с помощью модуля Подбор параметра и Поиск решения.

## 5.9. Определение коэффициентов эмпирических формул методом наименьших квадратов

**Цель занятия:** приобрести практические навыки в обработке экспериментальных данных методом наименьших квадратов с использованием встроенных функций электронной таблицы.

**Время:** 2 часа.

**Литература:** Л1 с. 433 – 440.

### 5.9.1. Общие сведения

**Постановка задачи:** в результатов испытаний (наблюдений) получен ряд значений переменных  $x$  и  $y$ :

$x_1$	$x_2$	...	$x_k$	...	$x_n$
$y_1$	$y_2$	...	$y_k$	...	$y_n$

где  $x_i$  - значения аргумента, изменяющиеся с постоянным шагом и расположенные в порядке возрастания их значений,  $y_i$  - экспериментальные значения функции, соответствующие данным значениям аргументов.

Однако характер функциональной зависимости между  $x$  и  $y$  неизвестен. Требуется по полученным данным найти аналитическое выражение зависимости между ними:  $y=f(x_i, a_1, a_2, \dots, a_m)$ , где функция  $f$  зависит не только от значения аргумента  $x_i$ , но и от некоторых **параметров**  $a_j$ .

Формулы, полученные в результате решения задач подобного рода, называются эмпирическими.

Разности между экспериментальными значениями функции и значениями функции, полученными по эмпирической формуле

$$y_i - f(x_i, a_1, a_2, \dots, a_m) = e_i, \quad i=1,2,3,\dots,n \quad (5.9.1)$$

называются отклонениями или погрешностями. Здесь

$x_i$  - числа из первой строки таблицы,

$y_i$  - числа из второй строки данной таблицы,

$f(x_i, a_1, a_2, \dots, a_m)$  - значения функции при соответствующих значениях аргумента  $x_i$  и коэффициентах  $a_j$ .

**Коэффициенты** (Параметры)  $a_j$  эмпирической формулы  $y=f(x, a_1, a_2, \dots, a_m)$  необходимо подобрать таким образом, чтобы отклонения  $e_i$  оказались наименьшими. Наиболее распространенным критерием является критерий, лежащий в основе **метода наименьших квадратов**: параметры функции выбирают так, чтобы сумма квадратов отклонений оказалась минимальной:

$$S = \min \sum_{i=0}^n e_i^2 = \sum_{i=0}^n [f(x_i, a_1, a_2, \dots) - y_i]^2 \quad (5.9.2)$$

Минимум функции находят, приравнявая нулю частные производные по переменным параметрам  $a_j$ :

$$\frac{dS}{da_1} = 0, \quad \frac{dS}{da_2} = 0, \dots \quad (5.9.3)$$

Полученные соотношения образуют систему уравнений, для определения коэффициентов  $a_j$ , для  $j=1,2,\dots,m$ .

Электронная таблица Excel располагает встроенными средствами для определения коэффициентов эмпирических формул – это функции ЛИНЕЙН, ЛГРФПРИБЛ, ТЕНДЕНЦИЯ и РОСТ. Все эти функции возвращают множество точек аппроксимирующей кривой. Функции ЛИНЕЙН и ЛГРФПРИБЛ возвращают, кроме того, и коэффициенты уравнений регрессии.

Функция ЛИНЕЙН использует модель многомерной линейной регрессии

$y=(x_{1,i}, x_{2,i}, \dots) = A+Bx_{1,i}+Cx_{2,i}+ \dots$ , а функция ЛГРФПРИБЛ – выражение  $y=A(B^{x_1})(C^{x_2}) \dots$

Функция ЛИНЕЙН имеет следующий формат:

ЛИНЕЙН(У - массив; X – массив; конст; статистика),

где У – массив – ссылка на массив данных Y; X – массив – ссылка на один или несколько массивов данных x; конст – логическое значение, определяющее константу сдвига; статистика – логическое значение, которое указывает, требуется ли вернуть дополнительную статистику регрессии: *стандартная ошибка коэффициентов, стандартная ошибка оценки у, число степеней свободы.*

Функция ЛГРФПРИБЛ реализует следующую модель:

$$y(x_1, x_2, \dots) = A(B^x)(C^{x^2}) \dots$$

Синтаксис функции:

ЛГРФПРИБЛ(У - массив; X – массив; конст; статистика)

Рассмотрим использование этих возможностей на примерах.

### 5.9.2. Задание

1. Найти параметры эмпирических формул для линейной, логарифмической и степенной зависимостей методом наименьших квадратов. Выбрать предпочтительный вариант. Сделать вывод о полученных результатах.
2. Построить графики функций по полученным результатам. Сделать распечатки графиков и таблицы.
3. Подобрать коэффициенты эмпирических формул, используя линию тренда.

#### Варианты заданий

Вариант	x	0,00	0,50	1,00	1,50	2,00	2,50	3,00	3,50	4,00	4,50
1	y	0,067	0,079	0,264	0,510	0,692	0,781	1,241	1,439	1,603	1,666
2	y	0,114	0,328	0,333	0,524	0,555	1,462	1,532	1,837	2,044	2,447
3	y	0,101	0,347	0,494	0,918	0,970	1,427	1,576	1,905	2,088	3,612
4	y	0,380	0,429	0,454	0,490	1,118	1,274	1,435	1,697	2,010	2,042
5	y	0,135	0,145	0,158	0,201	0,423	0,426	0,939	1,298	1,358	1,973
6	y	0,045	0,294	0,367	0,418	0,684	1,086	1,415	2,007	2,066	2,266
7	y	0,093	0,350	0,357	0,357	0,517	0,772	1,075	1,339	1,844	2,267
8	y	0,480	0,650	0,756	0,860	1,136	1,141	1,393	1,487	1,533	2,581
9	y	0,167	0,454	0,959	1,014	1,143	1,717	1,799	1,823	1,881	2,041
10	y	0,879	0,992	1,135	1,238	1,660	1,788	1,876	2,094	2,402	2,498
11	y	0,333	0,341	0,648	0,662	1,327	1,356	1,426	1,897	1,933	3,820
12	y	0,307	0,451	0,579	0,693	0,974	1,552	1,778	1,850	1,934	2,307
13	y	0,039	0,442	0,841	0,884	1,088	2,023	2,109	2,206	2,361	2,369
14	y	0,184	0,484	0,549	0,750	0,770	0,955	1,616	1,772	2,253	2,535
15	y	0,078	0,545	0,584	0,817	1,049	1,128	1,728	1,789	2,360	2,860

### 5.9.3. Указания и примеры выполнения задания

1. Линейная аппроксимация

- Введите в столбцы A и B заданные значения X и Y (листинг 5.9.1).
- Введите в ячейки F3 и G3 обозначение коэффициентов регрессии - символы "B" и "A". Присвойте ячейкам F4 и G4 имена B и A, соответственно (чтобы не использовать абсолютный адрес при копировании формул): выделите ячейку, введите команду **Вставка, Имя, Присвоить**.

- В ячейку C2 введите формулу  $B \cdot A^2 + A$  ( то есть используется простейшая формула линейной аппроксимации  $y = Vx + A$ ) и скопируйте эту формулу в соответствующие ячейки колонки C.

- Оформите таблицу регрессии в соответствии с Листингом 5.9.1.

- Выделите блок F4:G8 и введите в первую ячейку выделенного блока функцию ЛИНЕЙН(B2:B9;A2:A9,1,1) и вставьте ее во весь блок командой Ctrl+Shift+Enter.

Листинг 5.9.1. Использование функции ЛИНЕЙН

	A	B	C	D	E	F	G	H
1	X	Y	Ожидаемое					
2	250	0,445	0,36156	<b>Таблица регрессии</b>				
3	300	0,362	0,33631			B	A	Комментарий
4	350	0,302	0,31106			-0,00050	0,487797	Коэффициенты
5	400	0,256	0,28582	$S_A, S_B$		0,00006	0,033366	Стд. ошибка коэфф.
6	450	0,223	0,26057	$r^2$		0,87542	0,038741	Стд. ошибка оценки Y.
7	500	0,197	0,23532	F		77,29486	11	Степени свободы
8	550	0,176	0,21008	Сумма кв.		0,11601	0,01651	
9	600	0,158	0,18483					

## 2. Логарифмическая аппроксимация

- Функция логарифмического приближения применяется аналогично функции линейного приближения. Если в функции используется более одного вектора X, то в качестве блока аргумента x указывается область, например, A2:B12. Векторы X1 и X2 не должны совпадать.

- Введите в столбцы A, B и C заданные значения X1 и X2 и Y.

- Введите в ячейки F3 и G3 обозначение коэффициентов регрессии - символы "C", "B" и "A". Присвойте ячейкам F5, G5 и H5 имена C, B и A, соответственно.

- В ячейку C2 введите формулу  $A(B^{X1})(C^{X2})$ , то есть ограничьтесь двумя векторами X, и скопируйте эту формулу в соответствующие ячейки колонки D.

- Оформите таблицу регрессии в соответствии с листингом 5.9.2.

- Выделите блок F5:H9. Введите в первую ячейку выделенного блока функцию ЛГРФПРИБЛ(C3:C12;A3:B12,1,1) и вставьте ее в весь блок.

Листинг 5.9.2. Использование функции ЛГРФПРИБЛ

	A	B	C	D	E	F	G	H	I
1	<b>Логарифмическое приближение</b>								
2	X1	X2	Y	Ожидаемое					
3	0,678	1	1,155	1,32186	<b>Таблица регрессии</b>				
4	0,99	1,2	2,682	2,38376		C	B	A	
5	1,108	1,4	3,298	2,86939		0,73918	8,03326	0,43543	Коэффициенты
6	1,434	1,6	5,1	5,32765		0,12658	0,22310	0,13382	Стд. ошибка коэфф.

Листинг 5.9.2. Использование функции ЛГРФПРИБЛ

	A	B	C	D	E	F	G	H	I
7	1,514	1,8	5,561	5,92485	r^2	0,97899	0,09183	#Н/Д	Стд. ошибка оценки Y.
8	1,534	2	5,68	5,81467	F	163,12189	7,00000	#Н/Д	Степени свободы
9	1,567	2,2	5,875	5,86323	Сумма кв.	2,75139	0,05903	#Н/Д	
10	1,637	2,4	6,295	6,38601					
11	1,684	2,6	6,579	6,62994					
12	1,705	2,8	6,708	6,52022					

### 3. Аппроксимация степенной функцией

Функции Excel не рассчитаны на выполнение степенной регрессии, но функцию ЛИНЕЙН можно легко приспособить для вычисления коэффициентов эмпирических формул с использованием степенной регрессии  $y=A + Bx + Cx^2 + \dots$ . Для этого в выражении множественной регрессии вводят следующие замены:  $x_{1,i} = x_i$ ;  $x_{2,i} = x_i^2$ ;

$$x_{3,i} = x_i^3 \dots$$

Пример использования степенной регрессии приведен на листинге 5.9.3.

### 4. Аппроксимация с помощью линий тренда:

- составьте таблицу значений экспериментальных данных X и Y;
- постройте график функции (точечный);
- щелкните правой кнопкой мыши по линии графика – открывается контекстное меню;
- выберите в этом меню команду **Добавить линию тренда**. Открывается окно диалога Линия тренда;
- выберите в этом окне подходящую функцию, в данном примере полиномиальную, и выберите степень функции таким образом, чтобы она наиболее точно описывала экспериментальные данные. Достоверность аппроксимации оценивается коэффициентом  $R^2$  - коэффициент детерминированности является квадратом коэффициента корреляции (r). Он может принимать значения от 0 до 1. Чем больше этот коэффициент, тем ближе располагаются точки линии тренда к экспериментальным точкам на графике. Приближение считается хорошим, если  $R^2$  больше 0,9. Если  $R^2=1$ , то это означает полное совпадение прогнозируемых и фактических данных;

Листинг 5.9.3. Использование степенной регрессии

	A	B	C	D	E	F
1	<b>Степенная регрессия</b>					
2	X	x^2	Y	Ожидаемое		
3	0,678	0,460	1,155	1,1544		
4	0,99	0,980	2,682	2,6836		
5	1,108	1,228	3,298	3,2979		
6	1,434	2,056	5,1	5,0976		
7	1,514	2,292	5,561	5,5622		
8	1,534	2,353	5,68	5,6798		

### Листинг 5.9.3. Использование степенной регрессии

	A	B	C	D	E	F
9	1,567	2,455	5,875	5,8751		
10	1,637	2,680	6,295	6,2943		
11	1,684	2,836	6,579	6,5797		
12	1,705	2,907	6,708	6,7083		
16						
17	<b>Таблица регрессии</b>					
18		C	B	A	Комментарий	
19		0,70841	3,71973	-1,69324	Коэффициенты	
20	Sa, Sb	0,00460	0,01131	0,00649	Стд. ошибка коэфф	
21	r^2	1,00000	0,00128	#Н/Д	Стд. ошибка оценки Y	
22	F	9,7059 E+06	7,00000	#Н/Д	Степени свободы	
23	сумма кв.	31,58037	0,00001	#Н/Д		

– откройте закладку **Параметры** в окне диалога **Линия тренда** и установите флажки *Показывать уравнения на диаграмме* и *Поместить на диаграмму величину достоверности аппроксимации (R<sup>2</sup>)*.

#### Контрольные вопросы

1. Дайте постановку задачи для определения коэффициентов эмпирических формул методом наименьших квадратов.
2. В чем заключается суть метода наименьших квадратов?
3. Запишите уравнения линейной и квадратичной зависимостей.
4. Как построить графики функций линейной и квадратичной зависимостей?
5. Какие функции Excel могут использоваться для определения коэффициентов эмпирических формул.
6. Запишите уравнения регрессии для линейной и логарифмической функций.
7. Запишите уравнение квадратичной (степенной) регрессии и поясните, как ее можно реализовать с помощью функции ЛИНЕЙН.
8. Что такое линия тренда?
9. Как определить коэффициенты эмпирических формул с помощью линии тренда?

## 5.10. Автоматизация операций в Excel

**Цель занятия:** приобрести навыки в разработке макросов.

**Время:** 2 часа.

**Литература:** Л1 с. 451 – 454.

### 5.10.1. Общие сведения

Для автоматизации выполнения часто повторяющихся операций, например форматирования выделенных ячеек, оформления шапок таблиц и т. п. в Excel можно создавать макрокоманды – макросы.

Удобнее всего макросы создавать путем записи. Правда, при этом в макрос записываются все действия пользователя, в том числе и ошибочные. Однако, если алгоритм работы продуман, то проблем не возникает.

## 5.10.2. Задание

1. Создайте макрос для форматирования выделенной области.
2. Создайте макрос для построения графика функции.
3. Создайте макрос для вывода таблицы на печать
4. Назначьте макросы по п. 1 и 2 кнопкам панели инструментов.
5. Установите на рабочий лист кнопку и назначьте ей макрос вывода таблицы на печать.

## 5.10.3. Порядок создания макроса

### Создание макроса

1. Установите среднюю степень защиты от макровирусов:

- введите команду **Сервис, Макрос, Безопасность** и установить переключатель для средней степени защиты.

**Предупреждение.** Если макрос не запускается, проверьте установленную степень защиты (среднюю).

2. Для примера создайте макрос для формирования шапки таблицы.

Разработайте сценарий (алгоритма) работы:

- определить число столбцов в таблице;
- выделить область таблицы для шапки;
- отформатировать выделенную область:
  - выбрать команду **Формат, Ячейка**;
  - выбрать закладку **Выравнивание**;
  - установить выравнивание по горизонтали – по центру, по вертикали – по верхнему краю, установить флажок **“Переносить по словам”**;
  - выделить закладку **“Границы”**, выбрать толщину линии для внутренней границы, щелкнуть по кнопке **“Внутренние границы”**, выбрать толщину линии для внешней границы, щелкнуть по кнопке **“Внешние границы”**, щелкнуть по кнопке **ОК**;
  - установить с помощью панели инструментов **Форматирование** тип шрифта **Time New Roman Cyr**, размер шрифта – 12, начертание – курсив.

3. Запишите макрос в соответствии с разработанным сценарием (только процедуру форматирования):

– выделите область для форматирования ячеек;

– введите команду **Сервис, Макрос, Начать Запись**. Открывается окно диалога (рис. 5.10.1):

– введите имя макроса – **ShapkaTablilyz**, щелкните по кнопке **ОК**. Открывается панель диалога управления записью макроса с одной кнопкой – **Остановка**;

– выполните форматирование выделенной области согласно разработанному сценарию;

– завершите работу по записи макроса щелчком мыши по кнопке **Остановка записи макроса** или введите команду **Сервис, Макрос, Остановить запись**;

– проверьте работу макроса: введите команду **Сервис, Макрос, Макросы** выберите в списке макрос **ShapkaTablilyz** и дайте команду **Выполнить**.



Рис. 5.10.1. Запись макроса

### Назначение макроса кнопке панели инструментов

- введите команду *Сервис, Настройка*, выберите закладку *Команды* окна диалога *Настройка*, в списке *Категории* выберите команду *Макросы*;
- выберите в списке *Команды* кнопку “Настраиваемая кнопка” и перетащите ее на панель инструментов;
- щелкните по кнопке *Изменить выделенный объект* в окне диалога и выберите в открывшемся списке команду *Назначить макрос*, выберите в списке макросов макрос *ShapkaTablizu*.

Для удаления кнопки введите команду *Сервис, Настройка*, откройте закладку *Команды*, выберите категорию *Макросы* и перетащите кнопку с панели инструментов в окно диалога.

### Установка кнопки на рабочий лист

- введите команду *Вид, Панели инструментов*, установите флажок у панели “Элементы управления”;
- выделите на панели элементов управления мышью объект *CommandButton*. Переместите указатель мыши на лист и нарисуйте кнопку протягивая мышь при нажатой левой клавише.
- вызовите контекстное меню кнопки *Command Button* щелчком правой кнопки мыши по объекту. Выберите в контекстном меню команду *Свойства* – откроется окно диалога *Свойства (Properties)* кнопки. Измените значение свойства *Caption*, предлагаемое по умолчанию, на “*Форматирование шапки*”. Для переноса текста установите значение свойства *WordWrap* равным *True*;
- вызовите контекстное меню кнопки, выберите команду “*Исходный текст*” – открывается окно программы *Visual Basic for Application* с шаблоном обработчика события кнопки *Command Button*. Запишите в обработчике события имя макроса, который будет запускаться при щелчке мышью по этой кнопке, например, *ShapkaTablizu*. Процедура обработчика события будет выглядеть так:

```
Private Sub CommandButton1_Click()  
    ShapkaTablizu  
End Sub
```

- сохраните программу командой *Файл, Сохранить* и вернитесь в программу *Excel* командой *Close and Return to Microsoft Excel*;
- для редактирования текста программы кнопки щелкните мышью по кнопке *Режим конструктора* панели элементов управления;
- для перехода в режим использования кнопки для управления щелкните по кнопке *Выход из режима конструктора* (это имя получает кнопка *Режим конструктора* после ее активизации) на панели элементов управления.

### Контрольные вопросы

1. Для чего предназначены макросы?
2. Как записать макрос?
3. Как назначить макрос кнопке панели инструментов?
4. Как назначить макросу комбинацию клавиш?
5. Как создать кнопку на рабочем листе?
6. Как назначить кнопке макрос?
7. Как изменить надпись или рисунок на кнопке панели инструментов?

## 5.11. Решение дифференциальных уравнений первого порядка с начальными условиями

Цель занятия: приобрести практические навыки в решении дифференциальных уравнений первого порядка с начальными условиями средствами электронной таблицы.

Время: 2 часа.

Литература: Л1 с. 447 – 451.

### 5.11.1. Постановка задачи

Пусть некоторый процесс описывается дифференциальным уравнением первой степени  $y' = y(x)$ .

Известно значение процесса в некоторый момент времени и требуется оценить значение этого процесса в произвольный момент времени. Для приближенного решения этой задачи необходимо проинтегрировать данное уравнение от одного конца интервала с известными граничными значениями до другого конца интервала, на котором они неизвестны. Такие задачи получили название задачи Коши.

Задачи такого типа возникают обычно для уравнений с производными по времени. Для решения задач Коши могут использоваться разные методы: метод рядов Тейлора, метод Эйлера, модифицированный метод Эйлера и метод Рунге-Кутты. Эти методы отличаются сложностью используемых выражений, скоростью сходимости и точностью получаемых результатов.

Рассмотрим два метода: метод Эйлера и метод Рунге-Кутты.

#### Метод Эйлера

В методе Эйлера значение функции на следующем шаге вычисляется в соответствии со следующим выражением:

$$u(x+h) = u(x) + h \cdot u'(x) \quad (5.11.1)$$

Рассмотрим для примера следующее дифференциальное уравнение:

$$(1+x^2)^{1/2} \frac{du(x)}{dx} + u(x) = x \quad \text{при } x > 0 \quad (5.11.2)$$

с начальным условием  $u(0) = 0$

Требуется найти значение выражения (5.11.2) при  $x = 0,2$

**Решение:**

Приведем выражение (5.11.2) к виду выражения (5.11.1):

$$\begin{aligned} \frac{du(x)}{dx} &= (x - u(x)) / \sqrt{1+x^2} \\ du(x) &= dx \cdot (x - u(x)) / \sqrt{1+x^2}; \\ u(x+h) - u(x) &= dx \cdot (x - u(x)) / \sqrt{1+x^2} \\ u(x+h) &= u(x) + dx \cdot (x - u(x)) / \sqrt{1+x^2} \end{aligned} \quad (5.11.3)$$

Теперь для получения результата достаточно сгенерировать ряд значений аргумента  $x$  с шагом, например, 0,01 и протабулировать правую часть выражения (5.11.3) на отрезке от 0 до 0,2.

Для примера (5.11.2) известно аналитическое выражение для вычисления значения функции в произвольной точке:

$$u(x) = 0,5(x - \ln(x + \sqrt{1+x^2})) / (x + \sqrt{1+x^2}), \quad (5.11.4)$$

поэтому можно вычислить относительную погрешность вычисления по формуле  $\Delta = (u(x)_a - u(x)) / u(x)_a$ , где  $u(x)_a$  – значение функции, вычисленное по аналитической формуле.

### Метод Рунге-Кутты

Одним из наиболее распространенных методов решения дифференциального уравнения вида  $y' = f(x, y)$  на заданном отрезке  $[X_{нач}, X_{кон}]$  с начальными условиями является метод Рунге-Кутты. При решении данного уравнения точное значение  $y$  заменяют его приближенным значением:

$$Y_{i+1} = Y_i + 1/6 (K_1 + 2K_2 + 2K_3 + K_4), \quad (5.11.5)$$

где значения коэффициентов на  $i$ -том шаге вычисляются по формулам:

$$\begin{aligned} K_1 &= h * f(x_i, y_i) \\ K_2 &= h * f(x_i + 0.5h, y_i + 0.5K_1) \\ K_3 &= h * f(x_i + 0.5h, y_i + 0.5K_2) \\ K_4 &= h * f(x_i + h, y_i + K_3) \end{aligned} \quad (5.11.6)$$

Здесь  $x_{i+1} = x_i + h$ .

**Предупреждение:** в выражении 5.11.5 нельзя вынести  $h$  за скобку, что напрашивается при рассмотрении выражений 5.11.6, так как коэффициенты с меньшим индексом используются при вычислении коэффициентов с большим индексом. Игнорирование данного факта приведет к неправильным результатам.

Для проверки точности необходимо сделать второй проход с шагом  $h/2$ . Если разница между  $Y_k$  на  $k+1$  и  $k$  проходах будет меньше требуемой точности, то процесс вычисления прекращается. Значение точности выбирают в интервале от 0.001 до 0.00001. Итерационный процесс обычно быстро сходится.

Значение шага при переходе к следующей точке можно изменять. Правильность выбора шага проверяется по формуле

$$T = |(K_2 - K_3) / (K_1 - K_2)|, \quad (5.11.7)$$

величина  $T$  не должна превышать нескольких сотых.

Грубую оценку погрешности метода проводят с помощью двойного прохода

$$Y_k - Y(x_k) = |Y_{k+1} - Y_k| / 15, \quad (5.11.8)$$

где  $Y(x_k)$  – значение точного решения уравнения.

Для решения задачи необходимо вычислить значения коэффициентов при начальных условиях, выбрать начальный шаг и вычислить значения  $X_i$  и  $Y_i$ . Затем увеличить значение  $x$  и повторить процедуру вычисления.

### 5.11.2. Задание

1. Решить дифференциальное уравнение первого порядка согласно варианту задания методом Эйлера и методом Рунге-Кутты.
2. Сравнить полученные результаты, сделать выводы.

#### Варианты заданий

Решить дифференциальное уравнение методами Эйлера и Рунге-Кутты с шагом  $h=0.1$  на отрезке  $[a, b]$  при заданных начальных условиях:

1	$Y=x+\cos(y/5^{1/2})$	$y_0(1,8)=2,6$	$x \in [1,8;2,8]$
2	$Y=x+\sin(y/3)$	$y_0(1,6)=4,6$	$x \in [1,5;2,6]$
3	$Y=x+\cos(y/10^{1/2})$	$y_0(0,6)=0,8$	$x \in [0,6;1,6]$
4	$Y=x+\sin(y/7^{1/2})$	$y_0(0,5)=0,6$	$x \in [0,5;1,5]$
5	$Y=x+\cos(y/3,14)$	$y_0(1,7)=5,3$	$x \in [1,7;2,7]$
6	$Y=x+\sin(y/e)$	$y_0(1,4)=2,2$	$x \in [1,4;2,4]$
7	$Y=x+\cos(y/2^{1/2})$	$y_0(1,4)=2,5$	$x \in [1,4;2,4]$
8	$Y=x+\cos(y/3^{1/2})$	$y_0(0,8)=1,4$	$x \in [0,8;1,8]$
9	$Y=x+y^2$	$y_0(0)=0,5$	$x \in [0,1]$
10	$Y=0,2x+y^2$	$y_0(0,1)=0,1$	$x \in [0,1;1,2]$
11	$Y=0,1x+2xy$	$y_0(0)=0,8$	$x \in [0,1]$
12	$Y=2x^2+0,2xy$	$y_0(1,1)=2,6$	$x \in [1,1;2,1]$
13	$Y=xy+0,2y^2$	$y_0(1,8)=1,4$	$x \in [1,8;2,8]$
14	$Y=3x+0,8\cos(y/5^{1/2})$	$y_0(1,4)=2,6$	$x \in [1,4;2,4]$
15	$Y=0,3xy+y^2$	$y_0(0)=0,6$	$x \in [0,1]$

### 5.11.3. Указания и пример выполнения задания

1. Решите дифференциальное уравнение (5.11.2) методом Эйлера. Для этого выполните следующее:

- сгенерируйте в столбце А значения аргумента (листинг 5.11.1) от 0 до 0,2 с шагом 0,01;
- запишите в ячейку В6 правую часть выражения (5.11.3), в ячейку С6 – правую часть выражения (5.11.4), а в ячейку D6 – разницу между значением интеграла по формуле Эйлера и аналитическим выражением;
- скопируйте формулы из ячеек В6, С6 и D6 в область В7: D26.

Результат решения приведен на листинге 5.11.1.

При уменьшении шага ошибка будет уменьшаться. Так, при шаге 0.005 она составит около 0.02172, а при шаге 0.001 - 0.00433. То есть для достижения высокой точности методом Эйлера требуется выбирать очень малое значение шага, что увеличивает время вычисления.

2. Решите дифференциальное уравнение (5.11.3) методом Рунге – Кутты.

- Запишите исходные данные согласно заданию:  $h$  – в ячейку В4,  $X_l$  – в ячейку D4,  $Y(0)$  – в ячейку F4

- Заполните строку 5 шапки таблицы согласно листингу 5.11.2;

	A	B	C	D
1	Решение дифференциального уравнения			
2	методом Эйлера			
3	$(1+x^2)^{(1/2)}du(x)/dx+u(x)=x, X>0, u(0)=0$			
4		Метод Эйлера	Аналитика	
5	x	y	y	Погрешность
6	0	0,0000	0,0000	#ДЕЛ/0!
7	0,01	0,0000	0,0000	1
8	0,02	0,0001	0,0002	0,49664
...				
24	0,18	0,0144	0,0152	0,04918
25	0,19	0,0160	0,0168	0,04624
26	0,2	0,0177	0,0186	0,04359

Листинг 5.11.2. Пример решения дифференциального уравнения								
	A	B	C	D	E	F	G	I
1	Решение дифференциального уравнения методом Рунге-Кутты							
2	$((1+x^2)^{1/2})du(x)/dx+u(x)=x, X>0, u(0)=0$							
3	Исходные данные:							
4	h=	0,01	Xn=	0	Y(0)=	0		Контроль выбора шага
5	Xn=	Yn=	K1=	K2=	K3=	K4=	Yn+1=	
6	0	0	0				0	
7	0,01	0,0000	0,0000	0,0000	0,0000	0,0001	0,0000	0,0050
8	0,02	0,0002	0,0001	0,0001	0,0001	0,0002	0,0002	0,0050
...								
23	0,17	0,0136	0,0015	0,0015	0,0015	0,0015	0,0136	0,0049
24	0,18	0,0152	0,0015	0,0016	0,0016	0,0016	0,0152	0,0049
25	0,19	0,0168	0,0016	0,0017	0,0017	0,0017	0,0168	0,0049
26	0,2	0,0186	0,0017	0,0017	0,0017	0,0018	0,0186	0,0049

- Заполните строку 6:
    - в ячейку A6 внесите начальное значение аргумента  $x = D4$ ;
    - в ячейку B6 – начальное значение  $y = F4$ ;
    - в ячейку G6 – начальное значение  $y = F4$ ;
    - в ячейку C6 – формулу для вычисления коэффициента  $K1=B4*(A6-G6)/КОРЕНЬ(1+A6^2)$ ;
  - Заполните строку 7. Формулы этой строки будут использоваться для копирования:
    - в ячейку A7 – формулу для вычисления значения аргумента  $=A6+\$B\$4$ ;
    - в ячейку B7 – значение функции на текущем шаге  $Y_{n+1} = G7$ ;
    - в ячейку C7 – формулу для вычисления значения  $K1 = \$B\$4*(A6-G6)/КОРЕНЬ(1+A6^2)$ ;
    - в ячейку D7 – формулу для вычисления значения  $K2 = \$B\$4*((A6+\$B\$4/2)-(G6+C7/2))/КОРЕНЬ(1+(A6+\$B\$4/2)^2)$ ;
    - в ячейку E7 – формулу для вычисления значения  $K3 = \$B\$4*((A6+\$B\$4/2)-(G6+D7/2))/КОРЕНЬ(1+(A6+(\$B\$4/2))^2)$ ;
    - в ячейку F7 – формулу для вычисления значения  $K4 = \$B\$4*((A6+\$B\$4)-(B6+E7))/КОРЕНЬ(1+(A6+\$B\$4/2)^2)$ ;
    - в ячейку G7 – формулу для вычисления значения функции на текущем шаге  $=G6+(C7+2*D7+2*E7+F7)/6$ ;
    - в ячейку H7 – формулу для контроля выбора шага  $=(E7-D7)/(C7-D7)$ .
  - Скопируйте блок A7:H7 в блок A8:H26.
- Из сравнения результатов, полученных методом Рунге-Кутты и методом Эйлера, видно, что при том же шаге метод Рунге-Кутты дает более точный результат.

### Контрольные вопросы

1. Что такое дифференциальное уравнение? Приведите примеры задач, приводящих к решению дифференциального уравнения.
2. Как решается дифференциальное уравнение методом Эйлера?
3. Опишите порядок решения дифференциального уравнения методом Рунге – Кутты.

## 6. ИНТЕРНЕТ

### 6.1 Поиск информации в ИНТЕРНЕТ

**Цель работы:** Познакомиться с организацией работы в ИНТЕРНЕТ и интерфейсом браузера Internet Explorer. Овладеть навыками работы с поисковыми системами: <http://www.rambler.ru>, <http://www.yandex.ru>, <http://www.mail.ru>, <http://www.tut.by>, <http://www.google.ru>, <http://www.google.by>. Освоить приемы поиска информации в ИНТЕРНЕТ.

**Время:** 2 часа.

**Литература:** Л1 с. 418- 428.

#### 6.1.1. Задание

1. Проанализировать возможности поиска информации в поисковых системах:

- <http://www.rambler.ru>;

- <http://www.yandex.ru>;

- <http://www.mail.ru>;

- <http://www.tut.by>;

- <http://www.google.ru>;

- <http://www.google.by>.

Для поиска информации задать ключевое слово, фразу в соответствии с заданным вариантом и отобразить статистику поиска (количество найденных ссылок).

2. Указать какие возможности сортировки отображения информации характерны для каждой поисковой системы (по дате, .....).

3. Осуществите уточнение данных путем поиска нужного слова "в найденном" после выполнения 1 –го пункта. Так же укажите статистику для каждой поисковой системы.

4. Введите преднамеренную ошибку в слове поиска и запишите, какое сообщение при этом выдает поисковая система.

5. Составьте сложный запрос, используя знак AND между ключевыми словами, и укажите, какие поисковые системы и как на это реагируют.

6. Введите свою фамилию и укажите, были ли найдены ссылки в поисковых системах на вашу фамилию. Если были найдены ссылки, то конкретизируйте поиск, указав последовательно имя и отчество. Результаты сведите в таблицу.

#### 6.1.2. Указания к выполнению задания

Поиск нужной информации можно проводить с помощью Web-страницы поисковой системы. Для осуществления успешного поиска важно правильно сформулировать запрос путем задания фразы, одного или несколько ключевых слов, которые лучше всего описывают предмет поиска.

**Поиск по одному слову.** В поле запроса вводится одно или несколько слов, которые могут характеризовать содержание документа. Например, ввести слово **компьютер**, после чего запускается процесс поиска.

**Поиск в найденном.** В системах Рамблер, Апорт и Яндекс после выполнения поиска можно задать другое ключевое слово (или слова) и включить флажок **«Искать в найденном»**. Поиск по новым ключевым производится только среди ранее найденных документов.

**Логическая операция AND.** Связывает два элемента запроса. Будут найдены те страницы, на которых есть оба элемента.

**Логическая операция OR.** Связывает два элемента запроса. Ведется поиск страниц, на которых есть хотя бы один из этих элементов.

### 6.1.3 Варианты заданий

Варианты задания 6.1.1 представлены в нижеприведенных таблицах.

Таблица 6.1.1. Индивидуальные варианты задания

Пункты	Варианты задания		
	1	2	3
1	«Беларусь» «Вузы Республики Беларусь»	«Информатика» «Языки программирования»	«Государство» «Республика Беларусь»
3	«Брестский государственный технический университет»	«Basic»	«Брест»
4	Преднамеренная ошибка «сротельный»	Преднамеренная ошибка «алфайт»	Преднамеренная ошибка «населние»
5	«Факультет» «Строительный» «Сессия»	«Excel» «VBA»	«Брест» «Брестская крепость»

Продолжение табл. 6.1.1

Пункты	Варианты задания		
	4	5	6
1	«Информатика» «Операционные системы»	«Информатика» «Языки программирования»	«Информатика» «Табличный процессор»
3	«WINDOWS XP»	«Паскаль»	«Excel»
4	Преднамеренная ошибка «Загрзка »	Преднамеренная ошибка «функция»	Преднамеренная ошибка «ячека»
5	«Принтер», «Лазерный», «Драйвер»	«Алгоритмы» «Линейные»	«Фильтр», «Расширенный фильтр»

Продолжение табл. 6.1.1

Пункты	Варианты задания		
	7	8	9
1	«Информатика» «Текстовые редакторы»	«Информатика» «Операционная система DOS»	«Информатика» «Базы данных»
3	«Microsoft Word»	«Версия 6.22»	«Access»
4	Преднамеренная ошибка «Форматрование»	Преднамеренная ошибка «Инсталция»	Преднамеренная ошибка «Запас»
5	«Форматирование» «Шрифт» «Times New Roman»	«Autoexec.bat» «Config.sys»	«Access» «Связь» «Один к одному»

Продолжение табл. 6.1.1

Пункты	Варианты задания		
	10	11	12
1	«Информатика» «Microsoft Excel»	«Информатика» «Microsoft Word»	«Информатика» «Microsoft Word»
3	«Диаграммы»	«Шрифты»	«Стиль»
4	Преднамеренная ошибка «Граик»	Преднамеренная ошибка «Колотитулы»	Преднамеренная ошибка «Заголовк»
5	«Гистограмма» «Линейчатая» «Легенда»	«Список» «Маркированный»	«Форматирование» «Шрифт» «Arial»

Продолжение табл. 6.1.1

Пункты	Варианты задания		
	1	2	3
1	«Беларусь» «Города Республики Беларусь»	«Информатика» «Языки программирова- ния»	«Информатика» «Языки программиро- вания»
3	«Столица МИНСК»	«Basic»	«VISUAL BASIC»
4	Преднамеренная ошибка «Раоны»	Преднамеренная ошибка «фукции»	Преднамеренная ошибка «аниация»
5	«Областные центры» «Беларусь», «Населние»	«Basic», «Операторы» «Графика»	«Формы», «Объекты» «Кнопки»

#### 6.1.4. Примеры выполнения заданий

Примеры выполнения заданий приведены в таблице 6.1.2.

Таблица 6.1.2. Результаты выполнения задания 6.1

№	www.rambler.ru	www.yandex.ru	www.mail.ru	www.tut.by
1	Слово «КНИГИ»			
	507985 сайтов	36614сайтов	2120000 документов	632 сайта( РБ)* 34325сайтов (СНГ)
	Фраза «КОМПЬЮТЕРНАЯ ЛИТЕРАТУРА »			
	26330 сайтов	1015сайтов	117000 документов	154 сайта (РБ) 34325 сайтов (СНГ)
2	По дате, по релевантности	По дате, по релевантности	Не указано	Не указано
3	1082 сайта	787 сайтов	4370 документов	14 сайтов (РБ) 594 сайта (СНГ)
4	Преднамеренная ошибка в слове поиска, задав «ПРОРАММА»			
	2428 сайтов с ошибочным словом	1516 сайтов	256 документов	22 сайта (РБ) 1635 сайтов (СНГ)
	Преднамеренные 2 ошибки в слове поиска, задав «ПРРАММА»			
	14 сайтов	15 сайтов	«не найдено ни одного документа»	«искомая комбинация нигде не встречается»

№	www.rambler.ru	www.yandex.ru	www.mail.ru	www.tut.by
5	Запрос «(Алгоритм) AND (Программа) AND (Линейная структура)»			
	5743 сайта	Ничего не найдено	557 документов	6 сайтов (РБ) 113сайтов (СНГ)
6	Запрос «ИВАНОВ»			
	206682 сайта	15338 сайтов	568000 докумен- тов	1073сайта (РБ) 9657сайтов (СНГ)
	Запрос «ИВАНОВ ИВАН»			
	71237 сайтов	8185 сайтов	369000 докумен- тов	1079 сайтов (РБ) 8185 сайтов (СНГ)
	Запрос «ИВАНОВ ИВАН ИВАНОВИЧ»			
30393 сайта	1673 сайта	362000 докумен- тов	303 сайта (РБ) 2194 сайта (СНГ)	

\* РБ – Республика Беларусь, СНГ – Союз Независимых Государств

### Контрольные вопросы

1. Перечислите наиболее распространенные русскоязычные поисковые системы?
2. Что такое поиск «в найденном»?
3. Какие возможности сортировки отображения информации существуют в поисковых системах?
4. Какие существуют способы поиска информации в поисковых системах?
5. Какие логические операции используются при поиске информации?
6. Какое сообщение может выдавать поисковая система, если информация не найдена?
7. Допустимы ли с вашей точки зрения ошибки в информации, содержащейся в ИНТЕРНЕТ, и чем вы это объясняете?

## 6.2. Создание Web - странички

**Цель работы:** Познакомиться с основами языка HTML. Изучить структуру HTML – файла и основные теги. Овладеть навыками создания собственных Web – страничек.

**Время:** 2 часа.

**Литература:** Л1 с. 418- 428.

### 6.2.1. Задание

Создать Web – страничку по указанной тематике. Дизайн Web – странички разрабатывается самостоятельно. Необходимо чтобы при создании Web – странички были обязательно использованы теги шрифтов, цветовой палитры, выравнивания текста и таблиц. Другие теги могут быть использованы по мере необходимости. Работоспособность Web – странички необходимо продемонстрировать в Internet Explorer.

### 6.2.2. Указания к выполнению задания

HTML (Hyper Text Mark-up Language) является общепризнанным языком для создания Web-страниц. HTML-файл представляет собой текстовый файл, в котором записаны команды языка HTML. Команды HTML называются *тегами (tag)*. Тэги заключаются в уг-

ловые скобки. Для создания Web-документа можно воспользоваться блокнотом. Структура HTML-документа:

```
<HTML> начало Web-документа
<HEAD> заголовочная часть документа
<TITLE> заголовок окна броузера
</HEAD>
< BODY bgcolor="white">
    Тело документа, bgcolor=white определяет цвет фона (бельи)
.....
</BODY>
</HTML>
```

Для того чтобы вставить изображение из файла, необходимо использовать тэг <IMG SRC="filename.gif"> - вставить изображение из файла, например, filename.gif. Для автономного просмотра и редактирования Web – странички необходимо загрузить ее в браузер, например, Internet Explorer. В таблицах 6.2.1–6.2.5 приведены основные справочные данные, необходимые для выполнения данного задания.

Таблица 6.2.1. Базисные элементы HTML

БАЗИСНЫЕ ЭЛЕМЕНТЫ		
Тип документа	<HTML></HTML>	(начало и конец файла)
Имя документа	<TITLE></TITLE>	(должно быть в заголовке)
Заголовок	<HEAD></HEAD>	(описание документа, например, его имя)
Тело	<BODY></BODY>	(содержимое страницы)

Таблица 6.2.2. Тэги HTML оформления внешнего вида и работы с текстом

Наименование	Обозначение	Примечание
Жирный	<B></B>	
Курсив	<I></I>	
Подчеркнутый	<U></U>	(часто не поддерживается)
Верхний индекс	<SUB></SUB>	
Нижний индекс	<SUP></SUP>	
Форматированный	<PRE></PRE>	(сохранить формат текста как есть)
Ширина	<PRE WIDTH=?></PRE>	(в символах)
Мигающий	<BLINK></BLINK>	(наиболее осмеянный элемент)
Размер шрифта	<FONT SIZE=?></FONT>	(от 1 до 7)
Цвет шрифта	<FONT COLOR="#\$\$\$\$\$"></FONT>	
Ширина колонки	<MULTICOL WIDTH=?></MULTICOL>	
Выравнивание	<SPACER ALIGN=left right center>	

Таблица 6.2.3. Тэги HTML оформления внешнего вида Web – странички, указание цвета

Наименование	Обозначение	Примечание
Цвет фона	<BODY BGCOLOR="#\$\$\$\$\$\$"> [*]	(порядок: красный/ зеленый/ синий)
Цвет текста	<BODY TEXT="#\$\$\$\$\$\$"> [*]	
Цвет ссылки	<BODY LINK="#\$\$\$\$\$\$"> [*]	
Пройденная ссылка	<BODY VLINK="#\$\$\$\$\$\$"> [*]	
Активная ссылка	<BODY ALINK="#\$\$\$\$\$\$"> [*]	

Таблица 6.2.4. Тэги HTML создания и оформления таблиц

Наименование	Обозначение	Примечание
Определить таблицу	<TABLE></TABLE> [*]	
Окантовка таблицы	<table border=?></TABLE>	
Расстояние между ячейками	<TABLE CELSPACING=?>	
Желаемая ширина	<TABLE WIDTH=?>	(в точках)
Ширина в процентах	<TABLE WIDTH="%">	(проценты от ширины страницы)
Выравнивание	<TR ALIGN=LEFT RIGHT CENTER MIDDLE BOTTOM>	
Ячейка таблицы	<TD></TD>	(должна быть внутри строки)
Выравнивание	<TD ALIGN=LEFT RIGHT CENTER MIDDLE BOTTOM>	
Желаемая ширина	<TD WIDTH=?>	(в точках)
Ширина в процентах	<TD WIDTH="%">	(проценты от ширины страницы)
Цвет ячейки	<TD BGCOLOR="#\$\$\$\$\$\$">	
Заголовок таблицы	<TH></TH>	(как данные, но жирный шрифт и центровка)
Выравнивание	<TH ALIGN=LEFT RIGHT CENTER MIDDLE BOTTOM>	
Заглавие таблицы	<CAPTION></CAPTION>	
Выравнивание	<CAPTION ALIGN=TOP BOTTOM>	(сверху/снизу таблицы)

Таблица 6.2.5. Цветовая палитра, используемая в HTML

Название цвета на русском языке	Значение как название цвета
черный	<i>Black</i>
зеленый	<i>Green</i>
серебряный	<i>Silver</i>
лимонный	<i>Lime</i>
серый	<i>Gray</i>
оливковый	<i>Olive</i>
белый	<i>White</i>
желтый	<i>Yellow</i>
каштановый	<i>Maroon</i>
красный	<i>Red</i>
синий	<i>Blue</i>
пурпурный	<i>Purple</i>
бирюзовый	<i>Teal</i>
фуксиновый	<i>Fuchsia</i>
голубой	<i>Aqua</i>

### 6.2.3. Варианты заданий

Варианты задания 6.2.6 представлены в нижеприведенных таблицах.

Таблица 6.2.6. Индивидуальные варианты задания 6.2

Вариант	Задание
1	Создать Web – страничку «Республика Беларусь»
2	Создать Web – страничку «Город, в котором я живу»
3	Создать Web – страничку « Word. Назначение и возможности »
4	Создать Web – страничку «Microsoft Excel. Назначение и возможности»
5	Создать Web – страничку «Операционная система Microsoft Windows. Назначение и возможности»
6	Создать Web – страничку «Мои автобиографические данные»
7	Создать Web – страничку «Mathcad. Назначение и характеристики»
8	Создать Web – страничку «Microsoft Word. Работа с таблицами »
9	Создать Web – страничку «Visual Basic. Типы данных»
10	Создать Web – страничку «Microsoft Word. Форматирование в Word »
11	Создать Web – страничку «Вуз, в котором я учусь»
12	Создать Web – страничку «Факультет, на котором я учусь»
13	Создать Web – страничку «Описание пунктов меню Microsoft Word»
14	Создать Web – страничку «Описание пунктов меню Microsoft Excel»
15	Создать Web – страничку «Описание пунктов меню VISUAL BASIC»

### 6.2.4. Пример выполнения задания

В качестве примера рассмотрим HTML – код и дизайн справочной Web – странички «Назначение функциональных клавиш Far manager». Структура HTML – документа имеет вид.

```
<html>
<head>
<title>Программа управления ФАЙЛАМИ И АРХИВАМИ </title>
</head>
<center>
<br>
<FONT COLOR="red" SIZE=5> <H2>FAR MANAGER.</H2></FONT>
<font color="green"> НАЗНАЧЕНИЕ ФУНКЦИОНАЛЬНЫХ КЛАВИШ</font> </center>
</br>
<body>
<center>
```

```

<table BORDER=2 width=60%>
<tr align=center>
<td width=50% ><b><i>КЛАВИША</i></b></i></td>
<td width=50% ><b><i>НАЗНАЧЕНИЕ</i></b></i></td>
<tr align=center>
<td width=50% > <b><i>F1</i></b></i></td>
<td width=50% ><b><i>помощь</i></b></i></td>
</TR>
<tr align=center>
<td width=50% ><b><i>F2</i></b></i></td>
<td width=50% ><b><i>меню пользователя</i></b></i></td>
</TR>
<tr align=center>
<td width=50% ><b><i>F3</i></b></i></td>
<td width=50% ><b><i>просмотр файла</i></b></i></td>
</TR>
<tr align=center>
<td width=50% ><b><i>F4</i></b></i></td>
<td width=50% ><b><i>редактирование файла</i></b></i></td>
</TR>
<tr align=center>
<td width=50% ><b><i>F5</i></b></i></td>
<td width=50% ><b><i>копирование файлов и папок</i></b></i></td>
</TR>
<tr align=center>
<td width=50% ><b><i>F6</i></b></i></td>
<td width=50% ><b><i>переименование и перемещение файлов и па-
нок</i></b></i></td>
</TR>
<tr align=center>
<td width=50% ><b><i>F7</i></b></i></td>
<td width=50% ><b><i>создание новой папки</i></b></i></td>
</TR>
<tr align=center>
<td width=50% ><b><i>F8</i></b></i></td>
<td width=50% ><b><i>удаление файлов и папок</i></b></i></td>
</TR>
</TABLE>
<p align="justify">
Пример личной Web- странички <b>.</b>
</p>
</body>
</html>

```

Вид Web – странички представлен на рисунке 6.1

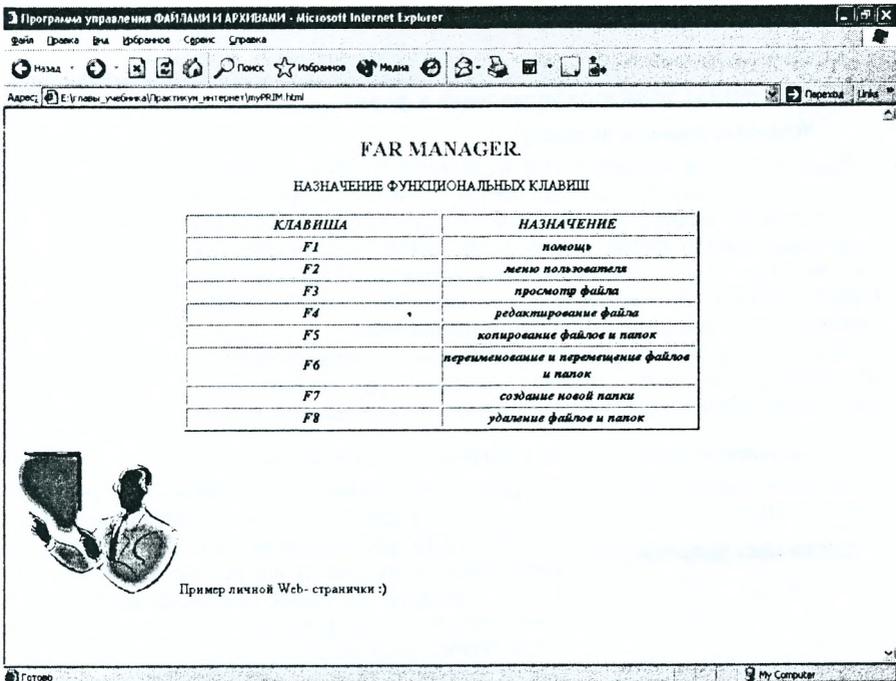


Рис. 6.1. Вид Web –странички  
“Назначение функциональных клавиш Far manager”

### Контрольные вопросы

1. Что представляет собой файл HTML?
2. Какие расширения может иметь файл, содержащий HTML – документ?
3. Какие тэги позволяют установить полужирный шрифт и курсив?
4. Как установить цвет и размер букв текста?
5. Как вставить изображение из файла в Web – страницу?
6. Какие тэги используются для формирования таблицы?
7. С помощью какого тэга можно регулировать толщину линий обрамления ячеек таблицы?

# Приложение 1

## Основные приемы работы в среде Visual Basic

### Размещение элементов управления в форме

#### Установка элемента на форму

Выделите на Панели элементов управления (Toolbox) элемент, который вы хотите поместить на форму. Выделенный элемент изменяет серый цвет на белый и становится объемным. Переместите указатель мыши на форму. Обратите внимание, что указатель мыши изменил форму и принял вид тонкого черного крестика. Установите указатель мыши в точку, где должен находиться верхний левый край объекта, нажмите левую клавишу мыши и, протаскивая мышь по диагонали к нижнему правому углу объекта, установите требуемые размеры элемента управления.

Имеется и другой, более простой способ установки объекта на форму: щелкните дважды по элементу управления, - элемент управления будет вставлен в центре формы. После этого можно переместить объект в требуемое положение и изменить его размеры.

#### Установка размеров элемента управления и его положения на форме

Выделите элемент управления на форме. Выделенный объект выделяется восемью черными точками (маркеры выделения) по периметру (рис. П1.1.). Если зацепить мышью за маркеры выделения, расположенные на сторонах объекта, то

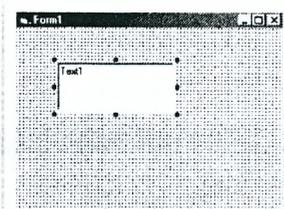


Рис. П1.1. Выделенный объект

можно изменить его ширину или высоту, а если зацепить мышью за маркеры выделения, расположенные в углах, то можно изменять одновременно оба параметра.

Для перемещения объекта в другое положение установите на него указатель мыши и нажмите левую клавишу, и, удерживая ее в нажатом состоянии, переместите объект в требуемое положение. При перемещении указателя мыши вместе с ним будет перемещаться и контур объекта.

Размеры элемента управления и его положение на форме можно установить также с помощью панели Свойств (Properties) – свойства: *Height* (высота объекта), *Width* (ширина объекта), *Top* (расстояние объекта от верхнего края формы), *Left* (расстояние объекта от левого края формы).

#### Копирование объекта

С помощью копирования можно создать объект с такими же свойствами, как и у исходного объекта. Наследуются все свойства исходного объекта, кроме свойств *Name*, *Index*, *TabIndex*, *Tag*. Для копирования объекта необходимо воспользоваться командами *Копировать* (*Copy*) и *Вставить* (*Paste*) пункта меню *Редактирование* (*Edit*), контекстного меню или соответствующими кнопками на панели инструментов. Контекстное (или всплывающее) меню появляется при щелчке правой кнопкой мыши по объекту:

выделите объект и выберите команду *Edit, Copy*. Копия объекта помещается в буфер (специальный участок оперативной памяти компьютера). Информация будет сохраняться в буфере до тех пор, пока в нее не будет помещена новая информация или не перезагрузят компьютер;

выберите команду *Edit, Paste* – копия объекта помещается в левый верхний угол формы или другого объекта – контейнера. Свойствами контейнера кроме формы обладают также *Рамка* (*Frame*) и *Окно с рисунком* (*PictureBox*). При вставке объекта программа выдаст запрос: Создать массив элементов управления или нет? Ответьте: Нет.

## Управление группой объектов

В некоторых случаях возникает необходимость изменить свойства одновременно нескольких объектов или переместить их на форме. Нужную операцию можно выполнить для каждого элемента управления по очереди, но для этого требуется много времени. Операцию можно ускорить, если выделить все эти элементы. Чтобы выделить группу элементов, выделите первый элемент, нажмите и удерживайте клавишу **Ctrl**, щелкните мышью по другим выделяемым объектам.

Другой способ выделения группы объектов – обводка. В верхнем левом углу панели элементов управления расположена кнопка с черной стрелкой – кнопка выделения. Эта кнопка по умолчанию активна. Если эта кнопка не выделена, щелкните по ней мышью.

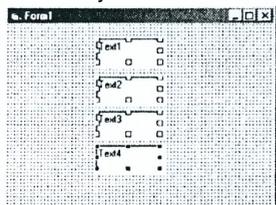


Рис. П1.2. Выделение группы объектов

Для выделения группы элементов обводкой установите указатель мыши выше и левее верхнего левого угла крайнего левого элемента группы, нажмите левую клавишу мыши и протягивайте мышшь по направлению к правому нижнему углу последнего элемента группы. Вслед за указателем мыши будет формироваться прямоугольник, выделенный пунктирной линией. Убедитесь, что пунктирный прямоугольник полностью охватывает все выделяемые фигуры и отпустите клавишу мыши. Все объекты будут выделены так, как показано на рис. П1.2.

Выделенную группу элементов можно перемещать, копировать, удалять или изменять общие свойства: цвет, шрифт, выравнивание текста и т. д. Для отмены выделения щелкните мышью по форме.

Научившись выделять группы элементов, Вы сможете управлять их размещением на форме: выравнивать по горизонтали, по вертикали или выравнивать расстояние между элементами. Эти операции выполняются с помощью команды главного меню Формат (Format). Команда Формат содержит ряд команд (опций):

*Align* – выравнивание. Позволяет выравнивать расположение элементов: по левому краю, правому краю или по центру выделенной группы; по верхнему краю, нижнему краю или по центру выделенной группы; по сетке формы.

*Size to Grid* – выравнивание по сетке. Выравнивает размеры элементов по сетке (это хорошо заметно при крупной сетке). Изменить расстояние между горизонтальными и вертикальными линиями сетки можно с помощью команды Tools, Options, General. Далее следует в группе Form Grid Settings установить нужные размеры сетки или убрать ее, сняв флажок Show Grid. По умолчанию установлен флажок автоматического выравнивания элементов управления по сетке Align Controls to Grid. При необходимости его можно снять.

*Make Same Size* – выравнивание размеров. Позволяет выравнивать размеры элементов выделенной группы по ширине, высоте или оба параметра.

*Horizontal Spacing u Vertical Spacing* – горизонтальные и вертикальные промежутки. Устанавливает равные расстояния между элементами по горизонтали и по вертикали.

*Center in Form* – центрирование в форме. Выравнивает положение элементов управления в самой форме по горизонтали и по вертикали.

*Order* – порядок. Позволяет изменить взаимное расположение элементов управления по слоям. Если элементы управления перекрывают или полностью закрывают один другой, то их можно поменять местами. Видимый сделать невидимым и наоборот.

*Lock Controls* – блокировка элементов управления. Позволяет запретить изменение размещения элементов на форме.

## Приложение 2

### Префиксы имен объектов

При выборе имен объектов и переменных рекомендуется руководствоваться правилами, получившими название "соглашение об именах". Эти соглашения делают текст программы более читабельным и создают определенные удобства при поиске объектов в списках окна программы.

Таблица П2.1

#### Префиксы объектов

Элемент управления	Префикс	Пример
Форма	frm	frmTabuljzjij
Кнопка	cmd	cmdCalc
Надпись	lbl	lblName
Текстовое поле	txt	txtShag
Поле со списком	cbo	cboMonat
Список	lst	lstVremenaGoda
Рамка	fra	fralschodDannye
Переключатель	opt	optFunction
Флажок	chk	chkOn
Графическое поле	pic	picStatica
Рисунок	img	imgBall
Полоса прокрутки	scr	scrRed
Таймер	tmr	tmrScorost
Список устройств	drv	drvDisk
Список каталогов	dir	dirDirect
Список файлов	fil	filFile
Линия	lin	linSeparator
Фигура	sha	shaCircle
Элемент данных	dat	datDataBases
Элемент OLE	ole	oleWordDoc
Дерево каталогов	twv	twvGroups
Табличный список	lvw	lvwUsers
Панель инструментов	tbr	tbrProject1
Список изображений	iml	imlGroups
Строка состояния	str	strStatus
Окно диалога	cdl	cdlOpen
Сетка	flx	flxGrid

Программа по умолчанию присваивает имена объектам. Однако это может создать определенные неудобства, когда на форме появится большое число однотипных объектов. Очевидно проще найти объект по его имени, чем запоминать индексы. При работе с формами также могут возникнуть проблемы при попытке добавить новую форму. Программа не позволит добавить в проект форму, если в нем уже существует форма с таким именем.

Примите за правило: создал объект – присвой ему уникальное имя, соответствующее назначению этого объекта.

Основные рекомендации о соглашении имен сводятся к следующему:

имена переменных могут содержать символы верхнего и нижнего регистров, но без пробелов, например, `FamilyName`;

имена констант должны содержать только прописные буквы, а вместо пробелов использовать символ подчеркивания;

имя переменной начинается с префикса, обозначающего тип переменной (см. табл. П2.1);

символ префикса пишется в нижнем регистре, а первый символ имени переменной – в верхнем регистре, например, `fXpach` – вещественная переменная одинарной точности. Префиксы имен переменных, наиболее часто используемых в программе, приведены в табл. П2.1;

имя объекта должно начинаться с префикса. Префикс пишется со строчной буквы, а первый символ имени объекта - с прописной, например, `txtTemperatura`, `frmVvodDannych`.

## Приложение 3

### Основные функции и типы данных

#### Арифметические операторы

<i>Оператор</i>	<i>Описание</i>
+	Вычисление суммы двух чисел
-	Вычисление разности двух чисел или изменение знака числового выражения
*	Вычисление произведения двух чисел
/	Вычисление частного от деления двух чисел
\	Вычисление частного от деления двух целых чисел (целочисленное деление).
^	Возведение числа в степень
Mod	Возвращает остаток при целочисленном делении двух чисел. Остаток равен нулю, если числа делятся без остатка, и единице, если числа не делятся без остатка

#### Логические операторы

<i>Оператор</i>	<i>Описание</i>	<i>Оператор</i>	<i>Описание</i>
<	меньше	<=	меньше или равно
>	больше	>=	больше или равно
=	равно	<>	не равно

### Арифметические функции

Функция	Описание
Abs(N) <sup>7</sup>	Вычисляет абсолютное значение числа N
Atn(N)	Вычисляет арктангенс числа N. <sup>8</sup>
Cos(N)	Вычисляет косинус аргумента.
Exp(N)	Возвращает число e, возведенное в указанную степень
Fix(N)	Возвращает целое число, меньшее или равное N для положительных чисел и большее или равное N для отрицательных чисел
Int(N)	Возвращает целое число, меньшее или равное N как для положительных, так и для отрицательных чисел
If(усл., V1,V2)	Условная функция, возвращает результат согласно выражению V1, если условие истинно, и согласно выражения V2, если условие ложно
Log(N)	Вычисляет натуральный логарифм аргумента
Rnd(N)	Возвращает случайное число в интервале от 0 до 1. При N<0 возвращает определенное число, зависящее от N; при N=0 – псевдослучайное число; при N>0 – новое случайное число
Round(N[, дес])	Возвращает число, округленное к заданному числу десятичных знаков
Sgn(N)	Возвращает знак числа: 1, если N>0; 0, если N=0; -1, если N<0
Sin(N)	Вычисляет синус угла
Sqr(N)	Вычисляет корень квадратный из аргумента
Tan(N)	Вычисляет тангенс угла
Val(C)	Преобразует аргумент строкового типа в число

### Логические функции

Функция	Описание
And	Вычисление логического И (умножения) для двух выражений
Eqv	Проверка логической эквивалентности двух выражений
Imp	Логическая импликация для двух выражений
Is	Сравнение двух переменных, содержащих ссылки на объекты
Like	Сравнение двух строковых выражений
Not	Операция логического отрицания над выражением
Or	Операция логического ИЛИ (сложения) для двух выражений
Xor	Операция исключающего ИЛИ для двух выражений
&	Операция слияния двух строковых выражений

<sup>7</sup> N – переменная числового типа, C – строка символов.

<sup>8</sup> В тригонометрических функциях аргумент измеряется в радианах. Для пересчета радианов в градусы надо умножить значение функции на 180/π. Для пересчета градусов в радианы надо умножить значение функции на π/180. Приближенное значение числа π равно 3,141593, а более точное значение можно вычислить по формуле  $\pi = 4 * \text{Atn}(1)$

### Функции и операторы работы с массивами

<b>Операторы и функции</b>	<b>Действие</b>
Option Base	Установка нижней границы индекса массива по умолчанию
Dim	Описание переменных и выделение для них областей памяти
Private	Описание локальных переменных и выделение для них областей памяти. Применяется на уровне модуля
Public	Описание общих переменных и выделение для них областей памяти. Применяется на уровне модуля
ReDim	Перераспределение памяти для переменных динамического массива. Применяется на уровне процедуры.
Static	Описание переменных и выделение памяти. Применяется на уровне процедуры. Переменные, описанные с помощью оператора Static, сохраняют свои значения при выходе из процедуры и используются при повторном вызове
Lbound	Определение нижней границы диапазона индексов массива
Ubound	Определение верхней границы диапазона индексов массива
Erase	Повторная инициализация элементов массивов фиксированной длины и освобождение памяти, отведенной для динамического массива
isArray	Проверка, является ли переменная массивом
Array	Создание массива типа Variant

### Функции работы с датами и временем

<b>Операторы и функции</b>	<b>Действие</b>
Date	Возвращает текущую дату в формате мм.дд.гг, где мм – месяц, дд – день, гг – год. Позволяет установить системную дату
DateAdd(интервал, число интервалов, дата)	Добавляет указанный интервал к дате или вычитает из нее. Параметр "интервал" – символьная переменная и может принимать следующие значения: Yyy – год, Q – квартал, M – месяц, Y – день в году, D – число месяца, W – день недели, Ww – неделя, H – Час, N – минута, S – секунда. Параметр "число интервалов" означает число интервалов, которые должны быть добавлены или отняты, положительные значения прибавляются, а отрицательные отнимаются. Параметр "дата" содержит допустимую дату.
DateDiff(интервал, дата1, дата2, первый день недели, первая неделя года)	Вычисляет, сколько временных интервалов проходит между двумя указанными датами. Параметр "интервал" может принимать те же значения, что и в функции DateAdd. Параметр "первый день недели" по умолчанию имеет значение "воскресенье". Параметр "первая неделя" принимает значения: 0 – установка приложения, 1 – неделя с первым января, 2 – неделя с не менее чем четырьмя днями нового года, 3 – первая полная неделя в новом году, по умолчанию принимает значение недели, на которую приходится 1 января.

Продолжение таблицы

DatePart(интервал, дата1, дата2, первый день недели, первая неделя года)	Вычисляет, к какой части указанного интервала относится дата Значения параметров аналогичны ранее описанным функциям
DateSerial(год, месяц, день)	Возвращает дату, состоящую из указанного года, месяца и дня
DateValue(C)	Возвращает дату, указанную в аргументе – строке.
Day(N)	Возвращает целое значение от 1 до 31, которое означает число месяца указанной даты.
Month	Возвращает целое число от 1 до 12, обозначающее месяц указанной даты
MonthName(N)	Возвращает строку с именем месяца по его номеру
Now	Возвращает системную дату и системное время компьютера
Weekday(дата [, первый день недели])	Возвращает целое число от 1 до 7, соответствующее дню недели указанной даты
WeekdayName(N)	Возвращает строку, указывающую определенный день недели, соответствующее числовому аргументу
Year(дата)	Возвращает целое число от 100 до 9999, соответствующее году указанной даты
Hour Month	Возвращает целое число от 0 до 23, обозначающее час указанного времени
Minute(N)	Возвращает целое число от 0 до 59, обозначающее минуту текущего часа указанного времени
Second(N)	Возвращает целое число от 0 до 59, обозначающее секунду текущей минуты указанного времени
Time	Возвращает текущее системное время
Timer	Возвращает число прошедших с полуночи секунд. Функция может служить в качестве секундомера для определения временных интервалов
TimeSerial	Возвращает значение времени при заданных часе, минуте и секунде
TimeValue	Преобразует строку, представляющую дату, в значение времени

**Функции обработки строк**

<b>Функция</b>	<b>Комментарий</b>
Asc (C )	возвращает ASCII код первого символа строки символов C
Chr(N)	возвращает символ, соответствующий коду ASCII (числа от 32 до 255)
Space(N)	генерирует строку символов из N пробелов
String(N,Cod)	генерирует строку из символов, повторяющую N раз символ, соответствующий ASCII – коду. Код может принимать числовые значения от 0 до 255. Числа больше 255 преобразуются по формуле (код Mod 256), т. е. целочисленное деление кода числа на основание 256

<b>Функция</b>	<b>Комментарий</b>
LCASE(C)	преобразует прописные буквы в строчные
UCASE(C)	преобразует строчные буквы в прописные
LTrim(C), RTrim(C), Trim(C)	удаляет левые, правые или и левые и правые пробелы из строки символов
Len(C)	возвращает длину строки
Left(C, N)	возвращает N символов с левого конца строки C
Right(C, N)	возвращает N символов с правого конца строки C
Mid(C, N1, N2)	возвращает N2 символов из строки C начиная с N1 символа
InStr ([I,] C, C1, Compare)	осуществляет поиск вхождения строки C1 в строку C. Поиск начинается с позиции i, если i отсутствует, то поиск осуществляется с начала строки. Опция Compare указывает способ сравнения: -1 – использует метод сравнения, заданный оператором Option Compare, 0 – двоичное сравнение выражений; 1 – посимвольное сравнение; 2 – только в Microsoft Access, сравнение осуществляется на основе параметров, установленных в базе данных
InStr (C, C1, I, Compare)	то же, что и InStr, но возвращает позицию первого вхождения последовательности символов C1 в C, начиная с конца строки
Str (N)	преобразует числовое выражение в строку
CInt(N), CLng(N), CSng(N), Dbt(N), CStr(N) и др.	преобразуют числовой аргумент в целое число одинарной или двойной точности, вещественное число одинарной или двойной точности или в строку символов, соответственно.

### Операторы и функции работы с файлами

<b>Оператор</b>	<b>Описание</b>
Open	Открытие файла
Close	Закрытие файла
Reset	Закрытие всех файлов, открытых с помощью оператора Open
EOF	Возвращает значение, показывающее, расположен ли указатель текущей записи в позиции, после последней записи объекта Recordset
FreeFile	Возвращает номер свободного канала, доступного для использования оператором Open
Loc	Возвращает значение, определяющее текущее положение указателя чтения/записи в открытом файле
LOF	Возвращает размер в байтах файла, открытого с помощью оператора Open
Get	Чтение данных из открытого файла прямого доступа на диске
Input	Чтение данных из файла, открытого в режиме Input или Binary
Input#	Чтение данных из открытого последовательного файла и присвоение их переменным
Input\$	Возвращает значение типа String, содержащее символы из файла, открытого в режиме Input или Binary

<b>Оператор</b>	<b>Описание</b>
Line Input#	Чтение строки из открытого последовательного файла и присвоение ее переменной типа String
Print#	Запись в файл
Put	Запись содержимого переменной в файл прямого доступа на диске
Write#	Запись не форматированных данных в файл последовательного доступа
Spс	Установка позиции вывода на экран или печать при использовании вместе с оператором Print# или методом Print
Tab	Используется совместно с оператором Print# или методом Print для задания позиции вывода на экран или печать
Width#	Задание ширины строки для файла, открытого с помощью оператора Open
FileCopy	Копирование файлов. В качестве аргументов могут указываться имя каталога или папки на диске
Dir	Получение имени файла, каталога или папки, удовлетворяющего шаблону имени файла, набору атрибутов файла или метке тома на диске
Kill	Удаление файлов с диска
Name	Изменение имени файла, каталога или папки
ChDir	Изменение текущего каталога или папки
ChDrive	Изменение текущего диска
MkDir	Создание нового каталога или папки
Rmdir	Удаление существующего каталога или папки
GetAttr	Считывание атрибутов файла
SetAttr	Задание атрибутов файла
LoadPicture	Загрузка графического изображения
SavePicture	Сохранение графического изображения

### Функции проверки значений выражений

<b>Функция</b>	<b>Описание</b>
IsArray	Возвращает значение типа Boolean, показывающее, является ли переменная массивом
IsDate	Возвращает значение типа Boolean, которое показывает, может ли значение выражения быть преобразовано в дату
IsEmpty	Возвращает значение типа Boolean, показывающее, была ли переменная инициализирована
IsMissing	Возвращает значение типа Boolean, которое показывает, был ли передан в процедуру необязательный параметр
IsNull	Возвращает значение типа Boolean, которое показывает, является ли результатом вычисления выражения значение Null
IsNumeric	Возвращает значение типа Boolean, которое показывает, является ли результат числовым значением
IsObject	Возвращает значение типа Boolean, которое показывает, является ли переменная объектной

## Средства обработки ошибок

Элемент	Описание
Err (объект)	Содержит информацию, однозначно определяющую ошибку периода выполнения
Clear (метод)	Сбрасывает до нулей или пустых строк ("" ) все значения свойств объекта Err
ErrOr (функция)	Генерирует ошибку периода выполнения
Raise (метод)	Создает ошибку, которая возникает на этапе выполнения программы
On ErrOr (оператор)	Перехватывает ошибки периода выполнения
IsErrOr (функция)	Возвращает значение типа Boolean, показывающее, представляет ли аргумент значение ошибки
CVErr (функция)	Возвращает значение типа Variant с подтипом Error, содержащее код ошибки, указанный пользователем

## Типы данных

Тип переменной	Описание
Boolean	16-разрядные (двухбайтовые) числа, которые могут иметь только два значения : True или False
Byte	8-разрядные (1 байт) числа без знака в диапазоне от 0 до 255
Currency	64-разрядные (8-байтовые) целые числа, которые после деления на 10000 дают число с фиксированной десятичной точкой с 15 разрядами в целой части и 4 разрядами в дробной ( Денежный тип, используется для финансовых расчетов. Диапазон изменения +/- 9*10 <sup>14</sup>
Date	64-разрядные (8-байтовые) числа с плавающей точкой стандарта IEEE, представляющие даты из диапазона от 1 января 100 года до 31 декабря 9999; и значения времени от 0:00:00 до 23:59:59
Double	64-разрядные (8-байтовые) числа с плавающей точкой стандарта IEEE в диапазоне от -3,402823E38 до -4,940 656 458 412 47E-324 для отрицательных значений и от 4.940 656 458 412 47E-324 до 1,797 693 134 862 32E308 для положительных значений
Integer	16-разрядные (2-байтовые) числа в диапазоне от -32768 до 32767
Long	32-разрядные (4-байтовые) числа со знаком в диапазоне от -2 147 483 648 до 2 147 483 647
Object	32- разрядные (4-байтовые) адреса, которые содержат ссылки на объекты
Single	32-разрядные (4-байтовые) числа с плавающей точкой стандарта IEEE в диапазоне от -3,402823E38 до -1, 401298E-45 для отрицательных значений и от 1, 401298E-45 до 3,402823E38 для положительных значений
String	Кодами для символов, образующих значения типа String, служат целые числа от 0 до 255. Строки переменной длины могут содержать 2 <sup>31</sup> символов; Строки постоянной длины могут содержать от 1 до 2 <sup>16</sup> символов
Variant	К данным типа Variant относятся все переменные, не описанные явно с другим типом данных (с помощью операторов Dim, Private, Public или Static)

## Приложение 4

### Классы и операторы Visual Basic<sup>9</sup>

<i>Класс</i>	<i>Наименование класса</i>
CheckBox	Флажок
ComboBox	Поле со списком
CommandButton	Командная кнопка
Common Dialog	Окно диалога Windows, используется для организации открытия файлов и сохранения их на диске, настройки принтера и печати документов, настройки параметров шрифтов и установки цвета
Control	Обобщенное имя объектов, помещенных на форму
DirListBox	Список каталогов
DriveListBox	Список дисков
ErrObject	Содержит код ошибки и системное сообщение об ошибке, а также некоторую другую информацию
FileListBox	Список файлов
Form	Форма
Frame	Рамка
Global	Абстрактный объект, обозначающий компьютер
HScrollBar	Горизонтальная линейка прокрутки
Image	Изображение
Label	Метка
Line	Линия
ListBox	Список
MDI-Form	Форма – контейнер для всех дочерних форм приложения
Menu	Меню
MSFlexGrid	Сетка. Служит для вывода данных
OptionButton	Переключатель
PictureBox	Окно с рисунком
Printer	Печатающее устройство
Screen	Экран Windows
Shape	Абстрактный объект, предназначенный для изображения геометрических фигур
Slider	Регулятор
StatusBar	Строка состояния
TextBox	Текстовое поле
Timer	Часы
ToolBar	Панель инструментов
VScrollBar	Вертикальная линейка прокрутки

<sup>9</sup> Приведен перечень классов и операторов, упоминаемых в учебном пособии

<b>Операторы и методы</b>	<b>Назначение</b>
Call	Вызов процедуры
Circle	Рисование окружностей
Cls	Очистка формы
Debug.Printer	Вывод информации в окно отладки
DefBool	Объявление переменной типа Boolean
DefByte	Объявление переменной типа Byte
DefCur	Объявление переменной типа Currency
DefDate	Объявление переменной типа Date
DefDbl	Объявление переменной типа Double
DefInt	Объявление переменной типа Integer
DefLng	Объявление переменной типа Long
DefSng	Объявление переменной типа Single
DefStr	Объявление переменной типа String
DefObj	Объявление переменной типа Object
DefVar	Объявление переменной типа Variant
Do Loop	Оператор цикла
For/Each	Оператор цикла
For/Next	Оператор цикла
Hide	Скрытие формы
If/Then/Else/End If	Условный оператор
Let	Оператор присваивания
Line	Рисование линии
Load	Загрузка формы
Load Picture	Загрузка рисунка
Move	Перемещение
OLEDrag	Перемещение объекта с помощью мыши
PaintPicture	Перерисовка изображения
Point	Определение цвета точки графического объекта
PopupMenu	Вывод контекстного меню
Print	Вывод информации на форму или на принтер
Print Form	Печать формы
Pset	Рисование точки
Refresh	Обновление изображения
Select Case	Оператор множественного выбора
Scale	Установка масштаба графического объекта
ScaleX	Установка масштаба графического объекта по оси X
ScaleY	Установка масштаба графического объекта по оси Y
SetFocus	Установка фокуса
Show	Показать объект
Unload	Выгрузить объект
While/Wend	Оператор цикла типа "Пока"
ZOrder	Изменение порядка элементов управления

## Приложение 5

### Моделирование движения механизма

Разработка программы

“Моделирование движения механизма”

#### Задание

Разработать программу для моделирования движения механизма. Программа должна обеспечивать воспроизведение механизма в исходном состоянии, в динамике (несколько положений), строить траекторию заданной точки, определять область допустимых перемещений звеньев механизма и область, занимаемую механизмом при движении.

#### П5.1. Пояснительная записка

##### П5.1.1. Краткие теоретические сведения

Под движением механизма понимается воспроизведение положения механизма в последовательные моменты времени. При моделировании движения механизма рассматривают следующие вопросы:

1. Воспроизведение положения механизма в исходном состоянии.
2. Воспроизведение положения механизма в нескольких промежуточных положениях.
3. Определение зон движения звеньев механизма.
4. Определение области, занимаемой механизмом в процессе движения.
5. Построение траектории движения заданной точки.

Основными элементами механизма являются (рис.П5.1): кривошип (1), шатун (2, 3), ролик (4) с радиусом (5), ползун (6).

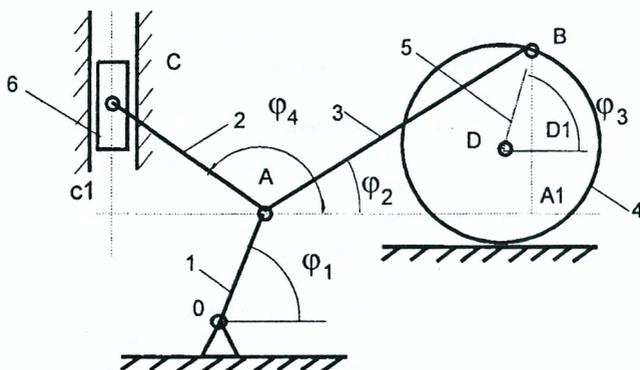


Рис. П5.1. Механизм

### П5.1.2. Исходные данные

К исходным данным относятся координаты опор, ползуна, ролика, длины звеньев механизма. Исходные данные удобно задавать в пикселях, тем более что координаты опор на экране иначе задать невозможно. Масштаб преобразования для линейных величин прием равным 100:1, то есть 100 пикселей равны 1 м.

Координаты опоры O : X=100, Y=50, угол  $\varphi_1 = 45$  градусов, длина звена OA=0,40 м. Длина звена AB=1,5 м. Координата X точки C равна 100, длина звена CA=0,8 м. Высота расположения опоры E для ролика равна 0,60 м. Радиус ролика DB равен 0,4 м.

Исходные данные сведем для удобства использования в таблицу (табл. П7.1). При записи данных в таблицу проведем преобразование линейных размеров звеньев механизма с учетом принятого масштаба.

Таблица П5.1

<i>Исходные данные</i>				
	X	Y	F	L
O	100	50	45	40
A	-	-	-	150
B	-	-	-	-
C	100	-	-	80
D	-	60+40	-	40
E	-	60	-	-

В табл.П5.1 номера строк обозначены номерами точек звеньев механизма O, A, B, C, D, E, номера столбцов обозначают координаты точек X, Y, угол положения звена механизма - F и длину звена механизма - L. Значения неизвестных параметров обозначены в таблице знаком "-".

Состояние механизма в статике определяется положением кривошипа (угол  $\varphi_1$ ).

Расчету в данном механизме подлежат значения координат точек A, B, координаты  $Y_c$  для точки C и  $X_d$  для точки D. В качестве промежуточных результатов необходимо иметь значения углов  $\varphi_2$  и  $\varphi_3$ .

### П5.1.3. Разработка математической модели

Координаты точки A определяются по длине кривошипа OA и значению угла  $\varphi_1$ :

$$X_a = X_o + L_1 \cos \varphi_1 \quad (5.1.1)$$

$$Y_a = Y_o + L_1 \sin \varphi_1 \quad (5.1.2)$$

Координата Y точки C определяется из прямоугольного треугольника SAC<sub>1</sub>. Длина CA известна, а  $C_1A = X_a - X_c$ , тогда

$$Y_c = Y_a + \sqrt{C A^2 - C_1 A^2} \quad (5.1.3)$$

В данном механизме ролик имеет две степени свободы. Он может вращаться вокруг своей оси и перемещаться в горизонтальном направлении. В исходных данных высота расположения ролика и радиус ролика заданы, а следовательно и вертикальная координата центра ролика  $Y_d$  известна. Горизонтальное положение ролика  $X_d$  - неизвестно. Наибольшие трудности возникают при определении значения угла  $\varphi_2$ .

Для определения значения угла  $\varphi_2$  необходимо составить и решить систему из пяти уравнений:

$$x_b = x_a + l_2 \cos \varphi_2 \quad (5.1.4)$$

$$x_b = x_d + l_3 \cos \varphi_3 \quad (5.1.5)$$

$$y_b = y_a + l_2 \sin \varphi_2 \quad (5.1.6)$$

$$y_b = y_d + l_3 \sin \varphi_3 \quad (5.1.7)$$

$$(x_b - x_d)^2 + (y_b - y_d)^2 = l_3^2 \quad (5.1.8)$$

Здесь  $l_2$  - длина шатуна АВ,  $l_3$  - радиус ролика. Первые четыре уравнения получают из рассмотрения прямоугольников  $ABB_1$  и  $DBD_1$ . Пятое уравнение составляется из следующих соображений: координаты точки В лежат на поверхности окружности, поэтому можно записать уравнение окружности с центром в точке D и радиусом, равным радиусу ролика  $l_3$ .

Из уравнений 5.1.4 и 5.1.5 находим значение Xd:

$$x_d = x_a + l_2 \cos \varphi_2 - l_3 \cos \varphi_3 \quad (5.1.9)$$

Из уравнений 5.1.6 и 5.1.7 находим значение угла  $\varphi_3$ :

$$\varphi_3 = \arcsin((y_a + l_2 \sin \varphi_2 - y_d) / l_3) \quad (5.1.10)$$

Заменив в выражении (5.1.9) угол  $\varphi_3$  на его значение из (5.1.10) и подставив значения  $X_b$  (5.1.4),  $X_d$  и  $Y_b$  (5.1.9) в (5.1.6) получим уравнение

$$\begin{aligned} & ((x_a + l_2 \cos \varphi_2 - (x_a + l_2 \cos \varphi_2 - l_3 \cos(\arcsin((y_a + l_2 \sin \varphi_2 - y_d) / l_3)))^2 + \\ & + (y_a + l_2 \sin \varphi_2 - y_d)^2) - l_3^2 = 0. \end{aligned} \quad (5.1.11)$$

В выражении (5.1.11) один неизвестный параметр - угол  $\varphi_2$ . Найти значение угла  $\varphi_2$  из данного уравнения можно с помощью одного из методов поиска значения корня, например методом итераций.

#### П5.1.4. Схема алгоритма программы

Список задач, выполняемых программой, полностью вытекает из задания. В соответствии с этим в программе необходимо предусмотреть следующие функциональные блоки:

Паспорт программы;

Меню;

Блок моделирования механизма в статике;

Блок моделирования механизма в динамике;

Блок моделирования движения заданной точки или произвольной точки механизма;

- Блок определения зон движения звеньев механизма и механизма в целом.

Все блоки независимы друг от друга. Вызов блоков программы осуществляется из меню пользователя и после окончания работы блока управление передается меню программы. Указанная особенность программы полностью согласуется с принципами структурного программирования, характерными для программирования в среде Visual Basic. Схема алгоритма приведена на рис. П5.2, где обозначено:

n=1 - переход на метку M1 - построение механизма в статике;

n=2 - переход на метку M2 - построение механизма в динамическом режиме;

n=3 - переход на метку M3 - построение зон действия звеньев механизма;

n=4 - переход на метку M4 - построение зоны занимаемой механизмом;

n=5 - переход на метку М5 - построение траектории движения точки;  
 n=6 - выход из программы.

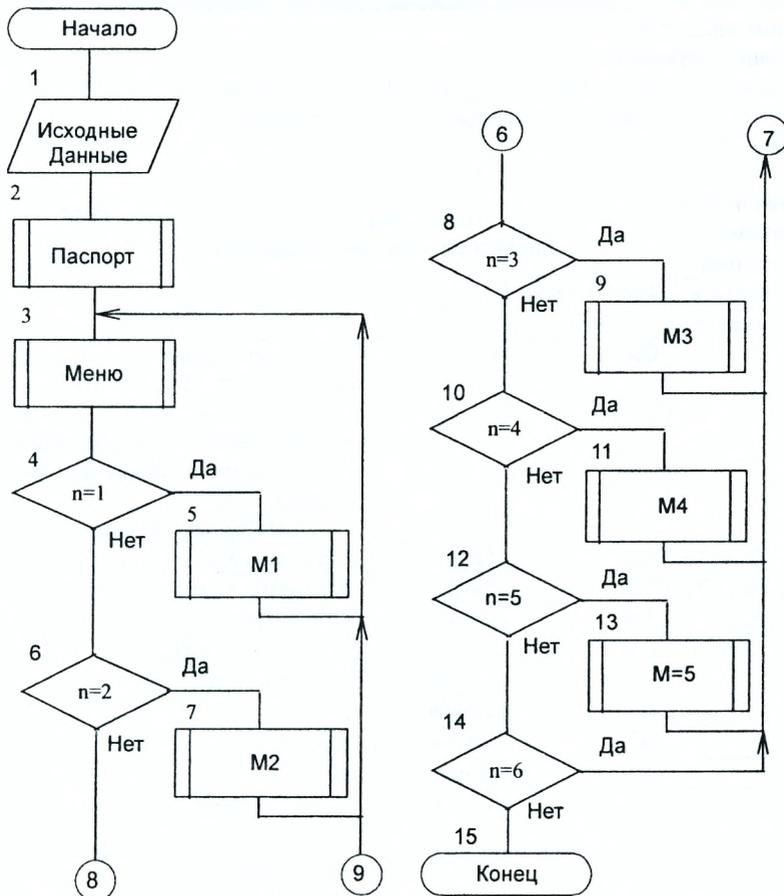


Рис. П5.2 . Схема алгоритма построения механизма

### П5.1.5. Разработка проекта программы

Программа должна состоять из нескольких форм в соответствии с определенными задачами. Совместную работу форм удобно организовать с помощью MDI-формы. В эту форму можно поместить меню пользователя и фоновый рисунок.

Эскизы форм представлены на рис. П5.3 – П5.6. Так как в задании не оговорены требования к интерфейсу пользователя при моделировании движения механизма, то состав элементов управления на форме разработчик определяет в соответствии со своим

представлением об удобстве управления демонстрацией свойств механизма. В простейшем случае можно ограничиться двумя кнопками: старт и выход. В учебных целях при разработке интерфейса целесообразно использовать как можно больше различных элементов управления.

### Описание переменных

Опишем глобальные переменные, используемые в программе (табл. П5.2)

Аналогично описываются переменные процедур и функций.

Таблица П5.2

### Глобальные переменные

Наименование программы Mechanizm	Тема проекта: Моделирование движения механизма			Лист _1_
	Листов __			
Обозначение переменной	Имя переменной	Тип	Примечание	
	Obj	Object	Глобальная переменная типа Object, заменяет имя объекта при вызове процедур, осуществляющих графические построения. В каждой форме, в которой рисуется механизм, введен оператор: Set Obj=<имя формы>	
Xo, Xa, Xb, Xc, Xd	X(5)	Single	Координаты X узлов механизма	
Yo, Ya, Yb, Yc, Yd	Y(5)	Single	Координаты Y узлов механизма	
$\varphi_1, \varphi_2, \varphi_3, \varphi_4$	f(5)	Single	Значения углов	
$l_1, l_2, l_3, l_4, l_5$	L(5)	Single	Длины звеньев механизма	
	fi	Single	Текущий угол поворота	
	pi	Double	число $\pi$	
	pi2	Double	число $2\pi$	
	dx	Single	шаг	
	Ft	Single	Вспомогательная переменная, хранит вычисленное значение угла $\varphi_2$	
	i1	Double	Максимальный угол поворота кривошипа $i1=2*\pi$	
	f1	Byte	Флаг,	
	f12	Byte	Флаг,	
	xo	Integer	Вспомогательная переменная, для хранения целых чисел, используется при построении опор.	
	yo	Integer	то же	
	xr	Integer	то же	
	yr	Integer	то же	
	n	Integer	то же	

## Описание элементов управления

Опишем основные свойства элементов управления формы Статика (табл. П5.3). Аналогично описываются элементы управления и других форм.

Таблица П5.3.

Свойства элементов управления формы Статика

Наименование программы Mechanizm	Тема проекта: Моделирование движения механизма		Лист _2_
			Листов __
Объект	Свойство	Значение	
Форма	Name	FormStatica	
	Caption	Статика	
	AutoRedraw	True	
	MDIChild	2 ' Maximize	
	WindowState	True	
Командная кнопка 1	Name	Command1Statica	
	Caption	Старт	
	Height	375	
	Width	1335	
Командная кнопка 2	Name	Command2Statica	
	Caption	Выход	
	Height	375	
	Width	1335	
Сетка	MSFlexGrid	MSFlexGridStatic	
	Rows	6	
	Cols	3	
Надпись	Name	LabelStatica	
	Caption	"Укажите угол поворота в градусах"	
	Alignment	2 ' Center	
Текстовое поле	Name	TextStatica	
	Text	0	

## П5.2. Текст программы

Текст программы MDI-формы

```
Private Sub MenuEnd_Click()
```

```
End
```

```
End Sub
```

```
Private Sub MenuDinamika_Click()
```

```
FormDinamic.Show
```

```
End Sub
```

```
Private Sub MenuPasport_Click()
```

```
FormPasport.Show
```

```
End Sub
```

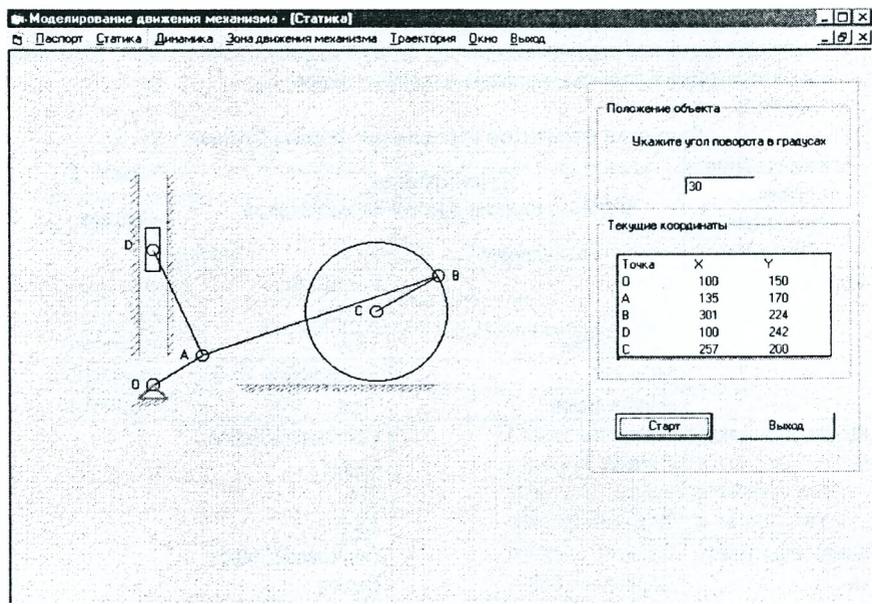


Рис.П5.3. Форма для демонстрации механизма в статическом режиме

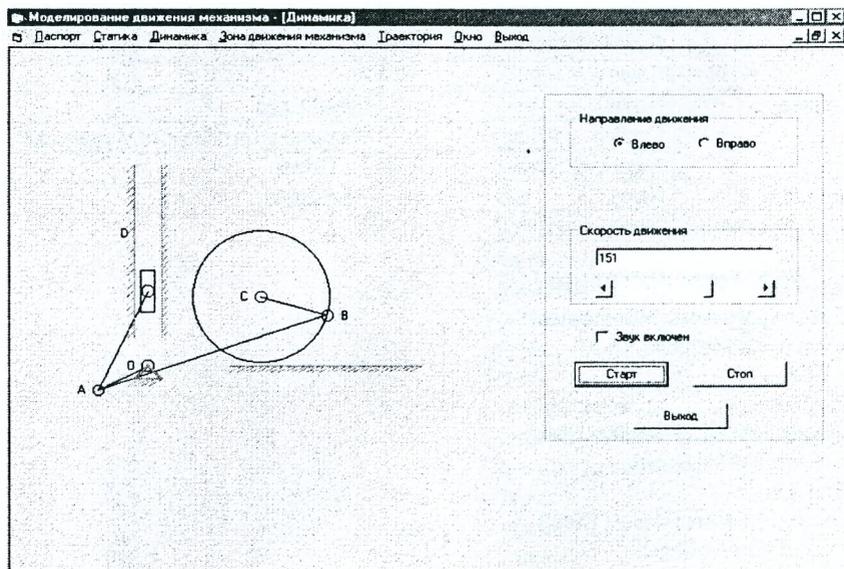


Рис.П5.4. Форма для демонстрации движения механизма

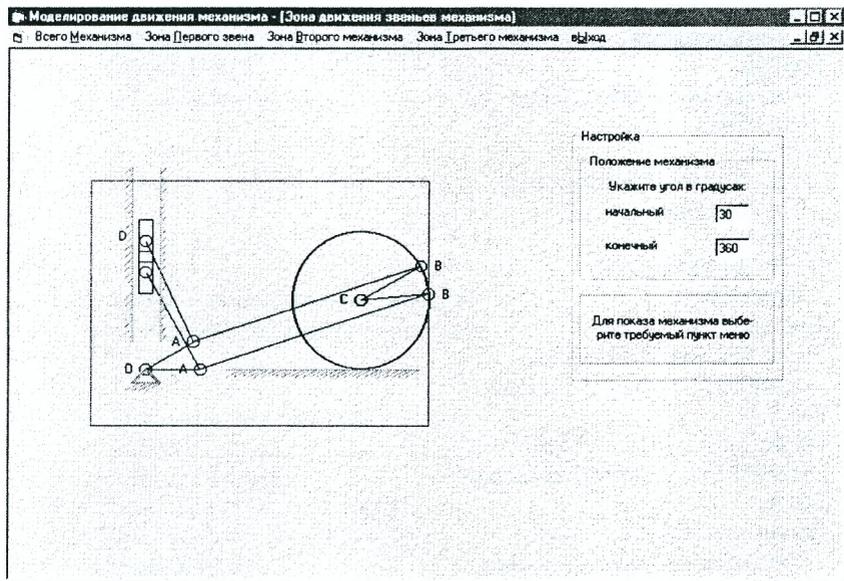


Рис. П5.5. Форма для демонстрации зон действия механизма и его звеньев

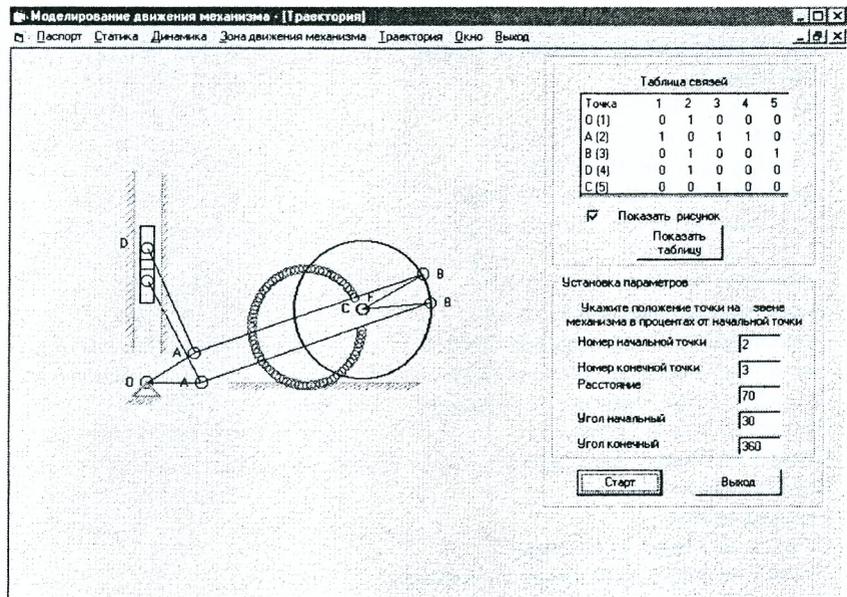


Рис. П5.6. Форма для демонстрации траектории движения заданной точки

```

Private Sub MenuSonaDvigenij_Click()
    FormZona.Show
End Sub
Private Sub MenuStatika_Click()
    FormStatika.Show
End Sub
Private Sub MenuTraektorij_Click()
    FormTrec.Show
End Sub
Private Sub Form_Load()
    VvodDannych
End Sub
Private Sub mnuCascade_Click()
    frmMDIForm.Arrange vbCascade
End Sub
Private Sub mnuHorizont_Click()
    frmMDIForm.Arrange vbTileHorizontal
End Sub
Private Sub mnuVertical_Click()
    frmMDIForm.Arrange vbTileVertical
End Sub

```

### **Текст программы Модуля**

```

Rem Объявление переменных и массивов
Option Base 1
Global Obj As Object
Global X(5) As Single, Y(5) As Single
Global f(5) As Single, l(5) As Single
Global dlina1 As Single, dlina2 As Single
Global dlina3 As Single, dlina4 As Single
Global msg0 As String, msg1 As String
Global msg2 As String, msg3 As String
Global msg4 As String, msg11 As String
Global msg12 As String, msg13 As String
Global msg14 As String, msg15 As String
Global fi As Single ' Текущий угол поворота
Global pi As Double ' Константа
Global i1 As Double ' Константа
Global fl As Byte
Global fl2 As Byte
Global xo As Integer, yo As Integer
Global xr As Integer, yr As Integer, n As Integer
Global pi2 As Single, dx As Single, Ft As Single

```

```

Rem процедура ввода данных
Public Sub VvodDannych()
y11 = 500: y12 = 4 / 3 * y11
X(1) = 100: Y(1) = 150: f(1) = 45: l(1) = 40
l(2) = 175
X(4) = 100: l(4) = 80
Y(5) = 200: l(5) = 50
Ft = 30
pi = Atn(1) * 4
pi2 = 2 * pi
i1 = pi2: 'максимальный угол поворота кривошипа
dx = 0.3: 'шаг построения изображения в динамике
End Sub

```

```

Rem процедура вычисления координат
Public Sub Raschet()
Dim x1 As Single, y1 As Single
Dim b1 As Single, f1 As Single, e As Single
Dim s1 As Single, ss1 As Single, df As Single
Dim ss2 As Single, f2 As Single
Dim a As Single, b As Single
f1 = fi
X(2) = X(1) + l(1) * Cos(f1)
Y(2) = Y(1) + l(1) * Sin(f1)
b1 = l(4) ^ 2 - (X(2) - X(4)) ^ 2
If b1 < 0 Then GoTo m1
Y(4) = Sqr(b1) + Y(2)
Rem вычисление значения угла f2
e = 1: Rem Точность поиска
Rem цикл изменения значения угла f1
s1 = e + 1: df = 0.01: f2 = pi / 10
Rem цикл изменения значения угла f2
While Abs(s1) > e And f2 < pi / 2
a = (Y(2) - Y(5) + l(2) * Sin(f2)) / l(5)
B = a * l(5)
If (1 - a ^ 2) <= 0 Then GoTo m1
ss1 = Cos(Atn(a / Sqr(1 - a ^ 2)))
ss2 = (X(2) + l(2) * Cos(f2) - (X(2) + l(2) * Cos(f2) - l(5) * ss1))
s1 = l(5) ^ 2 - (ss2 ^ 2 + B ^ 2)
GoTo m2
m1:
m2:
Ft = f2
f2 = f2 + df

```

Wend

```
Rem Расчет координат
f2 = Ft: f(2) = f2
a = (Y(2) - Y(5) + l(2) * Sin(f2)) / l(5)
If (1 - a ^ 2) <= 0 Then GoTo m3
f(3) = Atn(a / Sqr(1 - a ^ 2))
X(3) = X(2) + l(2) * Cos(f2)
Y(3) = Y(2) + l(2) * Sin(f2)
X(5) = X(3) - l(5) * Cos(f(3))
```

m3:

End Sub

Rem построение рисунка

Public Sub Postroenie()

```
Dim a1 As Single, a2 As Single, b1 As Single, b2 As Single
Obj.Circle (X(1), Y(1)), 4
Obj.Line (X(1), Y(1))-(X(2), Y(2))
Obj.Circle (X(2), Y(2)), 4
Obj.Line (X(2), Y(2))-(X(3), Y(3))
Obj.Circle (X(3), Y(3)), 4
Obj.Circle (X(5), Y(5)), l(5)
Obj.Circle (X(5), Y(5)), 4
Obj.PSet (X(5), Y(5))
Obj.Circle (X(4), Y(4)), 4
Obj.Line (X(2), Y(2))-(X(4), Y(4))
Obj.Line (X(5), Y(5))-(X(3), Y(3))
a1 = X(4) - 5: a2 = X(4) + 5: b1 = Y(4) - 15: b2 = Y(4) + 15
Obj.Line (a1, b1)-(a2, b2), , B
End Sub
```

\*\*\*\*\*

Rem построение опорных точек

Public Sub Psopor()

' опора кривошипа

```
Obj.CurrentX = X(1) - 15: Obj.CurrentY = Y(1) + 5
Obj.Print "O"
Obj.CurrentX = X(4) - 20: Obj.CurrentY = Y(1) + 100
Obj.Print "D"
Obj.CurrentX = X(2) - 15: Obj.CurrentY = Y(2) + 5
Obj.Print "A"
Obj.CurrentX = X(3) + 10: Obj.CurrentY = Y(3) + 5
Obj.Print "B"
Obj.CurrentX = X(5) - 15: Obj.CurrentY = Y(5) + 5
Obj.Print "C"
xo = X(1): yo = Y(1)
```

```

xr = xo - 15: yr = yo - 15: n = 4
Form_опора
Form_StrichG
' левая направляющая ползуна
xo = X(1) - 10: yo = Y(1) + 20
xr = xo: yr = yo: n = 25
Obj.Line (xo, yo)-(xo, yo + 125), QBColor(6)
xr = xo - 5
Form_StrichV
' правая направляющая ползуна
xo = X(1) + 10: yo = Y(1) + 20
xr = xo: yr = yo: n = 25
Obj.Line (xo, yo)-(xo, yo + 125), QBColor(6)
Form_StrichV
' Каток
xo = X(1) + 60: yo = Y(5) - 1(5)
xr = xo: yr = yo - 5: n = 27
Obj.Line (xo, yo)-(xo + 140, yo), QBColor(6)
Form_StrichG
End Sub
*****

Rem построение опоры
Public Sub Form_опора()
    Obj.PSet (xo, yo), QBColor(6) 'коричневый
    Obj.DrawWidth = 2
    Obj.Line -(xo - 10, yo - 10), QBColor(6)
    Obj.Line -(xo + 10, yo - 10), QBColor(6)
    Obj.Line -(xo, yo), QBColor(6)
End Sub
Rem горизонтальная штриховка
Public Sub Form_StrichG()
Dim ii As Integer
For ii = 1 To n
    Obj.DrawWidth = 1
    Obj.Line (xr, yr)-(xr + 10, yr + 5), QBColor(6)
    xr = xr + 5
Next ii
End Sub
Rem вертикальная штриховка
Public Sub Form_StrichV()
Dim ii As Integer
For ii = 1 To n
    Obj.DrawWidth = 1
    Obj.Line (xr, yr)-(xr + 5, yr + 5), QBColor(6)
    yr = yr + 5
Next ii
End Sub

```

### Текст программы формы Статика

```
Dim tc1(6, 3) As String, Ft As Integer
Private Sub Form_Load()
Set Obj = FormStatica
VvodDannych
tc1(1, 1) = "Точка": tc1(1, 2) = "X": tc1(1, 3) = "Y"
tc1(2, 1) = "O": tc1(3, 1) = "A": tc1(4, 1) = "B":
tc1(5, 1) = "D": tc1(6, 1) = "C"
End Sub
Private Sub Command1Statica_Click()
Dim rx As Integer, ry As Integer, i As Integer
Dim ax As Single, ay As Single
ry = 380: rx = 1.6 * ry
Scale (0, ry)-(rx, 0)
Statica
MSFlexGrid1Static.ColAlignment(0) = 3
MSFlexGrid1Static.ColWidth(0) = 1000
MSFlexGrid1Static.ColAlignment(0) = 3
For i = 1 To 2
MSFlexGrid1Static.ColAlignment(i) = 3
Next i
For i = 1 To 3
MSFlexGrid1Static.Col = i - 1
MSFlexGrid1Static.Row = 0
MSFlexGrid1Static.Text = tc1(1, i)
Next i
For i = 2 To 6
MSFlexGrid1Static.TextMatrix(i-1,0) = tc1(i, 1)
Next i
For i = 1 To 5
ax =Round(X(i),0)
MSFlexGrid1Static.TextMatrix(i, 1) = Str$(ax)
MSFlexGrid1Static.Col = 2
ay = Round(Y(i), 0)
MSFlexGrid1Static.TextMatrix(i,2) = Str$(ay)
Next i
End Sub
*****
' Rem построение механизма в статике
Public Sub Statica()
Cls
Ft = Val(TextStatic.Text)
fi = Ft / 180 * pi
Raschet
```

```

    Postroenie
    Psopor
End Sub
*****
Private Sub Command2Statica_Click()
    Unload Me
End Sub

```

### Текст программы формы Динамика

```

Dim TT As Integer
Dim TT1 As Integer, Scor As Double, FlagDin As Byte
Dim inf1(200) As Integer, inf2(200) As Integer

```

```

Private Sub ComDinEnd_Click()
    If FlagDin = 1 Then
        MsgBox "Остановите механизм", , "Динамика"
        Exit Sub
    Else
        Unload Me
    End If
End Sub

```

```

Private Sub Form_Click()
    Unload Me
End Sub

```

```

Private Sub Form_Load()
    Dim beep1 As Integer, i As Integer
    For i = 1 To 200: inf1(i) = i: inf2(i) = 201 - i: Next i
    Set Obj = FormDinamic
    VvodDannych
    HScroll1Dinamic.Value = 50
    Scor = HScroll1Dinamic.Value * 100
    ComDinStop.Cancel = False
    FlagDin = 1
End Sub
*****

```

Rem построение механизма в динамике

```

Private Sub Dinamica()
    Dim beep1 As Integer, i As Integer
    Cis
    dx = 0.3: beep1 = 0
    dinam2:
    Scor = HScroll1Dinamic.Value * 100

```

```

If Option1Dinamic = True Then
For fi = 0 To pi2 Step dx
Cls
beep1 = beep1 + 1
Scor = HScroll1Dinamic.Value * 100
If Scor <= 2 Or Scor / 100 > 148 Then
MsgBox "Предельная скорость", , "Динамика"
GoTo dinam1
End If
If ComDinStop.Cancel = True Then FlagDin = 0: Exit Sub
  TT = 0: TT1 = 0:
  Raschet
  Postroenie
  Psopor
  If CheckDinam = Checked Then
    If beep1 = 5 Then
      Beep
    End If
  End If
  While TT < Scor
    Timer1_Timer
    TT = TT + TT1
    DoEvents
  Wend
  For i = 1 To 200
    If Int(TT / 100) = inf1(i) Then
      TextDinamic1.Text = inf2(i)
    End If
  Next i
  If beep1 = 5 Then beep1 = 0
Next fi
End If
If Option2Dinamic = True Then
For fi = 2 * pi To Ft / 180 * pi Step (-dx)
Cls
beep1 = beep1 + 1
Scor = HScroll1Dinamic.Value * 100
If Scor <= 2 Or Scor / 100 > 148 Then
MsgBox "Предельная скорость", , "Динамика"
GoTo dinam1
End If
If ComDinStop.Cancel = True Then FlagDin = 0: Exit Sub
  TT = 0: TT1 = 0
  Raschet
  Postroenie
  Psopor
  If CheckDinam = Checked Then

```

```

    If beep1 = 5 Then
        Beep
    End If
End If
While TT < Scor
    Timer1_Timer
    TT = TT + TT1
DoEvents
Wend

For i = 1 To 200
    If Int(TT / 100) = inf1(i) Then
        TextDinamic1.Text = inf2(i)
    End If
Next i
If beep1 = 5 Then beep1 = 0
Next fi
End If
    GoTo dinam2
dinam1:
HScroll1Dinamic.Value = 50
TextDinamic1.Text = HScroll1Dinamic.Value
End Sub

```

\*\*\*\*\*

```

Private Sub ComDinStart_Click()
    Dim rx As Integer, ry As Integer
    ry = 380: rx = 1.6 * ry
    Scale (0, ry)-(rx, 0)
    Dinamica
End Sub

```

```

Private Sub ComDinStop_Click()
    ComDinStop.Cancel = True
End Sub

```

```

Rem Таймр
Private Sub Timer1_Timer()
    Static TimerTimes As Integer
    TimerTimes = TimerTimes + 1
    If TimerTimes = 2 Then
        TimerTimes = 0
    Else
        TT1 = TimerTimes
    End If
Exit Sub
End If
End Sub

```

## Текст программы формы определения зон действия механизма

Dim k As Byte

-----  
Private Sub Form\_Click()

    Unload Me

End Sub

Private Sub Form\_Load()

    FormZona.Height = 7095

    Set Obj = FormZona

End Sub

Private Sub MenuZona1\_Click()

    Cls

    fl = 0

    fl2 = 1

    Zonazv

End Sub

Private Sub MenuZona2\_Click()

    Cls

    fl = 0

    fl2 = 2

    Zonazv

End Sub

Private Sub MenuZona3\_Click()

    Cls

    fl = 0

    fl2 = 3

    Zonazv

End Sub

Private Sub MenuZonaEnd\_Click()

    Unload Me

End Sub

Private Sub MenuZonaVsego\_Click()

    Cls

    fl = 1

    Zonazv

End Sub

Rem определение зоны действия звеньев механизма

Private Sub Zonazv()

    Dim rx As Integer, ry As Integer

    Dim i As Integer, j As Integer, DrawStile As Integer

```

Dim maxx0 As Single, minx0 As Single
Dim maxy0 As Single, miny0 As Single
VvodDannych
ry = 380: rx = 1.6 * ry
Scale (0, ry)-(rx, 0)
Dim xa(100, 5) As Single, ya(100, 5) As Single
Dim maxx(5) As Single, maxy(5) As Single
Dim minx(5) As Single, miny(5) As Single

```

```
fi = 0
```

```
Raschet
```

```
For i = 1 To 5
```

```
    If i = 5 Then X(i) = X(i) + l(5)
```

```
    maxx(i) = X(i): maxy(i) = Y(i)
```

```
    minx(i) = X(i): miny(i) = Y(i)
```

```
Next i
```

```
i = 0: dx = 0.3
```

```
For fi = 0 To pi2 Step dx
```

```
    Raschet
```

```
    i = i + 1
```

```
    For j = 1 To 5
```

```
        xa(i, j) = X(j)
```

```
        ya(i, j) = Y(j)
```

```
        If xa(i, j) > maxx(j) Then maxx(j) = xa(i, j)
```

```
        If xa(i, j) < minx(j) Then minx(j) = xa(i, j)
```

```
        If ya(i, j) > maxy(j) Then maxy(j) = ya(i, j)
```

```
        If ya(i, j) < miny(j) Then miny(j) = ya(i, j)
```

```
    Next j
```

```
Next fi
```

```
If fl = 1 Then GoTo Zona2
```

```
    Select Case fl2
```

```
        Case 1
```

```
            DrawStile = 2
```

```
            Line (minx(2), maxy(2))-(maxx(2), miny(2)), QBColor(2), B
```

```
        Case 2
```

```
            DrawStile = 4
```

```
            Line (minx(2), maxy(4) + 15)-(maxx(2), miny(2)), QBColor(3), B
```

```
        Case 3
```

```
            DrawStile = 5
```

```
            Line (minx(2), Y(5) + l(5))-(maxx(3), miny(2)), QBColor(5), B
```

```
    End Select
```

```
    GoTo Zona1
```

```
Zona2:
```

```
    maxx0 = maxx(1): maxy0 = maxy(1)
```

```

minx0 = minx(1): miny0 = miny(1)
For j = 1 To 5
    If maxx(j) > maxx0 Then maxx0 = maxx(j)
    If maxy(j) > maxy0 Then maxy0 = maxy(j)
    If minx(j) < minx0 Then minx0 = minx(j)
    If miny(j) < miny0 Then miny0 = miny(j)
Next j
DrawStile = 3
Line (minx0, maxy0 + 15)-(maxx0, miny0), QBColor(4), B
Zona1:
k = 0
Ft = Val(TextZonaNach.Text)
fi = Ft / 180 * pi
Raschet
Postroenie
Psopor
k = 1
Ft = Val(TextZonaCon.Text)
fi = Ft / 180 * pi
Raschet
Postroenie
Psopor
Erase xa, ya
fi = 0
End Sub

```

### **Текст программы формы демонстрации траектории заданной точки**

```

Dim tc(5, 5) As Integer, XT(10) As Integer
Dim tc1(6, 6) As String, Ft As Integer

```

```

Private Sub Form_Load()

```

```

    Rem таблица связей

```

```

    Set Obj = FormTrec

```

```

    VvodDannych

```

```

        tc1(1, 1) = "Точка": tc1(1, 2) = "1": tc1(1, 3) = "2"

```

```

        tc1(1, 4) = "3": tc1(1, 5) = "4": tc1(1, 6) = "5"

```

```

        tc1(2, 1) = "O (1)": tc1(3, 1) = "A (2)": tc1(4, 1) = "B (3)":

```

```

        tc1(5, 1) = "D (4)": tc1(6, 1) = "C (5)"

```

```

        tc(1, 2) = 1: tc(2, 3) = 1: tc(2, 4) = 1: tc(3, 5) = 1

```

```

        tc(2, 1) = 1: tc(3, 2) = 1: tc(4, 2) = 1: tc(5, 3) = 1

```

```

        Ft = 30

```

```

    End Sub

```

```

Private Sub ComTrecRis_Click()
Dim rx As Integer, ry As Integer
Traectorij
If CheckTrec.Value = Checked Then
    ry = 380: rx = 1.6 * ry
    Scale (0, ry)-(rx, 0)
    fi = Ft / 180 * pi
    Raschet
    Postroenie
    Psopor
Else
    Exit Sub
End If
End Sub

Private Sub ComTrec2_Click()
    Unload Me
End Sub

Private Sub ComTrec1_Click()
Dim rx As Integer, ry As Integer
ry = 380: rx = 1.6 * ry
Scale (0, ry)-(rx, 0)
    Traectorij
    Traectorij1
End Sub

Rem построение траектории движения точки
Private Sub Traectorij()
Dim i As Integer, j As Integer, k As Integer
Cls
MSFlexGridTrec.ColWidth(0) = 1000
k = (MSFlexGridTrec.Width - 1000) / 5
MSFlexGridTrec.ColAlignment(0) = 2
For i = 1 To 5
    MSFlexGridTrec.ColAlignment(i) = 2
    MSFlexGridTrec.ColWidth(i) = k
Next i
' CurrentX = 50: CurrentY = 10
For i = 1 To 6
MSFlexGridTrec.Col = i - 1
MSFlexGridTrec.Row = 0
MSFlexGridTrec.Text = tc1(1, i)
'Print MSFlexGridTrec.Text
    Next i
    For i = 2 To 6
MSFlexGridTrec.Col = 0
MSFlexGridTrec.Row = i - 1

```

```

MSFlexGridTrec.Text = tc1(i, 1)
'Print MSFlexGridTrec.Text
Next i
For i = 1 To 5
  For j = 1 To 5
    MSFlexGridTrec.Col = i
    MSFlexGridTrec.Row = j
    MSFlexGridTrec.Text = Str$(tc(i, j))
    'Print MSFlexGridTrec.Text
  Next j
Next i
End Sub

Rem построение траектории движения точки
Private Sub Traectorij1()
Dim f1 As Single, f2 As Single, i As Integer, j As Integer
Dim n2 As Integer, n3 As Integer, n1 As Integer
Dim x1 As Single, y1 As Single
dx = 0.3
fi = Val(TextTrecUgol1.Text) / 180 * pi
f1 = fi
Raschet
Postroenie
Psopor
traekt1:
If TextTrec1.Text = "" Or TextTrec2.Text = "" _
  Or TextTrec3.Text = "" Then
  DoEvents
  GoTo traekt1
End If
n1 = Val(TextTrec1.Text)
n2 = Val(TextTrec2.Text)
n3 = Val(TextTrec3.Text)
If n1 > 5 Or n2 > 5 Or n1 = 0 Or n2 = 0 Then
  MsgBox "НЕТ ТАКОЙ ТОЧКИ", , "Траектория"
  TextTrec1.Text = ""
  TextTrec2.Text = ""
  TextTrec3.Text = ""
  GoTo traekt1
End If
For i = 1 To 5
  For j = 1 To 5
    If tc(n1, n2) = 0 Then
      MsgBox "НЕТ СВЯЗИ МЕЖДУ ТОЧКАМИ", , "Траектория"
      TextTrec1.Text = ""
      TextTrec2.Text = ""
      TextTrec3.Text = ""

```

```

    GoTo traekt1
End If
Next j
Next i
    fi = Val(TextTrecUgol2.Text) / 180 * pi
    f2 = fi
    Raschet
    Postroenie
    Psopor
    'DoEvents
For fi = f1 To f2 Step dx / 4
    Raschet
    x1 = X(n1) + n3 / 100 * (X(n2) - X(n1))
    y1 = Y(n1) + n3 / 100 * (Y(n2) - Y(n1))
    If fi = f1 Then
        CurrentX = x1 + 10: CurrentY = y1 + 5
        Print "F"
    End If
    Circle (x1, y1), 3, QBColor(12)

Next fi
End Sub

```

### **Паспорт программы**

```

Private Sub Form_Load()
    Picture1.AutoRedraw = True
    Picture1.Height = 7800
End Sub

Private Sub Picture1_Click()
    Unload Me
End Sub

Private Sub Picture1_KeyPress(KeyAscii As Integer)
    If Str(KeyAscii) <> "" Then
        Unload Me
    End If
End Sub

```

### **П5.3. Список использованной литературы**

1. Новожилов И. В. , Зацепин М. Ф. Типовые расчеты по теоретической механике на базе ЭВМ. - М.: Высшая школа, 1986. - 136 с.: ил.
2. Дьяконов В. П. Применение персональных ЭВМ и программирование на языке Бейсик. - М.: Радио и связь, 1989. - 288 с.: ил.
3. Вычислительная техника, программирование и математическое моделирование: методические указания – Брест: БрПИ, 1991. - 34 с.

## Приложение 6

### Проверка и обработка пользовательского ввода.

#### Обработка ошибок

##### 6.1. Проверка и обработка пользовательского ввода

###### 6.1.1. Контроль ввода

При вводе числовых данных, дат и времени целесообразно обеспечить контроль ввода данных, чтобы избежать получения некорректных результатов или появления неустраиваемых ошибок выполнения программы, приводящих к прерыванию ее выполнения.

Существует два способа контроля вводимых данных:

на уровне полей - данные проверяются после заполнения каждого поля;

на уровне формы - данные проверяются после того, как заполнены все поля на форме.

При контроле данных на уровне полей рекомендуется сообщать пользователю об ошибке звуковым сигналом и текстовой информацией в строке состояния или окне MsgBox.

```
If Not IsNumeric(txtZipCode.Text) Then
```

```
    Beep ' звуковой сигнал
```

```
    StatusBar1.Panels("ErrorDescription").Text= " код должен быть числовой "
```

```
End If.
```

В данном фрагменте программы проверяется содержание информации в свойстве Text объекта txtZipCode и, если это значение не числовое, то выдается звуковой сигнал и выводится текстовое сообщение в строку состояния.

При контроле на уровне формы на экран желательно выводить форму с сообщением о всех обнаруженных ошибках. После обнаружения ошибки фокус должен быть установлен на поле, содержащее ошибку. Если несколько полей формы содержат ошибку, то фокус надо установить на первое поле по порядку ввода данных (меньшее значение свойства TabIndex).

Установку фокуса можно выполнить программно с помощью метода SetFocus: txtZipCode.SetFocus

Для контроля ввода на уровне полей может использоваться событие потери фокуса LostFocus:

```
Private Sub txtZipCode_LostFocus ( )
```

```
    If Not IsNumeric (txtZipCode.Text) Then
```

```
        MsgBox " код должен быть числовой "
```

```
        txtZipCode.SetFocus
```

```
    End If
```

```
End Sub
```

Недостатком этого метода может быть опасность войти в бесконечный цикл.

###### 6.1.2. Реализация проверки данных на уровне формы

###### Обработчик клавиатуры на уровне формы

Обработчик клавиатуры - это более совершенный вариант проверки данных на уровне формы. При наличии единого обработчика клавиатуры можно управлять вводом данных в любые поля на форме.

Для реализации обработчика необходимо использовать события, связанные с клавиатурой. Можно задействовать три основных события: KeyPress, KeyDown и KeyUp.

#### **Использование события KeyPress**

Это событие возникает, когда пользователь нажимает клавишу, у которой имеется свой ASCII-код. К таким клавишам относятся все алфавитно-цифровые клавиши и ряд управляющих, например Enter (код 13), Backspace (код 8). Событие KeyPress не генерируется при нажатии функциональных и других специальных клавиш.

Событие KeyPress удобно для перехвата клавиш, нажимаемых при вводе в поля TextBox и ComboBox. Событие KeyPress предоставляет аргумент KeyAscii, в котором содержится код нажатой клавиши. Если нужно запретить ввод любых данных кроме числовых, то можно проверять этот аргумент в обработчике события KeyPress.

```
Private Sub Text1_KeyPress(KeyAscii As Integer)
    If KeyAscii <48 or KeyAscii>57 then
        Beep
        MsgBox "Данные должны быть числовые"
    End If
End Sub
```

В данном случае число 48 код цифры 0, а число 57 – код цифры 9 в кодовой таблице ASCII.

#### **Использование события KeyDown и KeyUp**

Обработчики этих событий могут использоваться для анализа нажатия функциональных клавиш, навигационных клавиш, клавиш редактирования. Коды этих клавиш не относятся к стандартным ASCII-кодам. Эти события перехватывают нажатие комбинации клавиш, вроде Ctrl + Del, Shift + Ctrl + Del и т.д.

#### **Использование события KeyPreview**

Если свойство формы KeyPreview установлено в True, то все события, связанные с клавиатурой, будут анализироваться сначала в обработчиках событий формы и лишь потом в обработчиках событий элементов управления, расположенных на данной форме. Это свойство позволяет проанализировать код нажатой клавиши прежде, чем включится в работу обработчик события активного элемента формы.

```
Private Sub Form1_Load()
    KeyPreview=True
'события формы будут генерироваться до событий
'элементов управления.
End Sub
Private Sub Form1_KeyDown(KeyCode As Integer, Shift As Integer)
    Select Case KeyCode
        ' анализируем код нажатой клавиши и сообщаем
        Case vbKeyF1
            MsgBox "нажата клавиша F1"
            [ операторы ]
        Case vbKeyF2
            MsgBox "нажата клавиша F2"
            [ операторы ]
        ...
        Case Else
    End Select
End Sub
```

VB имеет класс элементов управления Control, к которому относятся все элементы управления, размещенные на форме. Тип элемента управления анализируется с помощью функции TypeOf. Данный класс элементов управления в сочетании с функцией TypeOf можно успешно использовать для контроля правильности ввода данных на уровне формы. В примере П8.1 переменная ctl объявлена как переменная типа Control в разделе Общие формы.

**Пример П8.1.** Отключение элемента управления (кнопка cmdOK), пока не будут заполнены все поля ввода.

```
Dim ctl As Control
```

---

```
Private Sub Form_KeyUp(KeyCode As Integer, Shift As Integer)
    For Each ctl In Controls
        If TypeOf ctl Is TextBox Then
            If ctl.Text="" Then
                cmdOK.Enabled=False
            Exit Sub
            End If
        End If
    Next ctl
    cmdOk.Enabled = True
End Sub
```

В данном примере при нажатии любой клавиши с помощью цикла For Each просматриваются все элементы управления, установленные на форме, и проверяется состояние текстовых полей. Если хотя бы одно поле TextBox пусто, то свойству Enabled кнопки cmdOk присваивается значение False и осуществляется выход из процедуры. Когда все поля TextBox будут заполнены, то свойству Enabled кнопки cmdOk присваивается значение True и она становится доступной.

### 6.1.3. Реализация проверки данных на уровне полей формы

#### Использование свойств объектов

Контроль на уровне полей форм позволяет обнаруживать и устранять ошибки в момент ввода. Для этой цели можно использовать и некоторые свойства элемента TextBox: MaxLength, PasswordChar, Locked.

Свойство MaxLength задает максимальное количество символов, вводимых в текстовое поле. Когда пользователь пытается ввести в поле больше символов, чем указано в свойстве MaxLength, система подает звуковой сигнал.

Свойство PasswordChar позволяет скрывать вводимые символы, заменяя их звездочками. В качестве маскирующего можно использовать любой символ, но чаще применяют именно звездочки. Такой прием часто используется для скрытия паролей в диалоговых окнах регистрации, а также для ограничения доступа.

Свойство Locked запрещает редактирование поля ввода вводом данных с клавиатуры, но оставляет возможность программного изменения свойства Text.

#### Использование событий клавиатуры

Событие KeyPress распознает регистр нажатой клавиши, поэтому при контроле вводимых символов их целесообразно приводить к одному регистру и лишь потом анализировать.

```
Sub txtText1_KeyPress(KeyAscii As Integer)
    KeyAscii=Asc(VCase(Chr(KeyAscii)))
End Sub
```

Функция Chr преобразует код нажатой клавиши в символ, функция UCase переводит данный символ в верхний регистр, а функция Asc снова преобразует символ в числовой код.

Например, нажата клавиша с символом "а" - код 97. Код 97 преобразуется в символ "а" функцией Chr, функция UCase преобразует символ "а" в символ "А", функция Asc возвращает код 65.

События KeyDown и KeyUp распознают статус клавиатуры: первое событие генерируется в момент нажатия клавиши, а второе в момент отпускания. Кроме того, имеется возможность отслеживать нажатие функциональных клавиш Shift, Ctrl, Alt одновременно с нажатием символьной клавиши, например, Ctrl + С. Факт нажатия и отпускания клавиши возвращается через параметр KeyCode, а нажатие функциональной клавиши – через параметр Shift.

**Пример П6.2.** Контроль нажатой клавиши

```
Private Sub txtText1_KeyDown(KeyCode As Integer, Shift As Integer)
    Select Case Shift
        Case 1
            Print "Нажата клавиша Shift"
        Case 2
            Print "Нажата клавиша Ctrl"
        Case 3
            Print "Нажаты клавиши Shift и Ctrl"
        Case 4
            Print "Нажата клавиша Alt"
        Case 5
            Print "Нажаты клавиши Shift и Alt"
        Case 6
            Print "Нажаты клавиши Ctrl и Alt"
        Case 7
            Print "Нажата клавиша Ctrl, Shift и Alt"
    End Select
End Sub.
```

Следующий код контролирует нажатие клавиш "А" и Shift:

```
Private Sub txtText1_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode=65 And Shift=1 Then
        Print "Нажаты клавиши А и Shift"
    End If
End Sub.
```

#### 6.1.4. Проверка данных с использованием события Change

При небольшом числе полей ввода для контроля ввода данных можно использовать событие Change элемента управления TextBox. Например, если форма имеет два поля ввода txtText1 и txtText2, то код контроля будет иметь следующий вид:

```
Private Sub cmdOkEnabled_Click ( )
    If txtText1.Text <> " " And txtText2.Text <> " " Then
        cmdOk.Enabled=True
```

```

Else
    cmdOk.Enabled=False
End If
End Sub.
Private Sub txtText1_Change ( )
    Call cmdOkEnabled
End Sub
Private Sub txtText2_Change ( )
    Call cmdOkEnabled
End Sub

```

В данном примере процедура контроля cmdOkEnabled вызывается в обработчике события Change каждого поля ввода. Если хотя бы одно из полей ввода пустое, кнопка Ok остается недоступной.

### 6.1.5. Функции проверки данных

Visual Basic 6 имеет несколько функций, позволяющих проверять типы данных: IsNumeric, IsDate, IsNull.

Функция IsNumeric возвращает True, если аргумент является числовым и False в ином случае. Функция IsDate возвращает True, если в аргументе содержится допустимая дата. Эти функции можно применять совместно с функцией InputBox:

```

StrValue=InputBox("Введите число")
If IsNumeric (strValue) Then
    Print "Введено число"
Else
    Print "Некорректный ввод данных"
End If

```

**Пример П6.3.** Проверка ввода числовых данных на уровне полей формы.

Приведенная процедура обеспечивает контроль ввода данных в один элемент управления – окно ввода txtTextNumeral.

```

Public Sub txtTextNumeral(KeyAscii As Integer)
    Select Case KeyAscii
        Case Asc("0") To Asc("9"), Asc("."): 8
            ' если нажаты клавиши от 0 до 9, точка или клавиша BackSpace
            Numeral = Numeral + Chr(KeyAscii)
        Case Else
            ' если иначе, то код KeyAscii сбрасывается и никаких
            ' изменений не происходит
            KeyAscii = 0
            Beep
    End Select
End Sub

```

**Пример П6.4.** Проверка числовых данных на уровне формы. Контроль осуществляется после ввода данных.

```

Sub ControlVvodaNumber()
Dim ctl As Control, b As String
Dim FirstPoint As Boolean
For Each ctl In Controls
    If TypeOf ctl Is TextBox Then
        FirstPoint = False
        For l = 1 To Len(Trim(ctl.Text))
            Flag = False
            b = Mid(Trim(ctl.Text), l, 1)
            Select Case Asc(b)
                Case Asc("0") To Asc("9")
                    Case Asc(".") And FirstPoint = False
                        FirstPoint = True
                    Case Else
                        Beep
                        MsgBox "Ошибка ввода данных"
                        ctl.SetFocus
                        Flag = True
            End Select
        Next l
        ctl.SetFocus
    End If
Next ctl
End Sub

```

Данная процедура так же, как и в примере Пб.3, обеспечивает контроль правильности ввода в текстовые поля числовых данных. Отличие состоит в том, что в вводимом числе допускается только одна точка (разделитель целой и дробной части числа).

Программа проверяет, если тип элемента управления TextBox, тогда проверяется, является ли текущий символ цифрой или точкой. Не допускается в строке более одной точки

### Событие Validate.

Перед тем, как элемент управления потеряет фокус ввода, в нем возникает событие Validate – но только если свойство CausesValidation элемента управления, который вскоре получит фокус ввода, установлено как True. Событие Validate в сочетании со свойством CausesValidation позволяет проверять данные, когда пользователь пытается переключиться на другой элемент управления. Это событие предоставляет аргумент Cancel: установив его, как True, можно запретить переключение на другой элемент. Ниже приведен фрагмент программы использования свойства Validate.

```

Private Sub Text1_Validate(Cancel As Boolean)
    Cancel = False
    ControlVvodaNumber
    If Flag Then Cancel = True
End Sub

```

Объявите в разделе General переменную Flag типа Boolean. Если процедура контроля ввода ControlVvodaNumber обнаружит ошибку, то переменная Flag принимает значение True и курсор ввода остается в окне ввода Text1.

## 6.2. Обработка ошибок

### 6.2.1. Общие сведения об обработке ошибок

При разработке программы неизбежно возникновение ошибок. Ошибки могут быть вызваны невнимательностью при написании текста программы или из-за логических ошибок. Такие ошибки устраняются при отладке программы и ее тестировании. Но при использовании программы все равно могут возникать ошибки по вине пользователя, например, неправильно указано имя файла, не вставлен диск в дисковод и др., а также ошибки, не предусмотренные разработчиком. Такие ошибки принято называть ошибками периода выполнения.

В идеальном случае разработанная программа должна работать без ошибок, однако все возможные ситуации предусмотреть невозможно. В VB предусмотрены стандартные средства обработки ошибок. При обнаружении ошибок периода выполнения программа выводит на экран сообщение о коде ошибки и рекомендации по ее устранению. Однако после нажатия кнопки ОК в окне сообщения программа завершает работу. При этом обработчик события Unload игнорируется и пользователь теряет все не сохраненные данные. Чтобы при таких ситуациях программа не завершилась аварийно, необходимо в программу включать пользовательские обработчики ошибок. Обработчик ошибок перехватывает сообщение об ошибках периода выполнения и передает управление специальной процедуре обработки ошибок, которая, как правило, предлагает пользователю в режиме диалога устранить причину, вызвавшую ошибку.

Для обработки ошибок в VB6 предусмотрены специальные средства: операторы On Error и Resume, а также объект Err.

Оператор On Error активизирует режим обработки ошибок. Это означает, что при возникновении ошибки периода выполнения после ввода в программу этой строки выполнения предусмотренная в программе процедура обработки ошибок. Выполнение программы не прерывается и стандартное сообщение об ошибке не выводится.

Синтаксис оператора:

```
On Error {[GoTo <метка> \ Resume Next \ GoTo 0]}
```

Опция GoTo <метка> – передает управление на метку, <метка> должна находиться в той же процедуре, что и оператор On Error. Если в качестве метки указан 0, то предусмотренная разработчиком процедура обработки ошибок отключается и включается стандартный механизм обработки ошибок.

Опция Resume Next позволяет продолжить выполнение программы со следующего оператора, то есть ошибка в этом случае игнорируется.

Например:

```
On Error GoTo metka1 – управление передается на процедуру обработки ошибки по метке metka1;
```

```
On Error GoTo 0 – процедура обработки ошибки отключается;
```

```
On Error Resume Next – выполнение программы начинается со следующего оператора. Ошибка периода выполнения игнорируется.
```

Оператор Resume {[Label\Next]} – позволяет продолжить выполнение программы. Если параметры опущены, то управление передается на строку, вызвавшую ошибку, для ее повторного выполнения, опция Next возвращает управление на строку, следующую за строкой, вызвавшей ошибку, а опция Label передает управление на указанную метку. Объект Err содержит код ошибки и системное сообщение об ошибке, а также некоторую другую информацию.

Объект Err, как и все объекты, имеет несколько свойств и методов. Свойство Number возвращает код последней возникшей ошибки. Для совместимости с предыдущими версиями Visual Basic это свойство принимается по умолчанию, поэтому коды

```
ErrNum1=Err  
ErrNum2=Err.Number
```

будут давать одинаковые результаты.

Свойство Description выводит сообщение об ошибке.

Для очистки объекта Err используется метод Clear:

```
Err Clear
```

Если не очистить объект Err после возникновения ошибки, то последующие обработки будут давать сообщения об одной и той же ошибке.

## 6.2.2. Реализация локального обработчика ошибок

### Порядок обработки ошибок

Обработка ошибок производится в три этапа:

подготовка перехвата;

проверка и устранение ошибки;

продолжение выполнения программы.

Структура процедуры с обработчиком ошибок приведена на рис. Пб.1.

Первым шагом является расстановка "ловушек". Для этого предназначен оператор On Error. Оператор On Error должен располагаться в начале процедуры. Он не выполняет никаких действий, а только передает управление обработчику ошибок по указанной

```
Private Sub Primer ( )  
    'подготовка перехвата  
    On Error GoTo metka1  
    [ код программы ]  
    Exit Sub  
metka1:    ' обработчик ошибок  
    [ код обработчика ошибок ]  
    Resume  
    *   конец процедуры обработки  
ошибки
```

Рис. Пб.1. Структура процедуры с обработчиком ошибок

метке при их возникновении.

Проверку кода ошибки и устранение ошибок обеспечивают обработчики ошибок.

Продолжение выполнения программы осуществляется с помощью оператора Resume.

Обработчик ошибок должен располагаться в конце процедуры. Перед обработчиком ошибок вставляется оператор Exit Sub – выход из процедуры. Это необходимо для того, чтобы при нормальном выполнении программы обработчик ошибок не выполнялся.

## Реализация обработчика ошибок

Для реализации обработчика ошибок необходимо:

определить код ошибки;

определить действия, которые должны быть выполнены при соответствующей ошибке;

обеспечить продолжение выполнения программы после устранения ошибки.

Рассмотрим пример локального обработчика ошибок при открытии файла.

**Пример П6.5.** Локальный обработчик ошибок.

```
Private Sub cmdVvod_(Click)
    On Error GoTo OpenError
    ' Вызов процедуры ввода данных
    FileNameVvod: ' метка возврата при выходе из обработчика ошибок
    Nkanal=FreeFile ( )
    Name=Input "Введите имя файла"
    Open Name For Input As Nkanal
    Exit Sub
OpenError: ' метка процедуры обработчика ошибок
    Select Case Err.Number `проверяется номер ошибки
        Case 53 `ошибка файл не найден
            MsgBoxErr. Deskription
            ' выводится текст системного описания ошибки
            MsgBox "Введите правильно имя файла"
```

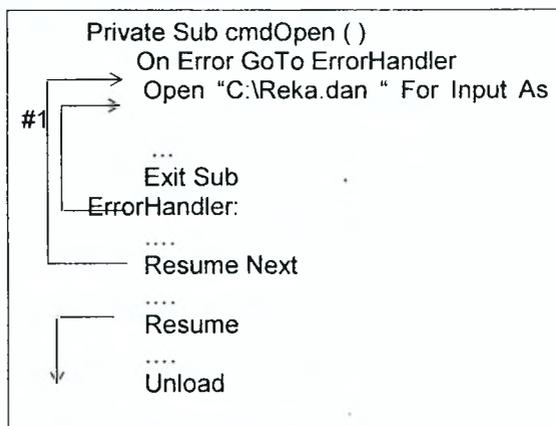


Рис. П6. 2. Передача управления оператором Resume

Case Else ` на случай появления непредвиденных ошибок

MsgBox "Неустановленная ошибка." & Err.Description

Exit Sub

End Select

Resume FileNameVvod

End Sub.

Алгоритм работы программы при обработке ошибок приведен на рис. Пб.2.  
Пример реализации обработчика ошибок ErrorHandler:  
Dim Answer As Integer

---

```
Error Handler:
    Answer = MsgBox("ошибка выполнения", vbAbortRetryIgnore)
    ' окно MsgBox содержит три кнопки: "Отказаться", _
    ' "Повторить" и "Игнорировать"
Select Case Answer
    Case vbAbort
        Unload Me ' отказаться, выйти из программы с сохранением _
        ' данных
    Case vbRetry
        Resume ' повторить операцию
    Case vbIgnore
        Resume Next ' игнорировать
End Select
```

### 6.2.3. Централизованная обработка ошибок

Для обеспечения надежной работы программы обработчик ошибок должен включаться в каждую процедуру. Некоторые процедуры, выполняющие сходные функции, могут иметь одинаковые обработчики ошибок. В целях сокращения числа обработчиков ошибок целесообразно создавать универсальные процедуры для выполнения той или иной операции и включать в нее обработчик ошибок, реагирующий на все возможные ситуации.

**Пример Пб.6.** Применение централизованного обработчика ошибок.

```
Private Sub cmdGetData_Click ( )
    Dim FileName As String
    FileName = InputBox ("Введите имя файла")
    If OpenFile(FileName)=True Then
        ' Open File (FileName) - функция, централизованный обработчик ошибок
        MsgBox "Файл успешно открыт"
    Else
        MsgBox "Файл не открыт. Проверьте пожалуйста имя файла"
    End If
End Sub
```

---

```
Function OpenFile (FileName As String) As Boolean
    ' активизируем локальный обработчик ошибок
    On Error GoTo OpenError
    ' открываем файл
    Open FileName For Input As # 1
    ' все в порядке, возвращаем True
    OpenFile=True
    ' выходим из функции, чтобы не попасть в локальный
    ' обработчик ошибок
    Exit Function
```

```

Open Error: ' локальный обработчик ошибок
  Select Case Err.Number
  Case 53
    ' Файл не найден
  Case 55
    ' Файл уже открыт
  Case Else
    ' неизвестная данной программе ошибка
  End Select
  ' Операция не удалась, возвращаем False
  OpenFile=False
End Function

```

В данном примере в обработчике ошибок не указаны конкретные действия, которые необходимо выполнить в той или иной ситуации, а даны лишь комментарии.

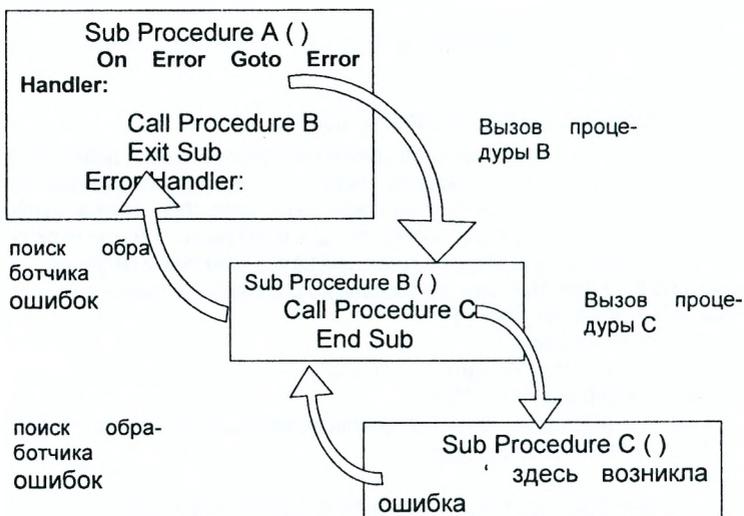


Рис. П6.3. Обработка ошибок при вложенных процедурах

### Обработка ошибок при вложенных вызовах процедур

При возникновении ошибки периода выполнения программа ищет обработчик ошибок в текущей процедуре, если в ней обработчик ошибок отсутствует, то программа ищет обработчик ошибок в процедуре, вызвавшей текущую процедуру, и т.д. до главной процедуры. Если и в главной процедуре нет обработчика ошибок, то программа вызывает стандартный обработчик ошибок и выводит сообщение о коде ошибки и системную информацию об ошибке. Алгоритм работы программы при обработке ошибок вложенных процедур приведен на рис. П6.3.

## Обработка ошибок объекта `CommonDialog`

Типичным примером использования обработчика ошибок может быть обработка свойства `CancelError` элемента управления `CommonDialog`. Если значение свойства `CancelError` равно `True`, то при щелчке на кнопке `Cancel` в одном из диалоговых окон элемента управления `CommonDialog` возникнет ошибка. Если значение этого свойства равно `False`, ошибка не возникает, но различить нажатие кнопок `OK` и `Cancel` в диалоговых окнах будет затруднительно.

```
Function OpenFile() As String
    On Error GoTo Cancel
    CommonDialog1.CancelError = True
    CommonDialog1.ShowOpen
    OpenFile = CommonDialog1.FileName
    Exit Function
```

Cancel:

```
If Err.Number = cdlCancel Then
    OpenFile = ""
    MsgBox Err.Description
    Exit Function
Else
    MsgBox Err.Description
    Stop
End If
```

```
End Function
```

Функция работает следующим образом. Если программа нашла имя файла, введенное в строке ввода диалогового окна открытия файла, то ошибки не возникает. Если при поиске файла возникнет ошибка, то вызывается обработчик ошибки по метке `Cancel`. Если пользователь нажал кнопку `Cancel` окна сообщения, тогда код, возвращаемый объектом `Err`, будет равен коду нажатой клавиши. В этом случае функция `OpenFile` получает значение "пусто" и процедура функция прекращает работу. В ином случае происходит выход из программы по команде `Stop`.

## Приложение 7

### Отладка программ на языке `Visual Basic`

При разработке программ могут возникать ошибки разного рода, допущенные по невнимательности или незнанию. Все ошибки можно разделить на три группы: синтаксические, периода выполнения и логические.

Синтаксические ошибки в исходном тексте программы возникают при неправильном наборе операторов языка `VB6`, пропуске символов при написании операторов или применении неправильных конструкций операторов. Такие ошибки могут быть выявлены и устранены до запуска программы на выполнение. Ошибки, связанные с не объявлением и не совпадением типов переменных, выявляются при первом запуске программы.

Ошибки периода выполнения возникают в процессе работы, когда программа пытается выполнить недопустимую или неподготовленную операцию. Такие ошибки возникают чаще всего из-за невнимательности оператора. Способы предупреждения ошибок периода выполнения рассмотрены в Приложении 6.

Логические ошибки обнаруживаются тогда, когда пользователь замечает, что программа выдает не тот результат, который ожидался. Такие ошибки обнаруживаются труднее всего. Для выявления логических ошибок необходимо подготавливать наборы тестовых данных и проверять программу с помощью этих тестов. С целью уменьшения ошибок при написании текста программы и ускорения поиска ошибок при отладке программ в системе программирования VB6 предусмотрен ряд средств.

### Средства уменьшения ошибок при написании программ

Для уменьшения ошибок пользователя при написании программ в редакторе VB6 предусмотрено ряд средств, которые могут подключаться пользователем. Эти средства

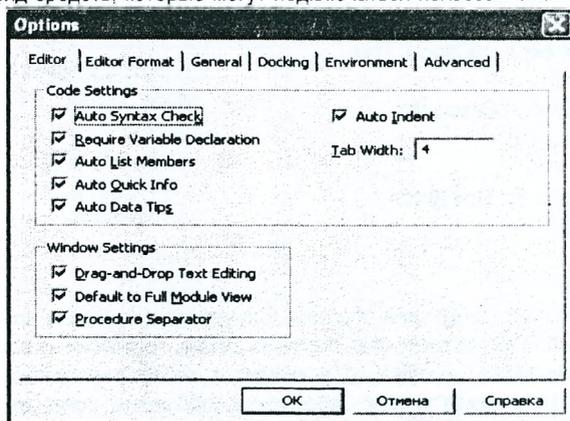


Рис. П7.1. Окно диалога Options

собраны на вкладке Editor (Редактор) окна диалога Options (Опции) меню Tools (Инструменты) (рис.П7.1). К таким средствам относятся:

- Auto Syntax Check - автоматическая проверка синтаксиса. После завершения написания программной строки и при переходе на новую строку или нажатии клавиши Ввод программа проверяет синтаксис и при обнаружении ошибки в программной строке выдает предупреждение. Строка, содержащая ошибку, подчеркивается волнистой линией красного цвета;

- Require Variable Declaration - контроль объявления переменных. При установке флажка у данного параметра в раздел General каждой вновь загружаемой формы автоматически включается оператор Option Explicit. В этом случае при запуске программы автоматически проверяется объявление переменных. Если какая-то переменная не объявлена, выдается сообщение об ошибке. Объявление всех переменных позволяет ускорить поиск и устранение ошибок, связанных с неправильным написанием имен переменных;

- Auto List Members - список свойств объектов. Если после написания имени объекта ввести символ "точка", то открывается список свойств данного объекта, из которого можно выбрать нужное свойство двойным щелчком мыши. Аналогично, если для свой-

ства объекта зарезервированы значения свойств, то при вводе символа "=" открывается список этих свойств, при объявлении типов переменных после ввода служебного слова As и пробела на экран выводится список типов переменных. При работе со списком имеется возможность быстрого поиска нужного свойства. Введите первый символ свойства – список сортируется и на экране появляются свойства, имя которых начинается с введенного символа. При вводе очередного символа снова производится сортировка списка и т. д. При небольших списках свойств достаточно ввести два первых символа требуемого свойства, чтобы найти его в списке;

- Auto Quick Info - быстрый поиск информации. При написании функций после ввода символа "(" на экран выводится подсказка о синтаксисе данной функции;

- Auto Data Tips - отображение типов переменных и их значение. При установке данного флажка при выделении переменной мышью появляется подсказка о типе переменной или ее значении. Это позволяет проконтролировать значения переменных при прерывании программы вследствие возникновения ошибки или в точке останова.

Флажок Auto Indent позволяет автоматически формировать отступы при форматировании текста программы, шаг табуляции устанавливается в окне Tab Width.

## Средства отладки программы

### Панель инструментов отладки Debug

Средства отладки программы собраны на панели инструментов Debug (рис. П 7.2). Панель вызывается с помощью одноименной команды главного меню Debug.

Кнопка Start (Старт) служит для запуска программы.

Кнопка Pause (Пауза) приостанавливает выполнение программы. В этом режиме имеется возможность просмотра значений переменных путем зависания мыши над теми переменными, значения которых вас интересуют, а также редактирования программы. После этого можно продолжить выполнение программы с помощью кнопки Старт.

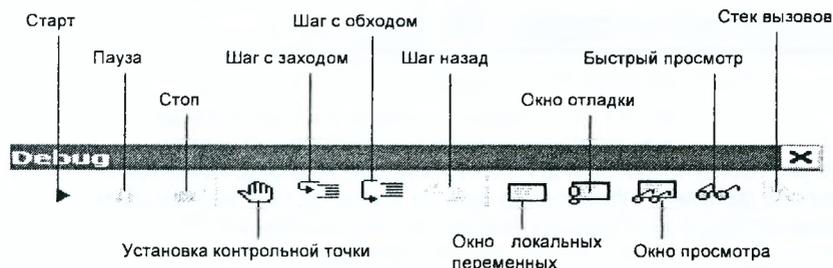


Рис. П7.2. Панель инструментов отладки (Debug)

Кнопка Break (Стоп) прерывает работу программы.

Кнопка Toggle Breakpoint (Установка контрольной точки) позволяет установить или снять контрольную точку. При достижении контрольной точки программа останавливается. После остановки программы можно просмотреть значения переменных. Контрольные точки можно устанавливать на любую строку программы, кроме комментариев и меток.

Кнопки Шаг с заходом, Шаг с обходом и Шаг назад служат для установки режима пошагового выполнения программы.

Кнопка Step Into (Шаг с заходом) выполняет одну строку программы. Для продолжения работы нажмите клавишу F8. Пошаговое выполнение программы позволяет проанализировать ход работы программы и устранить обнаруженные логические ошибки.

Кнопка Step Over (Шаг с обходом) позволяет вызвать процедуру или функцию без пошагового выполнения каждой строки. Это ускоряет отладку программы, когда в программе используются готовые, заранее отлаженные процедуры. Для пошагового выполнения программы с обходом можно использовать комбинацию клавиш Shift+F8.

После отладки программы следует снять все точки прерывания в проекте командой Debug/Clear All Breakpoints.

### Окна просмотра

Окна просмотра позволяют контролировать значения переменных.

Кнопка Locals Window (Окно локальных переменных) (рис. П7.3) отображает окно локальных переменных и автоматически выводит в нем значения всех переменных, попадающих в область видимости текущей процедуры. Значения в этом окне изменяются автоматически при каждом переходе из

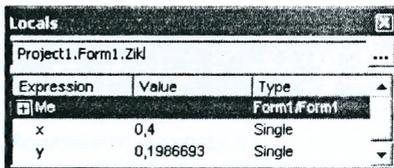
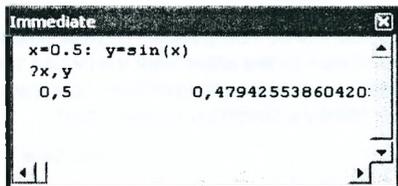


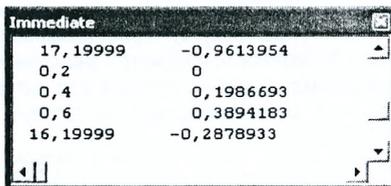
Рис. П7.3 Окно локальных переменных

режима выполнения программы в режим прерывания и обратно или при активизации окна Стек (Call Stack).

Кнопка Immediate Window (Окно непосредственных вычислений) отображает окно непосредственных вычислений (рис. П7.4). В этом окне можно проверить работу функций и состояние переменных, как во время проектирования, так и во время отладки.



а)



б)

Рис. П7.4. Использование окна непосредственных вычислений:  
а) во время проектирования; б) во время отладки программы

На рис. П7.4 а) показан пример использования окна непосредственного вычисления при разработке программы табулирования функции. В окно введено значение аргумента  $x$ , выражение для функции и результаты выведены в окно оператором Print (символ "?").

На рис.П7.4 б) приведен пример использования окна непосредственных вычислений при отладке программы. Для этого в нужном месте программы необходимо вставить оператор Debug.Print x,y.

Во время выполнения программы можно в реальном времени следить за состоянием интересных вас переменных. Это можно делать с помощью окна непосредственных вычислений или с помощью специального окна Watch Window – окна просмотра (рис.П7.5), которое вызывается кнопкой View Watch Window. При этом в окне можно отображать состояние не только переменных, но и объектов.

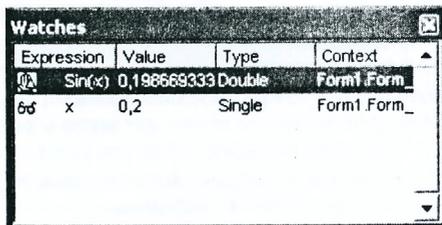


Рис.П7.5. Окно просмотра

Для настройки окна просмотра необходимо вызвать окно диалога Add Watch командой Add Watch или Edit Watch из меню Debug (рис.П7.6). В строке Expression указать имя контролируемой переменной, выражение или имя объекта. Указать в соответствующих окнах ввода имя процедуры, формы, а также установить один из переключателей в группе Watch Type (Тип просмотра): просмотр выражения, остановка при достижении заданного значения или остановка при изменении значения.

Окно Watches появляется при каждом прерывании программы.

Кнопка Quick Watch – быстрый просмотр выводит на экран окно диалога с текущим значением выделенного выражения (рис.П7.7). Кнопка Add позволяет добавить это выражение в окно просмотра Watch, чтобы можно было просмотреть значение этого выражения в будущем.

Окно Вызовов стека вызывается кнопкой Call Stack во время прерывания выполнения программы. Оно содержит список всех вызванных, но не завершенных процедур. В примере на рис.П7.8 этот список содержит две процедуры.

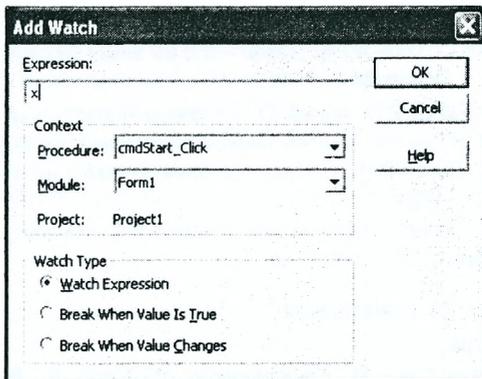


Рис.П7.6. Окно диалога для настройки окна просмотра

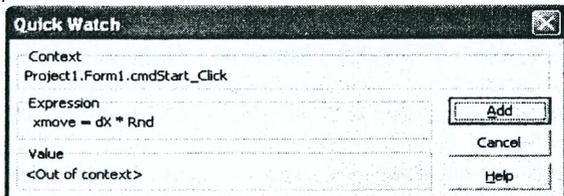


Рис. П7.7. Окно быстрого просмотра

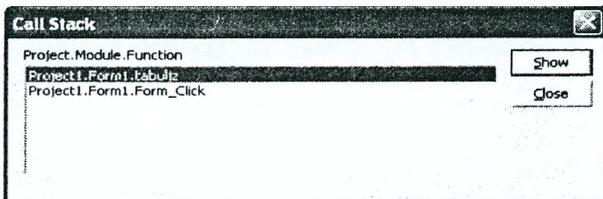


Рис. П7.8. Окно Вызовов стека

### Команды меню отладки (Debug)

Команда меню Debug Run to Cursor позволяет в режиме прерывания выполнить программу от текущего положения до того места, где установлен курсор.

Команда Set Next Statement (Новый оператор) задает следующий оператор, подлежащий выполнению. При этом весь текст программы между текущей точкой остановки и новым положением курсора пропускается. Эта команда позволяет также вернуться к предыдущим частям программы и проверить их работу при других значениях свойств или переменных.

Команда Show Next Statement (Показать новый оператор) перемещает курсор ввода в оператор, который будет выполняться следующим. Для использования данной команды переведите программу в режим прерывания и введите команду, чтобы увидеть следующий выполняемый оператор.

# Литература

## Основная

1. Быков В.Л., Ашаев Ю. П. Основы информатики. – Брест: БГТУ, 2005. – 488 с.
2. Быков В.Л. Основы информатики: Конспект лекций – Брест: БГТУ, 2003. – 257 с.
3. Быков В. Л. Основы программирования на языке Visual Basic 6.0. – Брест, БГТУ, 2002. – 230 с.
4. Основы информатики: Учебное пособие/под ред. А. Н. Морозевича. – Мн.: Новое знание, 2001. – 544 с.: ил.

## Дополнительная

### К разделу 1

- Алексеев А. П., Информатика, - М.: "Солон-Р", 2002. – 400 с., ил.
- Информатика/ под ред. Н. В. Макаровой. – М.: Финансы и статистика, 2000. – 768 с.: ил.
- Информатика. Базовый курс / Симонович С. В. и др. – СПб: Питер, 2000. – 640 с.: ил.
- Использование Microsoft Windows XP Home Edition. Специальное издание.: - М.: Издательский дом "Вильямс", 2002. – 896 с.: ил.
- Коуров Л. В. Словарь-Справочник по информатике: - Мн.: Амалфея, 2000. – 176 с.
- Леонтьев В. П. Новейшая энциклопедия персонального компьютера 2000. – 2-евро-издание, перераб. и доп. – М.: ОЛМА-ПРЕСС, 2000. – 847 с.:ил.
- Ляхович В. Ф. Основы информатики. - Ростов н/Д.: Изд-во "Феникс", 1996. –640 с.
- Новейший самоучитель работы на компьютере. - Москва: издательство "ДЕСС КОМ", 2000. - 654 с.
- Острейковский В. А. Информатика. – М.: "Высшая школа", 2000.-511 с.: ил.
- Фигурнов В. Ф. IBM PC для пользователя. - М.: Финансы и статистика, 2003.
- Экономическая информатика / под ред. П. В. Конюховского и Д. Н. Колесова. – СПб.: Питер, 2001. – 560 с.: ил.

### К разделу 2

- Математический энциклопедический словарь/ Гл. ред. Ю.В. Прохоров; ред. кол.: С. И. Адян, Н.С. Бахвалов, В.И. Ботюцков, А.П. Ершов, Л.Д. Кудрявцев, А.Л. Онищик, А.П. Юшкевич. – М.: Сов. Энциклопедия, 1988. – 847 с.: ил.
- Гурский Д.А. Вычисления в MathCad – Мн.: Новое знание, 2003. – 814 с.
- Mathcad 6.0 PLUS. Финансовые, инженерные и научные расчеты в среде Windows 95 /Перевод с англ. – М.: Информационно – издательский дом «Филинь», 1996. – 712 с.
- Численные методы решения задач строительной механики: /Справ. пособие / В.П. Ильин, В.В. Карпов, А.М. Масленников/Под общ. ред. В.П. Ильина, - Мн.: Выш. шк., 1990. -349 с.
- Кулаичев А.П. Методы и средства анализа данных в среде Windows. Stadia6.0. Изд. 2-е. перераб. и доп. - М.: Информатика и компьютеры, 1998. - 270 с.
- Кирьянов Д. В. Самоучитель MathCad 11. – СПб.: БХВ-Петербург, 2003.- 560 с.: ил.
- Рыжиков Ю. И. Решение научно-технических задач на персональном компьютере. – СПб.: КОРОНА принт, 2000. – 272 с.

### К разделу 3

- Брайн Сайлер и Джефф Скоттс. Использование VB 6.0 - М.: СПб.; К.: Издательский дом "Вильямс", 2001.- 832 с.: ил.

- Браун С. Visual Basic 6: учебный курс. – СПб.: Питер, 2002. – 576 с. ил.
- Быков В. Л. Основы программирования на языке VB 6.0. – Брест.: БГТУ, 2002. - 230 с.
- Волчёнков Н. Г. Программирование на Visual Basic 6. – М.: ИНФРА, 2000.
- Гарри Корнель . Программирование в среде VB5: - Мн.:ООО "Папури", 1998.-608 с.:ил.
- Дьяконов В. П. Справочник по алгоритмам и программам на языке БЕЙСИК для персональных ЭВМ: Справочник. - М.: Наука. Гл. ред. физ.-мат. лит., 1987. - 240 с.
- Светозарова Г. И. и др. Практикум по программированию на языке БЕЙСИК. - М.: Наука, 1988.
- Михаэль Рейтингу, Геральд Муч. Visual Basic 6.0-К.: Издательская группа BHV, 2000.- 288 с.: ил.
- Microsoft Corporation. разработка приложений на VB 6.0 - М.: Издательско-торговый дом "Русская Редакция", 2000. - 400 с.: ил.
- Турчак Л. И. Основы численных методов. – М.: Наука. Гл. ред. физ.-мат. лит., 1987. – 320 с.
- К разделу 4**
- См. п. п. 3,4, 9, 12
- К разделу 5**
- См. раздел 1 п. п. 9, 10, 14
- Гарнаев А.Ю. Использование MS EXCEL и VBA в экономике и финансах. – СПб.: БХВ – Санкт- Петербург, 1999. – 336 с.
- Гельман В. Я. Решение математических задач средствами Excel: Практикум. – СПб.: Питер, 2003. – 240 с.: ил.
- Орвис Вильям. Excel для ученых, инженеров и студентов: Пер с англ. – К.: Юниор, 1999. – 528 с.: ил.
- К разделу 6**
- См. раздел 1 п. п. 8, 9, 10, 11, 14
- Нозль Истабрук. Internet. Освой самостоятельно за 24 часа / Пер. с англ. – М.: ЗАО «Издательство БИНОМ», 1998 -320 с.: ил.
- Холмогоров В. Основы Web–мастерства. Учебный курс. – СПб.: Питер, 2002. - 352 с.: ил.

Учебное издание

**Быков Вячеслав Леонидович**  
**Ашаев Юрий Павлович**

# ОСНОВЫ ИНФОРМАТИКИ Практикум

*Рекомендовано советом Брестского государственного  
технического университета в качестве пособия для студентов  
технических специальностей*

ISBN 985-493-048-3



Ответственный за выпуск **Быков В.Л.**  
Редактор **Строкач Т.В.**  
Компьютерная верстка **Боровикова Е.А.**  
Корректор **Никитчик Е.В.**

Лицензия № 02330/0133017 от 30.04.2004 г.  
Подписано в печать 31.10.2006 г. Бумага «Снегурочка».  
Формат 60x84 1/16. Гарнитура Arial Narrow. Усл. печ. л. 18,4.  
Уч.-изд. л. 19,75. Зак. № 1024. Тираж 200 экз.  
Отпечатано на ризографе учреждения образования  
«Брестский государственный технический университет».  
Лицензия № 02330/0148711 от 30.04. 2004 г.  
224017, г. Брест, ул. Московская, 267.