

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

КАФЕДРА ИНФОРМАТИКИ И ПРИКЛАДНОЙ МАТЕМАТИКИ

# ЛАБОРАТОРНЫЙ ПРАКТИКУМ

по дисциплине «Информатика»

для студентов специальности

«Машины и аппараты пищевых производств»

машиностроительного факультета

дневной формы обучения

*2-е издание, исправленное и переработанное*



Группа \_\_\_\_\_

Фамилия \_\_\_\_\_

Имя \_\_\_\_\_

Отчество \_\_\_\_\_

м/тел. +375 (\_\_\_\_) \_\_\_\_-\_\_\_\_-\_\_\_\_

Вариант № \_\_\_\_\_

Брест 2019

УДК 004

Практикум предназначен для студентов специальности 1 – 36 09 01 «Машины и аппараты пищевых производств» I курса машиностроительного факультета, изучающих дисциплину «Информатика». В него входят задания для лабораторных работ и задания для их защиты. В теоретической части изложены методические рекомендации по работе с текстовым редактором MS WORD, электронными таблицами MS EXCEL, системой компьютерной алгебры MathCAD и системой программирования Visual Basic for Application (VBA).

2-е издание, исправленное и переработанное.

Составители: Гучко И.М., ст. преподаватель  
Кофанов В.А., к.т.н., доцент  
Рамская Л.К., ст. преподаватель

Рецензент: доцент кафедры прикладной математики и информатики  
БрГУ имени А.С. Пушкина, к.ф.-м.н. Матысик О.В.

## Общие указания

Практикум предназначен для организации самостоятельной и лабораторной работы студентов при изучении дисциплины «Информатика».

Перед началом работы необходимо привести информацию на титульном листе, то есть должны быть указаны данные ее владельца.

В специальном блоке «Условие» на странице с лабораторной работой переписывается условие задачи согласно варианта из электронного файла задания. При выполнении лабораторных работ результаты вычислений заносятся в отчет в том же виде, в котором они отображаются на экране монитора. Ведение записей выполняется четко и разборчиво шариковой ручкой. Неправильные (ошибочные) записи на страницах практикума необходимо исправлять с использованием корректирующих средств.

Каждая лабораторная работа считается выполненной только при наличии отметки преподавателя о ее защите, подтвержденной его подписью (личной печатью) на данной странице. Данные из этого листа служат основанием для допуска к зачету по дисциплине.

### Отметки о защите лабораторных работ

<i>Наименование работы</i>	<i>Дата</i>	<i>Защита</i>	<i>Примечание</i>
Лабораторная работа № 1			
Лабораторная работа № 2 (1)			
Лабораторная работа № 2 (2)			
Лабораторная работа № 3 (1)			
Лабораторная работа № 3 (2)			
Лабораторная работа № 4			
Лабораторная работа № 5			
Лабораторная работа № 6			

Допуск к экзамену:

\_\_\_\_\_ (дата)

\_\_\_\_\_ (подпись)

# Методические указания к выполнению лабораторных работ

## ЭЛЕКТРОННАЯ ТАБЛИЦА MICROSOFT EXCEL

Каждая рабочая книга электронной таблицы Excel состоит из одного или нескольких рабочих листов, а рабочий лист, в свою очередь, состоит из строк и столбцов (рис. 1).

На пересечении строки и столбца расположена отдельная ячейка. В каждой ячейке может храниться число, формула или текст. В любой момент времени только одна ячейка может быть активной. Активная ячейка выделяется темным контуром. Ее адрес, т.е. буква столбца и номер строки, указывается в поле **Имя**.

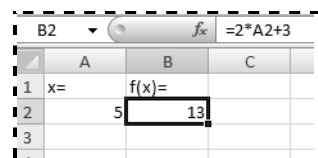


Рисунок 1 – Фрагмент рабочего листа Excel

### Соккрытие и отображение строк и столбцов

Excel позволяет скрывать строки и столбцы. Это может понадобиться в тех случаях, если необходимо скрыть от пользователя какую-либо часть информации, либо исключить часть информации при печати итогового отчета.

Чтобы скрыть строки (столбцы), выделите их и щелкните правой кнопкой мыши на заголовках этих строк (столбцов). В открывшемся контекстном меню выберите команду **Скрыть** (рис. 2). Также можно воспользоваться после выделения строк (столбцов) командой Excel версии 2007+: **Главная/Ячейки/Формат/Скрыть** или **отобразить/Скрыть строки (Скрыть столбцы)**.

Для возвращения на экран скрытой строки или столбца выделите вначале строки или столбцы, прилегающие к скрытой строке или столбцу (выделите хотя бы по одной строке или столбцу по обе стороны), затем щелкните правой кнопкой мыши на заголовках этих строк или столбцов и в открывшемся контекстном меню выберите команду **Отобразить** (рис. 2).

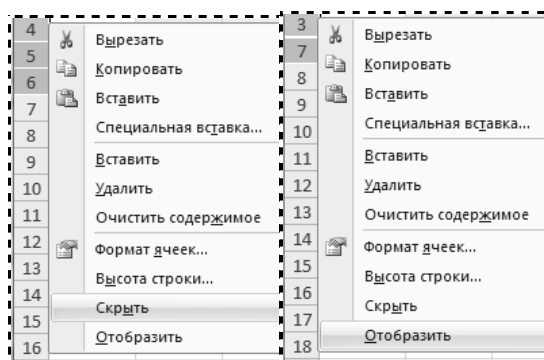



Рисунок 2 – Фрагменты контекстного меню заголовков строк

### Форматирование числовых значений

После ввода числовые значения, как правило, никак не отформатированы. Другими словами, они просто состоят из последовательности цифр. Некоторые операции форматирования программа Excel способна выполнять автоматически. Например, если вы вводите в ячейку число 12.2 (с точкой, а в настройках по умолчанию разделитель дробной и целой части – запятая), то программа автоматически применяет формат **Дата**.

Изменить числовой формат ячейки можно в диалоговом окне **Формат ячеек** (2003: **Формат/Ячейки**; 2007+: **Главная/Число/Пиктограмма в правом нижнем углу группы** ) , закладка **Число** (рис. 3). Если выбрать один из предложенных на этой закладке числовых форматов, то он сразу будет применен к активной ячейке. Если предварительно выделено несколько ячеек, то числовой формат будет применен ко всем выделенным ячейкам.

На вкладке **Число** диалогового окна **Формат ячеек** предусмотрено двенадцать категорий числовых форматов. При выборе соответствующей категории из списка **Числовые форматы** правая сторона панели изменяется так, чтобы отобразить соответствующие опции. Ниже в таблице 1 приведен список часто используемых в лабораторных работах категорий числовых форматов.

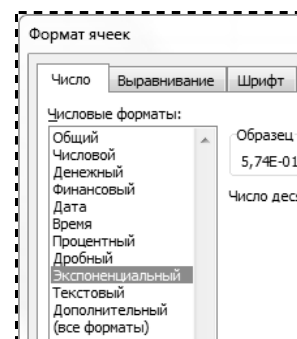


Рисунок 3 – Фрагмент диалогового окна **Формат ячеек**

Таблица 1 – Описание категорий числовых форматов

Категория формата	Описание категории
Общий	Формат, принятый по умолчанию. При его выборе числа отображаются в виде целых чисел, десятичных дробей или, если число слишком большое и не помещается в ячейке, в экспоненциальном формате.
Числовой	Позволяет указать число десятичных знаков и определить разделитель групп разрядов.
Дата	Позволяет выбрать один из форматов отображения дат.
Экспоненциальный	В этом формате число всегда отображается с буквой E. Число 2,35E+02 в экспоненциальном формате тоже, что и 235 в числовом.
Текстовый	Его применение к числовому значению заставляет Excel рассматривать это число как текст (даже если текст выглядит как число). По умолчанию текст выравнивается по

левому краю ячейки.

Важно понимать, что применение числового формата к ячейке никоим образом не изменяет само число, которое там находится. Форматирование изменяет только внешний вид отображаемого числового значения. Например, если в ячейке находится число 0,5742, его можно отформатировать так, чтобы на экране оно выглядело как 57%. Но если на ячейку сделана ссылка в формуле, то во время вычислений будет использоваться полное числовое значение 0,5742, а не отображаемое 57%.

### Относительные, абсолютные и смешанные ссылки

Практически во всех формулах в Excel используются ссылки на ячейки или диапазоны ячеек. Эти ссылки позволяют формулам работать с данными, содержащимися в этих ячейках и диапазонах, а не просто использовать фиксированные значения.

В формулах используются три типа ссылок на ячейки и диапазоны.

- **Относительные ссылки.** При копировании формул эти ссылки автоматически изменяются в соответствии с новым положением формулы.
- **Абсолютные ссылки.** Эти ссылки не изменяются при копировании формул.
- **Смешанные ссылки.** В этих ссылках номер строки (или столбца) является абсолютным, а столбца (строки) – относительным.

Отличительной особенностью абсолютных ссылок являются два знака доллара «\$»: один перед буквой столбца и второй перед номером строки (например, \$A\$5). В Excel также допускаются смешанные ссылки, в которых только одна часть адреса является абсолютной (например, \$A4 или A\$4).

По умолчанию Excel создает в формулах относительные ссылки (без знака \$). Различие между разными типами ссылок проявляется при копировании формул.

На рисунке 4 показан процесс табулирования функции  $y(x) = x^2$ . Для получения очередного значения  $x$  (начиная с ячейки A8) необходимо к предыдущему значению  $x$  прибавить значение шага табулирования  $h$  (ячейка B4). Формула в ячейке A8 будет выглядеть следующим образом: =A7+\$B\$4. Обратите внимание на то, что ссылка на ячейку A7 является относительной, а ссылка на ячейку B4 – абсолютной.

При копировании формулы из ячейки A8 в ячейку A9 ссылка A7 изменится, а ссылка B4 нет. В результате в ячейке A9 получим правильную формулу – =A8+\$B\$4. В случае если обе ссылки в формуле ячейки A8 были бы относительными (либо абсолютными), то формула в ячейке A9 была бы неверной (=A8+B5 или =\$A\$7+\$B\$4).

При копировании формулы по вертикали в относительных ссылках меняется номер строки, по горизонтали – имя столбца. В рассмотренном примере формула копируется вниз, поэтому вместо абсолютной ссылки на ячейку B4 (\$B\$4) можно использовать смешанную ссылку, т.е. B\$4.

Абсолютные или смешанные ссылки можно ввести вручную, вставив в нужных местах знаки доллара. Можно также воспользоваться клавишей <F4>, которая является удобным клавиатурным эквивалентом для этой операции. При вводе ссылки в ячейку – либо вручную, либо путем указания – нужно нажать клавишу <F4> несколько раз, чтобы программа «прокрутила» по циклу все четыре варианта ссылок.

	A	B	C
1	a=	0	
2	b=	2	
3	n=	10	
4	h=	0,2	
5			
6	x	y=x^2	
7	0		
8	0,2		
9	0,4		
10	0,6		

Рисунок 4 – Фрагмент процесса табулирования функции

### Именованные диапазоны ячеек в Excel

Назначив имя ячейке или диапазону ячеек, можно вставлять эти имена в формулы. Адресация по имени абсолютна. Существуют обязательные требования к наименованию группы ячеек: в нём не должно быть пробелов; оно обязательно должно начинаться с буквы; его длина не должна быть больше 255 символов; оно не должно быть представлено координатами вида A1 или R1C1; в книге не должно быть одинаковых имен.

Чтобы присвоить имя, можно использовать два способа:

**1 способ:** выделите ячейку или диапазон ячеек, перейдите в поле **Имя** и наберите имя, которое вы хотите использовать и нажмите клавишу <Enter> (рис. 5). Кликнув на стрелочку в правой части поля **Имя**, вы увидите список всех имен в текущей книге. При выборе имени в данном списке Excel автоматически выделяет соответствующую ячейку.

**2 способ:**

- выделите ячейку или диапазон ячеек;
- выберите команду (2003: **Вставка/Имя/Присвоить**; 2007+: **Формулы/Определенные имена/Вставить имя**);
- в появившемся диалоговом окне **Создание имени** (рис. 6) введите нужное имя.

Вид топлива	Цена за 1 л., руб.
Бензин А92	1,17
Бензин А95	1,25
Диз. топливо	1,29
Газ	0,66

Рисунок 5 – Ввод имени диапазона в поле **Имя**

Создание имени

Имя: Номер\_машины

Область: Книга

Примечание:

Диапазон: =Рассчетная таблица!\$C\$5:\$C\$14

Рисунок 6 – Диалоговое окно **Создание имени**


## Копирование (тиражирование) и перемещение (вырезание и вставка)

Копирование и перемещение содержимого ячейки (ячеек) – очень распространенная операция в программах электронных таблиц. Между копированием и перемещением существует различие: при копировании исходный диапазон не изменяется, а при перемещении – удаляется.

У операций копирования, вырезания и вставки существуют клавиатурные эквиваленты.

- <Ctrl+C>. Копирует выбранные ячейки в буферы обмена Windows и Office.
- <Ctrl+X>. Вырезает выбранные ячейки в буферы обмена Windows и Office.
- <Ctrl+V>. Вставляет содержимое буфера обмена в выбранную ячейку или диапазон ячеек.

Excel позволяет копировать и перемещать содержимое ячейки или диапазона ячеек путем перетаскивания. Выделите ячейку (или диапазон ячеек), которую нужно скопировать, а затем переместите указатель мыши к одной из ее (или его) четырех границ. Когда указатель примет вид стрелки, нажмите клавишу <Ctrl>. К указателю добавится маленький знак «плюс». После этого перетащите выбранные ячейки на новое место, продолжая удерживать нажатой клавишу <Ctrl>. Исходное множество ячеек останется на прежнем месте, а когда вы отпустите кнопку мыши, Excel создаст новую копию данных. Для перемещения данных повторите все описанные действия, но не нажимайте клавишу <Ctrl>.

Кроме того, можно скопировать формулы в соседние ячейки при помощи маркера заполнения  (процесс тиражирования). Маркер заполнения – небольшой черный квадрат в правом нижнем углу выделенного блока. При наведении на маркер заполнения указатель принимает вид черного креста. Удерживая левую клавишу мыши и передвигая маркер заполнения по строке или столбцу, происходит копирование содержимого исходной ячейки (ячеек) в выделенный диапазон.

Следует запомнить правило: при вырезании и вставке формулы (т.е. при перемещении ее в другую ячейку) относительные ссылки в формуле не изменяются, а при копировании – изменяются.

Не всегда требуется полное копирование исходного диапазона в диапазон назначения. Например, при копировании ячеек, содержащих формулы, часто требуется скопировать вычисленные значения, а не сами формулы. Для выполнения подобных операций используется диалоговое окно **Специальная вставка** (рис. 7), которое открывается с помощью команды (2003: **Правка/Специальная вставка**; 2007+: **Главная/Буфер обмена/Вставить/Специальная вставка**). Это окно можно также открыть, щелкнув правой кнопкой мыши на ячейке, куда будут вставляться скопированные данные, и выбрав из контекстного меню команду **Специальная вставка**. В диалоговом окне **Специальная вставка** расположено несколько переключателей, описание основных из них приведем ниже.

• *Все*. Из буфера обмена Windows копируется содержимое ячейки, форматы и параметры функции проверки допустимости данных.

• *Формулы*. Копируются только формулы и значения (без атрибутов форматирования), содержащиеся в исходном диапазоне.

• *Значения*. Эта опция позволяет скопировать только значения, вычисленные по формулам, причем диапазоном назначения может быть как новый диапазон, так и исходный. В последнем случае первоначальные формулы будут заменены полученными значениями.

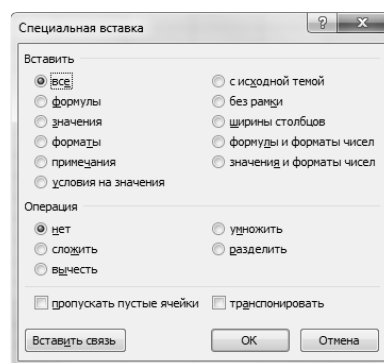


Рисунок 7 – Диалоговое окно **Специальная вставка**

## Встроенные функции Excel

Excel содержит огромное количество встроенных функций (основные из них приведены в соответствующем разделе далее), которые вы можете использовать в своих формулах. Ввести формулу, содержащую функции, можно вручную после знака равенства «=», либо с помощью диалогового окна **Мастер функций** (рис. 8). Открыть это окно можно одним из следующих способов:

1 способ: выберите команду (2003: **Вставка/Функция**; 2007+: **Формулы/Библиотека функций/Вставить функцию**);

2 способ: щелкните на кнопке **Вставить функцию**, которая находится в левой части строки формул;

3 способ: нажмите комбинацию клавиш <Shift+F3>.

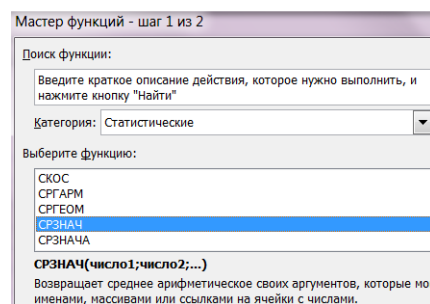


Рисунок 8 – Диалоговое окно **Мастер функций**

## Логические функции Excel (категория Логические)

Функция **«ЕСЛИ»** используется при проверке условий для значений и формул. Синтаксис этой функции: **ЕСЛИ(лог\_выражение; значение\_если\_истина; значение\_если\_ложь)**. Первый аргумент функции «ЕСЛИ» должен содержать логическое выражение. Второй и третий аргументы содержат выражение. В случае если логическое выражение истинно, то результат функции «ЕСЛИ» будет соответствовать ее второму аргументу. Если

логическое выражение ложно, то результат функции «ЕСЛИ» будет соответствовать ее третьему аргументу. Пара примеров использования функции «ЕСЛИ» показана в таблице 2.

Таблица 2 – Примеры создания разветвляющихся функций с помощью встроенной функции «ЕСЛИ»

Разветвляющаяся функция	Формула в ячейке В1
$f(x) = \begin{cases} 2 \cdot x, & x < 0 \\ x^2, & x \geq 0 \end{cases}$	
$f(x) = \begin{cases} 2 \cdot x, & x < 0 \\ x^2, & 0 \leq x \leq 3 \\ 2 \cdot x^2, & x > 3 \end{cases}$	

Функция «И» проверяет на истинность условия и возвращает ИСТИНА, если все условия истинны и возвращает ЛОЖЬ, если хотя бы одно условие ложно (рис. 9).

Функция «ИЛИ» проверяет на истинность условия и возвращает ИСТИНА, если хотя бы одно из условий истинно и возвращает ЛОЖЬ, если все условия ложны (рис. 9).

Функции И() и ИЛИ() воспринимают от 1 до 255 проверяемых условий, которые должны быть логическими значениями.

	A	B	C
1	1	ЛОЖЬ	B1=И(A1>5;A2<5)
2	2	ИСТИНА	B2=И(A1<5;A2<5)
3		ЛОЖЬ	B3=ИЛИ(A1>5;A2>5)
4		ИСТИНА	B4=ИЛИ(A1<5;A2>5)

Рисунок 9 – Примеры использования функций И(), ИЛИ()

### Функции поиска и просмотра данных Excel (категория Ссылки и массивы)

Функция «ПРОСМОТР» позволяет возвращать значение из строки, столбца, массива в зависимости от используемой синтаксической формы: "вектор" или "массив".

*Векторная форма* просматривает диапазон, в который входят значения только одной строки или одного столбца (так называемый вектор) в поисках определенного значения и возвращает значение из другого столбца или строки. Синтаксис векторной формы: ПРОСМОТР(искомое\_значение; вектор\_просмотра; вектор\_результата). Аргумент *искомое\_значение* – значение, которое функция ищет в первом векторе. *Просматриваемый\_вектор* – интервал, содержащий только одну строку или один столбец, причем значения в данном аргументе должны быть расположены в порядке возрастания (в противном случае функция может вернуть неверный результат). *Вектор\_результатов* – интервал, содержащий только одну строку или столбец, при этом он должен иметь тот же размер, что и просматриваемый\_вектор.

*Форма массива* просматривает первую строку или первый столбец массива, находит указанное значение и возвращает значение из аналогичной позиции последней строки или последнего столбца массива. Синтаксис формы массива: ПРОСМОТР(искомое\_значение; массив). Аргумент *искомое\_значение* – значение, которое функция ищет в массиве. *Массив* – интервал ячеек, содержащих текст, числа или логические значения, которые требуется сравнить с искомым\_значением.

*Замечания:* 1) Если ПРОСМОТР не может найти искомое\_значение, то подходящим считается наибольшее значение в аргументе вектор\_просмотра, которое меньше, чем искомое\_значение. 2) Если искомое\_значение меньше, чем наименьшее значение в аргументе вектор\_просмотра, то функция возвращает значение ошибки #Н/Д.

	A	B	C	D	E
1	Справочная таблица				
2		Вес груза (кг)	За доставку (руб)		
3	от 10 до 100 кг.	10	10		
4	от 100 до 500 кг.	100	20		
5	свыше 500 кг.	500	30		
6	Расчетная таблица:				
7	Вес груза (кг):	За доставку:			
8	5	#Н/Д	B9=ПРОСМОТР(A9;B3:B5;C3:C5)		
9	100	20	B10=ПРОСМОТР(A10;B3:C5)		
10	1000	30	B11=ПРОСМОТР(A11;B3:B5;C3:C5)		
11	150	#Н/Д	B11=ВПР(A11;B3:C5;2;0)		
12	20	10	B12=ВПР(A11;B3:C5;2;1)		

Рисунок 10 – Примеры использования функций ПРОСМОТР(), ВПР()

	A	B	C	D	E
1	Справочная таблица				
2	Город	Брест	Кобрин	Минск	
3	За доставку (руб)	0	5	20	
4	Расчетная таблица:				
5	Город	За доставку	Формула		
6	Кобрин	5	B6=ПРОСМОТР(A9;B3:B5;C3:C5)		
7	Барановичи	#Н/Д	B7=ПРОСМОТР(A7;B2:D2;B3:D3)		
8	Минск	20	B8=ПРОСМОТР(A8;B2:D3)		
9	Брест	0	B9=ГПР(A9;B2:D3;2;0)		
10					

Рисунок 11 – Примеры использования функций ПРОСМОТР(), ГПР()

Функция «**ПОИСКПОЗ**» возвращает относительное положение элемента массива, который соответствует заданному значению указанным образом. Синтаксис функции: **ПОИСКПОЗ**(*искомое\_значение*, *просматриваемый\_массив*, *тип\_сопоставления*). Аргумент *искомое\_значение* – это значение, используемое при поиске значения в таблице. *Просматриваемый\_массив* – это непрерывный интервал ячеек, возможно, содержащих искомые значения. *Тип\_сопоставления* – это число -1, 0 или 1, которое указывает, как Excel сопоставляет искомое\_значение со значениями в аргументе просматриваемый\_массив. Если *тип\_сопоставления* = 1, то функция находит наибольшее значение, которое меньше или равно значению аргумента *искомое\_значение*, при этом *Просматриваемый\_массив* должен быть упорядочен по возрастанию (приблизительный поиск). Если *тип\_сопоставления* = 0, то функция находит первое значение, равное аргументу *искомое\_значение*, при этом *Просматриваемый\_массив* может быть не упорядочен (точный поиск). Если *тип\_сопоставления* = -1, то функция находит наименьшее значение, которое больше или равно значению аргумента *искомое\_значение*, при этом *Просматриваемый\_массив* должен быть упорядочен по убыванию. Если аргумент *тип\_сопоставления* опущен, то предполагается, что он = 1.

*Замечания:* 1) **ПОИСКПОЗ** не различает регистры при сопоставлении текстов. 2) Если функция не находит соответствующего значения, то возвращается значение ошибки #Н/Д.

Функция «**ВПР**» – ищет значение в крайнем левом столбце таблицы или массива значений и возвращает значение в той же строке из указанного столбца таблицы, т.е. используется, когда сравниваемые значения расположены в столбце слева от искомых данных. Синтаксис: **ВПР**(*искомое\_значение*; *инфо\_таблица*; *номер\_столбца*; *интервальный\_просмотр*). Аргумент *Искомое\_значение* – это значение, которое должно быть найдено в первом столбце массива. *Инфо\_таблица* – это таблица с информацией, в которой ищутся данные. *Номер\_столбца* – номер столбца в массиве *инфо\_таблица*, в котором должно быть найдено соответствующее значение. *Интерв\_просмотр* – логическое значение, которое определяет, нужно ли, чтобы **ВПР** искала точное или приближенное соответствие. Если этот аргумент имеет значение **ИСТИНА** (= 1) или опущен, то возвращается приблизительно соответствующее значение; т.е., если точное соответствие не найдено, то возвращается наибольшее значение, которое меньше, чем *искомое\_значение* (в этом случае значения в первой строке аргумента *инфо\_таблица* должны быть расположены в возрастающем порядке; в противном случае функция может выдать неправильный результат). Если этот аргумент имеет значение **ЛОЖЬ** (= 0), то функция ищет точное соответствие (если таковое не найдено, то возвращается значение ошибки #Н/Д). В этом случае *инфо\_таблица* не обязана быть сортированной.

Функция «**ГПР**» – ищет значение в верхней строке таблицы или массива значений и возвращает значение в том же столбце из заданной строки таблицы или массива, т.е. используется, когда сравниваемые значения расположены в верхней строке таблицы данных, а возвращаемые значения расположены на несколько строк ниже. Синтаксис: **ГПР**(*искомое\_значение*; *инфо\_таблица*; *номер\_строки*; *интерв\_просмотр*). Аргумент *Искомое\_значение* – это значение, которое должно быть найдено в первом столбце массива. *Инфо\_таблица* – это таблица с информацией, в которой ищутся данные. *Номер\_строки* – это номер строки в массиве *инфо\_таблица*, из которой будет возвращено сопоставляемое значение. Назначение аргумента *Интерв\_просмотр* то же, что для функции **ВПР**.

*Замечания для функций ВПР(ГПР):* 1) Если *искомое\_значение* меньше, чем наименьшее значение в первом столбце (верхней строке) *инфо\_таблицы*, то функция возвращает значение ошибки #Н/Д. 2) Если значение аргумента *номер\_столбца* (*номер\_строки*) меньше 1, то функция возвращает значение ошибки #ЗНАЧ!. 3) Если значение аргумента *номер\_столбца* (*номер\_строки*) больше, чем количество столбцов в аргументе *инфо\_таблица*, то функция возвращает значение ошибки #ССЫЛ!.

Поиск данных в трехмерной таблице (расположение информации в которой осуществляется в нескольких разрезах: заголовков строк и заголовков столбцов) предполагает совместное использование функций поиска. При этом функция **ПОИСКПОЗ** является вложенной для функций **ВПР** или **ГПР** (на месте третьего аргумента: *номер\_столбца* или *номер\_строки*). Примеры такого использования (правильного, ошибочного – без учета замечаний) приведены на рисунке 12. В примерах, где функции возвращают сообщение об ошибке, подчеркиванием выделены исходные данные или аргументы, которые эти ошибки провоцируют.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Справочная таблица: Размер штрафа(%) в зависимости от срока (дней) и суммы задолженности (руб)					Расчет величины штрафа (руб): сумма задолженности * % штрафа								
1														
2	<i>Срок</i>	10	30	60		<i>Сумма долга</i>	<i>Срок (дней)</i>	<i>Штраф (%)</i>	<i>Штраф (руб)</i>					
3	<i>Сумма</i>	100	1%	2%	5%	100	5	#Н/Д	#Н/Д	H3=ВПР(F3;A3:D5;ПОИСКПОЗ(G3;A2:D2;1);1)				
4		200	2%	3%	10%	50	10	#Н/Д	#Н/Д	H4=ВПР(F4;A3:D5;ПОИСКПОЗ(G4;A2:D2;1);1)				
5		300	3%	5%	15%	150	10	#Н/Д	#Н/Д	H5=ВПР(F5;A3:D5;ПОИСКПОЗ(G5;A2:D2;1);0)				
6						250	15	2%	5	H6=ВПР(F6;A3:D5;ПОИСКПОЗ(G6;A2:D2;1);1)				
7						350	65	0,15	52,5	H7=ГПР(F7;B2:D5;ПОИСКПОЗ(F7;A2:A5;1);1)				

Рисунок 12 – Примеры использования функций **ВПР** (), **ГПР** () и **ПОИСКПОЗ** ()



## Построение графика функции одной переменной

Для построения графиков в Excel используются диаграммы. Диаграммы создаются на основе данных, содержащихся на рабочем листе. Порядок создания диаграммы:

- Определите данные, по которым будет построена диаграмма.
- Выделите диапазон ячеек, содержащий эти данные с заголовками строк/столбцов.
- Выполните команду (2003: **Вставка/Диаграмма/Точечная**; 2007+: **Вставка/Диаграммы/Точечная**).

В лабораторных работах мы будем использовать тип диаграммы **Точечная** (рис. 13) различных подтипов (с маркерами или без них, с гладкими кривыми или с прямыми отрезками и др.), которая позволит задавать значения координат точки (маркера) по горизонтальной и вертикальной оси.

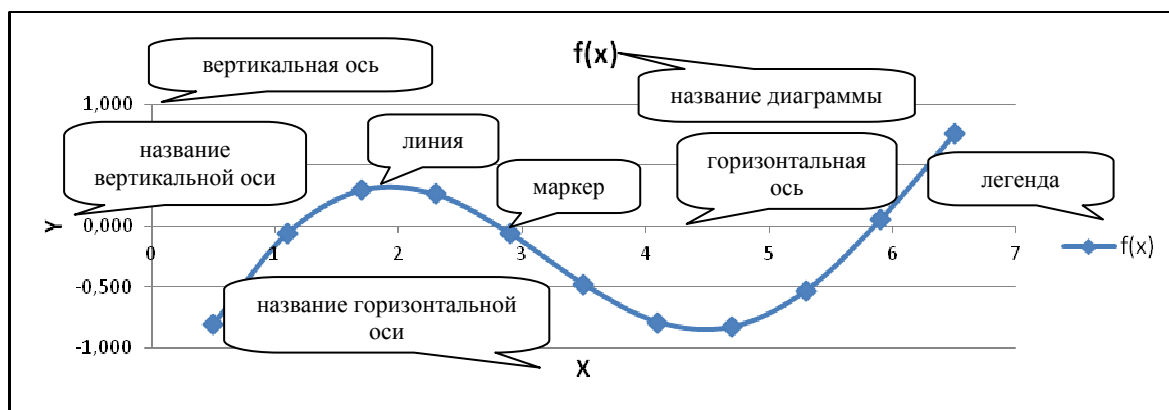


Рисунок 13 – Элементы диаграммы **Точечная** с гладкими кривыми и маркерами

Чтобы изменить настройки любого элемента диаграммы, необходимо правой клавишей мыши щелкнуть по этому элементу и в контекстном меню выбрать формат этого элемента. Для каждого элемента открывается свой определенный набор настроек. Для изменения размера диаграммы в Excel 2007+ можно воспользоваться контекстной вкладкой диаграммы **Формат/Размер**.

## Определение отрезков, содержащих нули и экстремумы функции. Определение точек наибольшего и наименьшего значения функции

Для того чтобы определить отрезки, содержащие нули, экстремумы функции, либо точки, имеющие наибольшее и наименьшее значения функции, необходимо предварительно протабулировать функцию, т.е. построить таблицу ее значений при различных значениях аргумента  $x$ , который обычно изменяется равномерно с фиксированным шагом  $h$ . Для примера, возьмем функцию  $y(x) = \sin x + \sqrt{x} - 2$ . В полученной таблице (рис. 14) необходимо найти места, удовлетворяющие следующим критериям:

- Произведение значений координат  $Y$  соседних точек меньше нуля, или где значение функции меняет свой знак. Соответствующие этим значениям  $Y$  координаты  $X$  образуют отрезок, содержащий нуль функции. В нашем примере нуль функции на участке  $[5,3..5,9]$ .

- Значение координаты  $Y$  выбранной точки меньше значения координаты  $Y$  предыдущей и последующей точки. Координаты  $X$  предыдущей и последующей точки образуют отрезок, содержащий локальный минимум функции. Локальный минимум на участке  $[4,1..5,3]$ .

- Значение координаты  $Y$  выбранной точки больше значения координаты  $Y$  предыдущей и последующей точки. Координаты  $X$  предыдущей и последующей точки образуют отрезок, содержащий локальный максимум функции. Локальный максимум на участке  $[1,1..2,3]$ .

- Наибольшее значение координаты  $Y$  во всей таблице. Соответствующая этой координате координата  $X$  укажет на место нахождения максимального значения функции в рассматриваемом диапазоне среди приведенных точек. Максимальное значение функции при  $X = 6,5$ .

- Наименьшее значение координаты  $Y$  во всей таблице. Соответствующая этой координате координата  $X$  укажет на место нахождения минимального значения функции в рассматриваемом диапазоне среди приведенных точек. Минимальное значение функции при  $X = 4,7$ .

X	Y
0,5	-0,813
1,1	-0,060
1,7	0,296
2,3	0,262
4,1	-0,793
4,7	-0,832
5,3	-0,530
5,9	0,055
6,5	0,765

<b>Локальный максимум</b>	
1,1	-0,060
1,7	0,296
2,3	0,262
<b>Локальный минимум</b>	
4,1	-0,793
4,7	-0,832
5,3	-0,530
<b>Минимальное значение</b>	
4,7	-0,832
<b>Нуль функции</b>	
5,3	-0,530
5,9	0,055
<b>Максимальное значение</b>	
6,5	0,765

Рисунок 14 – Определение характерных отрезков

## Уточнение нулей функции с помощью инструмента «Подбор параметра»

Инструмент «Подбор параметра» (2003: **Сервис/Подбор параметра**; 2007+: **Данные/Работа с данными/Анализ «что-если»/Подбор параметра**) является частью блока задач анализа «что-если», когда желаемый результат одиночной формулы известен, но неизвестны значения, которые требуется ввести для получения этого результата.

Например, уточним нуль функции  $y(x) = \sin x + \sqrt{x} - 2$  из рассмотренного ранее примера.

Порядок выполнения с помощью средства **Подбор параметра**:

1. Зарезервируем ячейку A5 на рабочем листе Excel с установленным в ней начальным приближением (значение из интервала [5,3..5,9]) для поиска значения нуля.
2. В соседней ячейке (B5) введём формулу функции, в которой в качестве аргумента  $x$  надо сослаться на зарезервированную для результата ячейку – A5 (рис. 15).
3. Откроем диалоговое окно **Подбор параметра** (рис. 16), в котором:

- в поле «**Установить в ячейке**» укажем адрес ячейки с формулой функции – B5;
- в поле «**Значение**» введём 0 (т.к. по определению в математике нуль функции  $x^*$  – это точка из области определения функции, в которой она принимает нулевое значение:  $y(x^*) = 0$ );
- в поле «**Изменяя значение ячейки**» укажем адрес зарезервированной для результата ячейки – A5.

Если в диалоговом окне «Результат подбора параметра» будет выдано сообщение «Решение найдено», то в подбираемой ячейке (A5) можно будет увидеть результат (рис. 17).

	A	B	C	D
1	X	Y		
2	5,3	-0,530		
3	5,9	0,055		
4	Уточненный нуль			
5	5	-0,723		

Рисунок 15 – Исходные данные для уточнения нуля функции

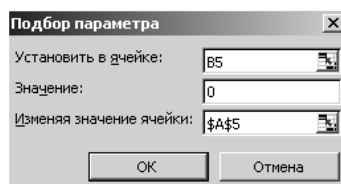


Рисунок 16 – Окно Подбор параметра

	A	B	C	D	E	F
1	X	Y				
2	5,3	-0,530				
3	5,9	0,055				
4	Уточненный нуль					
5	5,8509	0,000				
6						
7						

Рисунок 17 – Результат выполнения

**Замечание:** Excel в своих алгоритмах инструментов анализа данных использует метод подстановки. Он подставляет вместо  $x$  разные значения и анализирует, насколько результат вычислений отклоняется от условий, указанных в параметрах инструмента. Как только будет достигнут результат вычисления с максимальной точностью, процесс подстановки прекращается. По умолчанию инструмент выполняет 100 повторений (итераций) с точностью 0.001. Если есть необходимость в повышении точности вычислений, можно изменить настройки: увеличить параметр предельного числа итераций или изменить относительную погрешность (2003: **Сервис/Параметры/вкладка Вычисления**; 2007+: **Файл/Параметры Excel/Формулы/Параметры вычислений**).

## Настройка «Поиск решения»

Доступ к инструменту **Поиск решения** осуществляется с помощью команды (2003: **Сервис/Поиск решения**; 2007+: **Данные/Анализ/Поиск решения**). Если в группе ленточных команд Excel 2007+ **Данные/Анализ** отсутствует команда **Поиск решения**, то необходимо загрузить соответствующую надстройку.

- Выберите команду **Office/Параметры Excel**.
- В диалоговом окне **Параметры Excel** щелкните на вкладке **Надстройки**.
- В нижней части вкладки в раскрывающемся списке **Управление** выберите опцию **Надстройки Excel** и щелкните на кнопке **Перейти**. Откроется диалоговое окно **Надстройки**.
- В диалоговом окне **Надстройки** в списке **Доступные надстройки** установите флажок **Поиск решения** и щелкните на кнопке **ОК**.

В Excel 2003 диалоговое окно **Надстройки** открывается командой **Сервис/Надстройки**.

После выполнения этих действий команда **Поиск решения** будет доступной.

## Уточнение нулей и экстремумов функции с помощью надстройки «Поиск решения»

Например, из предыдущего примера на рисунке 14, скопируем с помощью **Специальной вставки значения** координат трех точек, в диапазоне которых содержится локальный максимум в ячейки A2:B4 (рис. 18). В отдельной ячейке A6 зададим значение  $X$  средней точки, в соседней ячейке B6 определим по формуле значение  $Y$  от указанного значения  $X$ .

В диалоговом окне **Поиск решения** задаем целевую ячейку B6 (в целевой ячейке обязательно должна находиться формула). Поскольку наша цель – максимизировать значение в этой ячейке, устанавливаем переключатель **Равной максимальному значению**. Затем определяем изменяемые ячейки, которые в данном случае состоят из одной ячейки A6. Далее задаем ограничения, по нажатию кнопки **Добавить** ( $A6 \leq A4$  и  $A6 \geq A2$ , т.е. найденное значение  $X$  должно быть больше начала отрезка A2 и одновременно меньше конца отрезка A4). Заданные ограничения отображаются в списке **Ограничения** (рис. 19).

	A	B	C	D
1	X	Y		
2	1,1	-0,060		
3	1,7	0,296		
4	2,3	0,262		
5	Уточненное значение			
6	1,7	0,296		

Рисунок 18 – Исходные данные для определения локального максимума

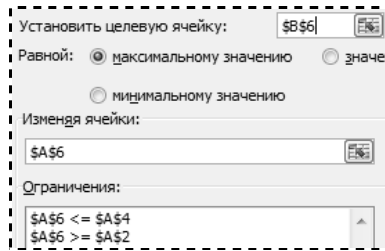


Рисунок 19 – Окно Поиск решения

	A	B	C	D
1	X	Y		
2	1,1	-0,060		
3	1,7	0,296		
4	2,3	0,262		
5	Уточненное значение			
6	1,938153	0,325		

Рисунок 20 – Результат работы процедуры Поиск решения

Чтобы начать процесс решения задачи, щелкните по кнопке **Выполнить**. В строке состояния будет отображаться ход решения задачи. Через некоторое время отобразится информация о том, что решение найдено. В ячейке A6 появится значение X, при котором функция имеет максимальное значение на участке [1,1..2,3] (рис. 20).

Значение локального минимума определяется аналогично. Все отличие состоит в том, что в отдельные ячейки копируются координаты трех точек, входящих в отрезок, содержащий локальный минимум и в диалоговом окне **Поиск решения** переключатель устанавливается в положение **Равной минимальному значению**.

Похожая ситуация с уточнением нуля функции. В отдельные ячейки копируются координаты двух точек, входящих в отрезок, содержащий нуль функции и в диалоговом окне **Поиск решения** переключатель устанавливается в положение **Равной значению 0**.

### Решение СЛАУ с помощью надстройки «Поиск решения»

В этом примере покажем, как можно с помощью надстройки **Поиск решения** решить систему из двух линейных алгебраических уравнений с двумя неизвестными. Пусть имеется следующая система линейных уравнений:

$$\begin{cases} 2 \cdot x + 3 \cdot y = 15 \\ -9 \cdot x + 6 \cdot y = 18 \end{cases}$$

Задача формулируется так: найти такие значения переменных X и Y, которые удовлетворяли бы одновременно двум уравнениям.

Вводим на рабочий лист исходные данные и формулы (рис. 21).

	A	B	C	D	E	F
1		x	y			
2		0	0			
3						
4				Левая часть	=	Правая часть
5	ур. 1	2	3	0	=	15
6	ур. 2	-9	6	0	=	18

Рисунок 21 – Исходные данные для решения системы уравнений

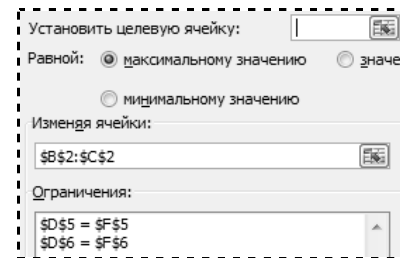


Рисунок 22 – Окно Поиск решения

На листе используем две ячейки B2 и C2, которые до запуска инструмента **Поиск решения** будут содержать начальные приближения переменных X и Y (здесь, например, мы задали их равными нулю). Диапазон ячеек B5:C6 содержит коэффициенты при неизвестных X и Y в первом и втором уравнениях. Ячейки D5 и D6 содержат формулы, по которым определяются значения левых частей обоих уравнений. В ячейки F5 и F6 заносим известные значения правых частей уравнений.

Запускаем средство **Поиск решения** (рис. 22). В этой задаче нет целевой ячейки, поскольку нам нет нужды что-то минимизировать или максимизировать. Поэтому поле **Установить целевую ячейку** диалогового окна **Поиск решения** остается пустым. Изменяемые ячейки – B2 и C2: от них зависит значения левых частей уравнений, которые в свою очередь должны быть равны правым частям этих уравнений. Отсюда вытекают следующие ограничения – D5 = F5 и D6 = F6 (равенство обеих частей уравнений).

	A	B	C	D	E	F
1		x	y			
2		0,92308	4,38462			
3						
4				Левая часть	=	Правая часть
5	ур. 1	2	3	15	=	15
6	ур. 2	-9	6	18	=	18

Рисунок 23 – Результат работы процедуры Поиск решения

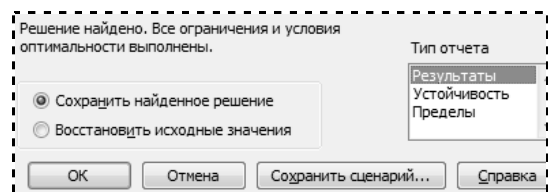


Рисунок 24 – Выбор типа отчета

По завершении процедуры **Поиск решения** ячейки B2 и C2 будут содержать решение ( $X = 0,92308$  и  $Y = 4,38462$ ) заданной системы линейных алгебраических уравнений (рис. 23). Также по результатам работы процедуры **Поиск решения** можно сформировать отчет на отдельном листе (рис. 24).

### Работа с формулами массивов

Когда вы вводите формулу массива в ячейку или диапазон ячеек, необходимо завершить ввод с помощью комбинации клавиш <Ctrl+Shift+Enter>.

Вы всегда сможете отличить формулу массива, так как в строке формул она заключена в фигурные скобки. При вводе формулы массива фигурные скобки вводить не нужно – Excel сделает это сама. Если результат работы формулы массива – несколько значений, то, прежде чем вводить формулу, не забудьте выделить нужные ячейки. Если этого не сделать, то в ячейке отобразится только первое значение массива.

Последовательность операций при использовании формулы массива:

- Выделить диапазон ячеек будущего ответа.
- Ввести формулу.
- Нажать комбинацию клавиш <Ctrl+Shift+Enter>.

Если формула массива помещена в несколько ячеек, вы должны отредактировать все ячейки диапазона как одну ячейку. Запомните, нельзя изменить только один элемент, содержащий формулу массива!

### Решение СЛАУ с помощью обратной матрицы

Воспользуемся предыдущим условием и решим систему из двух линейных алгебраических уравнений с двумя неизвестными, используя формулы массивов. В матричной форме указанная система уравнений выглядит так:

$$A \cdot z = b,$$

где  $A$  – массив, содержащий коэффициенты при неизвестных;

$z$  – вектор-столбец, содержащий неизвестные  $X$  и  $Y$ ;

$b$  – вектор-столбец, содержащий свободные члены уравнений.

Чтобы найти вектор неизвестных, необходимо обратную матрицу  $A$  ( $A^{-1}$ ) умножить на вектор  $b$ .

Для нахождения обратной матрицы ( $A^{-1}$ ) можно использовать встроенную функцию **МОБР(массив)**.

Для скалярного перемножения матриц ( $A^{-1} \cdot b$ ) можно использовать встроенную функцию **МУМНОЖ(массив\_1; массив\_2)**.

Прделаем эти операции на рабочем листе Excel (рис. 25).

	A	B	C	D	E	F
1	A=	2	3		b=	15
2		-9	6			18
3						
4	A =	39	- определитель матрицы			
5						
6	A <sup>-1</sup> =	0,153846	-0,07692		z= (x)	0,923077
7		0,230769	0,051282		(y)	4,384615

Рисунок 25 – Решение СЛАУ матричным способом

Ячейку B4 используем для вычисления определителя матрицы  $A$  (=МОПРЕД(B1:C2)). Если определитель матрицы  $A$  не равен нулю (матрица невырожденная), значит существует ей обратная матрица.

Диапазон B6:C7 = МОБР(B1:C2) содержит элементы обратной матрицы  $A$ , а диапазон F6:F7 = МУМНОЖ(B6:C7;F1:F2) – результат решения СЛАУ.

### Использование формы для ввода данных в таблицу

При вводе данных в таблицу Excel (пример такой таблицы представлен на рис. 26) можно использовать **форму** ввода данных – диалоговое окно, которое автоматически создается после определения заголовка списка.

С помощью формы можно также осуществлять поиск и редактирование записей, которые удовлетворяют простому или множественному критерию сравнения, в котором условия могут соответствовать логической операции **И**.

	A	B	C	D	E	F
1	<b>Наименование товара</b>	<b>Наименование магазина</b>	<b>Дата реализации</b>	<b>Кол-во</b>	<b>Цена</b>	<b>Выручка</b>
2	Газовая плита	ЦУМ	10-май-2010	5	100 000р.	500 000р.
3	Микроволновая печь	1000 мелочей	12-май-2010	8	150 000р.	1 200 000р.
35	Варочная панель	тд Немига	10-май-2010	1	150 000р.	150 000р.
36	Электрическая плита	ЦУМ	15-май-2010	9	150 000р.	1 350 000р.
37	Соковыжималка	ЦУМ	30-январь-2012	3	230 000р.	690 000р.

Рисунок 26 – Таблица исходных данных

Для вызова окна "Форма" (рис. 27) необходимо поместить указатель ячейки в любое место внутри списка и выбрать команду **Данные/Форма** (для Excel 2003). В Excel 2007+ команды Форма нет на ленте, поэтому её надо добавить на панель быстрого доступа, для чего:

- щелкнуть правой кнопкой мыши на панели быстрого доступа и выбрать в контекстном меню пункт **Настройка панели быстрого доступа**. На экране появится раздел **Панель быстрого доступа** диалогового окна "Параметры Excel".
- в раскрывающемся списке **Выбрать команды из** необходимо выбрать **Команды не на ленте**. в списке ниже выбрать **Форма**, а затем нажать кнопку **Добавить**.
- нажать кнопку **ОК**, чтобы закрыть диалоговое окно "Параметры Excel".

После этого панель быстрого доступа будет включать новый значок

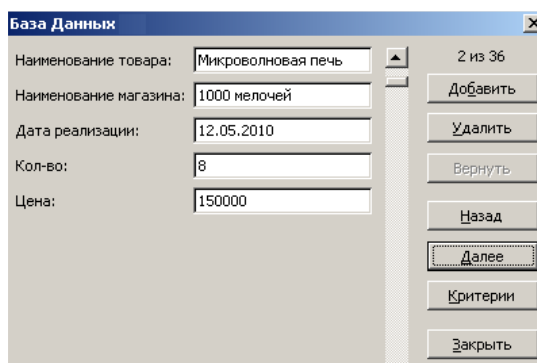


Рисунок 27 – Диалоговое окно **Форма**

### Сортировка таблицы

Сортировка переупорядочивает строки таблицы на основе значений из одного столбца. Для реализации сортировки по выбранному столбцу или множественной сортировки необходимо воспользоваться диалоговым окном **Сортировка** (2003: **Данные/Сортировка**; 2007+: **Главная/Редактирование/Сортировка и фильтр/Настраиваемая сортировка**).

В диалоговом окне **Сортировка** надо указать, по какому столбцу будет идти сортировка и в каком порядке (возрастающем или убывающем). При задании множественной сортировки необходимо вначале указать столбец, по которому будет выполняться первоначальная сортировка, затем столбец, в котором сортировка будет выполняться в случае совпадения значений в предыдущем столбце и т.д.

*Например, отсортируем таблицу по названию магазина в алфавитном порядке, а для одинаковых магазинов отсортируем их по выручке в возрастающем порядке.* Для этого выделим любую ячейку в таблице, затем откроем и заполним диалоговое окно **Сортировка** как показано на рисунке 28.

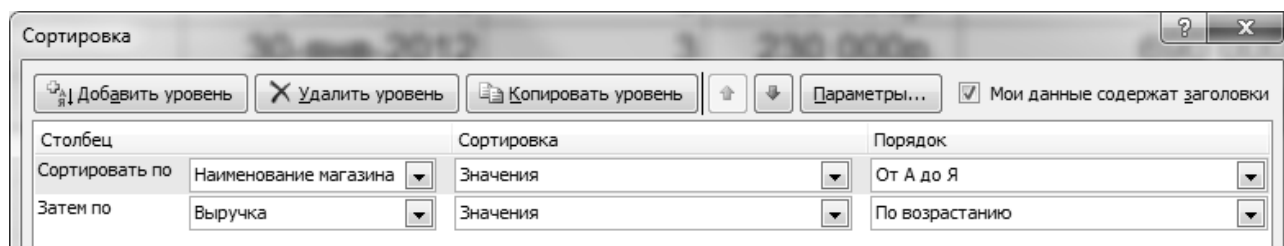


Рисунок 28 – Диалоговое окно **Сортировка** для выполнения множественной сортировки (2007+)

Результат множественной сортировки показан на рисунке 29.

	A	B	C	D	E	F
	<b>Наименование товара</b>	<b>Наименование магазина</b>	<b>Дата реализации</b>	<b>Кол-во</b>	<b>Цена</b>	<b>Выручка</b>
1						
2	Духовой шкаф	1000 мелочей	11-май-2010	2	150 000р.	300 000р.
3	Портативная газ/плита	1000 мелочей	15-май-2010	5	100 000р.	500 000р.
35	Духовой шкаф	ЦУМ	15-май-2010	9	110 000р.	990 000р.
36	Электрическая плита	ЦУМ	12-май-2010	9	140 000р.	1 260 000р.
37	Электрическая плита	ЦУМ	15-май-2010	9	150 000р.	1 350 000р.

Рисунок 29 – Результат множественной сортировки

### Условное форматирование

Условное форматирование позволяет задать определенный формат диапазона ячеек в зависимости от содержания этого диапазона. Как правило, когда вы задаете условия для форматирования диапазона ячеек (2003: **Формат/Условное форматирование/Формула**; 2007+: **Главная/Стили/Условное форматирование/Использовать формулу для определения форматируемых ячеек**), Excel проверяет каждую ячейку диапазона для определения ее соответствия заданным вами условиям. Затем Excel применяет выбранный вами для данного условия формат во всех ячейках, удовлетворяющих этому условию. Если содержимое ячейки не отвечает любому из заданных условий, форматирование ячейки не меняется.

Предположим, что необходимо выделить ячейки массива A1:B2, в которых значения *больше среднего значения чисел, содержащихся в этом массиве*. Выделяем на рабочем листе диапазон A1:B2 и создаем формулу в диалоговом окне **Условное форматирование** (таблица 3).

Таблица 3 – Пример создания формулы в диалоговом окне **Условное форматирование**

Фрагмент рабочего листа	Фрагмент формы <b>Создание правила форматирования</b> (формула)									
<table border="1"> <tr> <td></td> <td>A</td> <td>B</td> </tr> <tr> <td>1</td> <td>5</td> <td>2</td> </tr> <tr> <td>2</td> <td>3</td> <td>4</td> </tr> </table>		A	B	1	5	2	2	3	4	<p><b>Форматировать значения, для которых следующая формула является истинной:</b></p> <p>=A1&gt;CPЗНАЧ(\$A\$1:\$B\$2)</p>
	A	B								
1	5	2								
2	3	4								

Первая ссылка в формуле относительная – при автоматическом тиражировании формулы в другие ячейки эта ссылка должна изменяться. Вторая ссылка на диапазон абсолютная – при тиражировании этот диапазон не должен изменяться.

### Промежуточные итоги

С помощью команды **Промежуточные итоги** можно автоматически подсчитать промежуточные и общие итоги в списке для столбца таблицы. Перед выполнением этой команды необходимо обязательно выполнить сортировку по тому столбцу, по которому подводятся итоги.

Параметры промежуточных итогов задаются в диалоговом окне **Промежуточные итоги** (2003: **Данные/Итоги**; 2007+: **Данные/Структура /Промежуточные итоги**).

В случае *если необходимо определить сумму выручки по каждому магазину*, то предварительно выполняется сортировка как показано в предыдущем пункте. Затем для этой таблицы открывается диалоговое окно **Промежуточные итоги** (рис. 30). В диалоговом окне:

- в поле **При каждом изменении в** выбрать столбец для подсчета итогов – **Наименование магазина**;
- в поле **Операция** выбрать итоговую функцию для вычисления промежуточных итогов – **Сумма**;
- в поле **Добавить итоги по** установить флажок для каждого столбца, содержащего значения, по которым необходимо подвести итоги – **Выручка**.

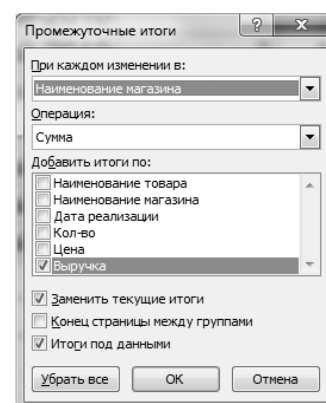


Рисунок 30 – Диалоговое окно **Промежуточные итоги**

При необходимости команду **Промежуточные итоги** можно использовать снова, чтобы добавить дополнительные строки итогов с использованием других функций. Во избежание перезаписи имеющихся итогов снимите флажок **Заменить текущие итоги**. Если наоборот необходимо заменить предыдущие итоги, то устанавливаем флажок **Заменить текущие итоги**.

Чтобы расположить итоговую строку под строкой данных, установить флажок **Итоги под данными**.

Результат выполнения настроек диалогового **Промежуточные итоги** показан на рисунке 31.

	A	B	C	D	E	F
	Наименование товара	Наименование магазина	Дата реализации	Кол-во	Цена	Выручка
1						
10		<b>1000 мелочей Итог</b>				5 920 000р.
11	Духовой шкаф	тд Граффи	14-май-2010	1	120 000р.	120 000р.
12	Портативная газ/плита	тд Граффи	12-май-2010	3	140 000р.	420 000р.
13	Газовая плита	тд Граффи	13-май-2010	4	110 000р.	440 000р.
14	Портативная газ/плита	тд Граффи	11-май-2010	5	150 000р.	750 000р.
15	Духовой шкаф	тд Граффи	13-май-2010	8	110 000р.	880 000р.
16	Газовая плита	тд Граффи	12-май-2010	6	150 000р.	900 000р.
17	Микроволновая печь	тд Граффи	12-май-2010	9	150 000р.	1 350 000р.
18		<b>тд Граффи Итог</b>				4 860 000р.
27		<b>тд Немига Итог</b>				5 110 000р.
41		<b>ЦУМ Итог</b>				9 540 000р.
42		<b>Общий итог</b>				25 430 000р.

Рисунок 31 – Результат подведения промежуточных итогов

### Автофильтрация таблицы

Фильтрация таблицы означает отображение в таблице только тех строк, значения в которых удовлетворяют определенным условиям. Включить автофильтрацию можно выбрав соответствующую команду (2003: **Данные/Фильтр/Автофильтр**; 2007+: **Данные/Сортировка и фильтр/Фильтр**). Справа от заголовков (в шапке таблицы) столбцов появятся кнопки со стрелками автофильтра . Если щелкнуть стрелку автофильтра, отображается список различных вариантов фильтрации и сортировки по возрастанию и убыванию. При выполнении фильтрации стрелка активного автофильтра окрашивается в синий цвет.

Для того чтобы после фильтрации отображались все записи необходимо нажать активную кнопку и выбрать команду (2003: **Все**; 2007+: **Снять фильтр с «...»**) или выполнить команду (2003: **Данные/Фильтр/Отобразить**

все; 2007+: **Данные/Сортировка и фильтр/Очистить**), а для удаления автофильтра необходимо снять флажок команды **Данные/Фильтр/Автофильтр** (2003) либо отключить его **Данные/Сортировка и фильтр/Фильтр** (2007+).

На примере исходной таблицы (рис. 26) предположим, что необходимо *отобразить данные только по магазину ЦУМ*. Для этого необходимо открыть раскрывающийся список в заголовке столбца **Наименование магазина** и установить в нем сначала флажок напротив слова **ЦУМ**. При этом, на кнопке раскрытия списка заголовка столбца **Наименование магазина** появится значок, который означает, что таблица отфильтрована по значению этого столбца (рис. 32).

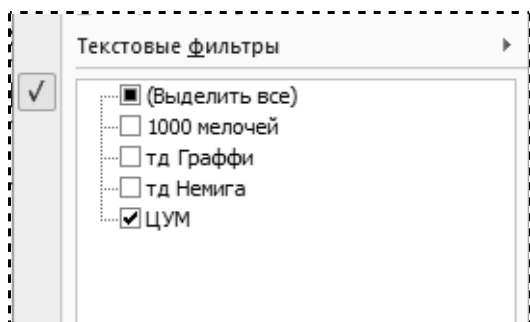


Рисунок 32 – Раскрывающийся список в заголовке столбца **Наименование магазина**

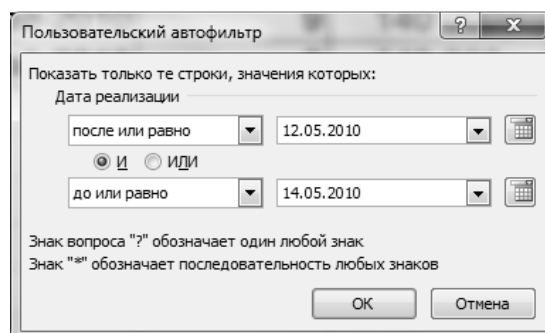


Рисунок 33 – Пользовательский фильтр в столбце **Дата реализации**

Большие возможности для фильтрации таблиц предоставляют находящиеся в раскрывающихся списках заголовков столбцов команды **Текстовые фильтры** (если в столбце записаны текстовые данные), **Числовые фильтры** (если в столбце хранятся числовые данные) и **Фильтры по дате** (если значения в столбце отформатированы одним из форматов даты или времени).

Для фильтрации данных можно использовать любое количество столбцов таблицы. Например, в исходной таблице (рис. 26) *можно использовать фильтр, когда в столбце **Наименование магазина** задан магазин **ЦУМ*** (рис. 32), *а в столбце **Дата реализации** – диапазон с 12.05.2010 по 14.05.2010* (рис. 33).

Результат применения автофильтра по двум столбцам таблицы показан на рисунке 34.

	A	B	C	D	E	F
1	<b>Наименование товара</b>	<b>Наименование магазина</b>	<b>Дата реализации</b>	<b>Кол-во</b>	<b>Цена</b>	<b>Выручка</b>
13	Варочная панель	ЦУМ	13-май-2010	6	110 000р.	660 000р.
17	Портативная газ/плита	ЦУМ	13-май-2010	2	150 000р.	300 000р.
20	Микроволновая печь	ЦУМ	12-май-2010	4	140 000р.	560 000р.
26	Варочная панель	ЦУМ	12-май-2010	1	120 000р.	120 000р.
29	Электрическая плита	ЦУМ	12-май-2010	9	140 000р.	1 260 000р.
31	Газовая плита	ЦУМ	12-май-2010	6	140 000р.	840 000р.

Рисунок 34 – Результат фильтрации по нескольким столбцам

### Расширенный фильтр

Команда «Расширенный фильтр» используется для фильтрации по более сложным условиям отбора записей, чем автофильтр, например, по нескольким условиям отбора в одном столбце, по нескольким условиям отбора в нескольких столбцах или для отбора записей по условиям отбора с помощью формулы.

Перед применением команды «Расширенный фильтр» необходимо подготовить условия отбора, которые, как правило, располагаются в отдельных от таблицы ячейках.

Создание диапазона условий отбора следующее:

- в выбранной свободной строке заполнить ячейки заголовками полей, по значениям которых будет происходить фильтрация;
- строки ниже заполнить значениями, которые и будут являться критериями расширенного фильтра.

Например, критерий для расширенного фильтра, *когда в столбце **Наименование магазина** задан магазин **ЦУМ**, а в столбце **Дата реализации** – диапазон с 12.05.2010 по 14.05.2010*, будет выглядеть так, как показано на рисунке 35.

	A	B	C
40	<b>Наименование магазина</b>	<b>Дата реализации</b>	<b>Дата реализации</b>
41	ЦУМ	>=12.05.2010	<=14.05.2010

Рисунок 35 – Критерий для расширенного фильтра

После того как условие отбора подготовлено необходимо выполнить следующие действия:

- выделить любую ячейку в таблице;
- открыть окно **Расширенный фильтр** (2003: **Данные/Фильтр/Расширенный фильтр**; 2007+: **Данные/Сортировка и фильтр/Дополнительно**) – рисунок 54;
- установить переключатель **Обработка** в положение **Скопировать результаты в другое место**, если необходимо, чтобы Excel выводил результаты фильтрации рядом с таблицей и указать диапазон или ячейку для начала размещения результатов в поле **Поместить результат в диапазон**;
- проверить исходный диапазон ячеек фильтруемой таблицы, диапазон должен содержать все ячейки списка с учетом ячеек заголовков столбцов;
- указать диапазон критериев, он должен содержать все ячейки диапазона условий отбора (рис. 36) с учетом ячеек заголовков столбцов;

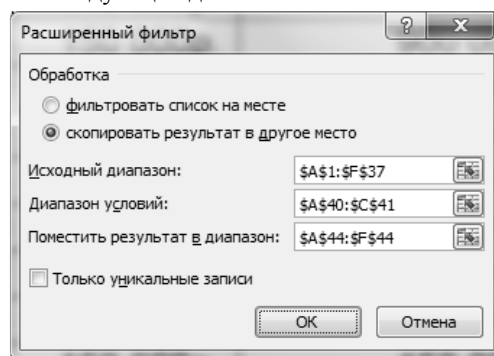


Рисунок 36 – Диалоговое окно **Расширенный фильтр**

- нажать на кнопку **ОК** для выполнения фильтрации. Начиная с ячейки A44, будут отображены результаты фильтрации (рис. 34).

При формировании критерия отбора расширенного фильтра возможны следующие варианты.

1. Критерий отбора содержит одно или несколько условий, накладываемых на один столбец (одно поле). Если критерий содержит несколько условий, то они связываются логической операцией **ИЛИ** (рисунок 37б).

2. Критерий отбора содержит несколько условий, накладываемых на несколько столбцов (полей) одновременно. Здесь возможны следующие варианты:

- необходимо наложить несколько условий отбора на несколько столбцов, причем эти условия должны связываться логической операцией **ИЛИ**. Тогда условия отбора задаются в разных строках критерия (рис. 37в);
- необходимо одновременно наложить несколько условий отбора на несколько полей, причем условия отбора должны быть связаны логической операцией **И**. Тогда все условия задаются в одной строке критерия (рис. 35, 37г);
- необходимо несколько условий наложить на несколько полей, причем связываться они могут обеими логическими операциями **И** и **ИЛИ** (рис. 37а).

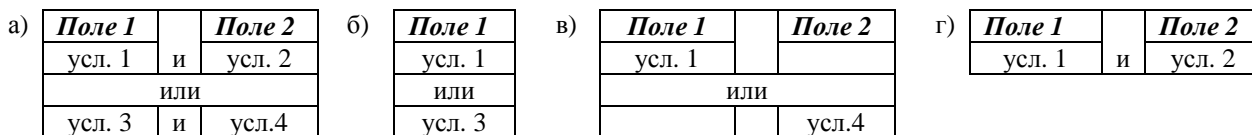


Рисунок 37 – Правила составления критериев для расширенного фильтра

3. Вычисляемый критерий. Условия отбора могут содержать формулу. Полученное в результате вычисления формулы значение будет участвовать в сравнении. Правила формирования вычисляемого критерия следующие:

- в диапазоне критерия нельзя указывать существующие в таблице имена полей. Следует ввести новое имя заголовка;
- при создании формул вычисляемых критериев следует использовать первую строку списка (не строку заголовков), т. е. первую ячейку в сравниваемом столбце;
- если в формуле используются ссылки на ячейки списка, они задаются как относительные;
- если в формуле используются ссылки на ячейки вне списка, они задаются как абсолютные;
- вычисляемые критерии можно сочетать с невычисляемыми;
- не следует обращать внимание на результат, выдаваемый формулой в области критерия (обычно ИСТИНА или ЛОЖЬ).

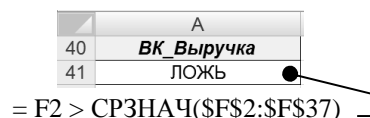


Рисунок 38 – Пример вычисляемого критерия

Для условия с вычисляемым критерием *приведем пример, в котором необходимо отыскать записи* (исходная таблица – рис. 26) *с выручкой выше средней по всему столбцу* (рис. 38).



## Сводная таблица

Инструмент **Сводная таблица** является чрезвычайно мощным и удобным средством анализа и трансформации данных. С его помощью можно сделать практически любой «разрез» таблицы. Сводная таблица во многом объединяет возможности других обобщающих инструментов, таких как **Итоги**, **Фильтр**, **Консолидация**. В таких таблицах можно произвольным образом изменить структуру исходных данных, а также получить все необходимые итоги по любым данным и сортировки.

Порядок создания сводных таблиц одинаков. Отличие только в ее настройках, которые зависят от конечной цели. Разберем построение сводной таблицы, которая показала бы на основе исходной таблицы (рис. 29) информацию о том, какова суммарная выручка от реализации каждого товара в каждом магазине и отобразила в качестве страницы в разрезе поле «Дата реализации».

Для создания сводной таблицы в Excel 2003 необходимо выделить источник данных (таблицу с заголовками столбцов) и запустить инструмент **Мастер сводных таблиц**. Перед построением сводной таблицы из источника данных обязательно должны быть убраны промежуточные итоги и наложенные фильтры.

Запустить **Мастер сводных таблиц** можно путем выбора команды **Данные/Сводная таблица**. На экране появится первое диалоговое окно **Мастера сводных таблиц и диаграмм – шаг 1 из 3**.

- Шаг 1. Выбрать вариант создания таблицы на основе данных, находящихся в списке Microsoft Excel. Вид создаваемого отчета – сводная таблица.
- Шаг 2. Выбрать диапазон таблицы, если он не указан.
- Шаг 3. Выбрать будущее расположение сводной таблицы – новый лист (рис. 39). Однако перед тем как нажать кнопку **Готово**, надо нажать кнопку **Макет...** (рис. 40). Перетягиваем мышкой поле **Дата реализации** в отдельно стоящую область **Страница**, поле **Наименование товара** – в область **Строка**, поле **Наименование магазина** – в область **Столбец**, а поле **Выручка** – в поле **Данные**.

**Замечание:** Для поля **Выручка** автоматически выбирается тип операции – Сумма (т.к. формат данного поля – числовой).

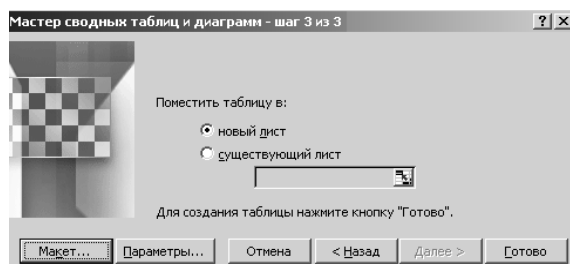


Рисунок 39 – Третий шаг построения сводной таблицы

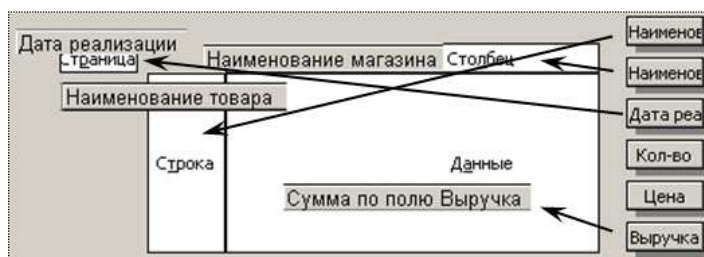


Рисунок 40 – Настройка областей сводной таблицы

По завершению третьего шага на новом листе получается результат, представленный на рисунке 41.

	А	В	С	Д	Е	Ф
1	Дата реализации	(Все)				
2						
3	Сумма по полю Выручка	Наименование магазина				
4	Наименование товара	1000 мелочей	тд Граффи	тд Немига	ЦУМ	Общий итог
5	Варочная панель			710000	1230000	1940000
6	Газовая плита		1340000	1350000	1340000	4030000
7	Духовой шкаф	1260000	1000000		990000	3250000
8	Микроволновая печь	2810000	1350000	450000	1400000	6010000
9	Портативная газ/плита	1850000	1170000	1350000	1280000	5650000
10	Соковыжималка				690000	690000
11	Электрическая плита			1250000	2610000	3860000
12	Общий итог	5920000	4860000	5110000	9540000	25430000

Рисунок 41 – Сводная таблица

Для создания сводной таблицы в Excel 2007+ необходимо выделить источник данных и выбрать команду **Вставка/Таблицы/Сводная таблица**.

В диалоговом окне **Создание сводной таблицы** необходимо проверить правильность выбора источника данных и указать лист расположения сводной таблицы.

На новом листе сформируется пустой шаблон сводной таблицы, а справа окно для ее настройки (рис. 42).

Далее повторяются те же действия, что и Excel 2003. В объединенную группу ячеек на рабочем листе с текстом «Перетащите сюда поля страниц» перетягивается мышкой поле **Дата реализации**. Далее перетягиваются: поле **Наименование товара** – в область **Названия строк**, поле **Наименование магазина** – в область **Наименование столбцов**, а поле **Выручка** – в поле **Значения**. При этом, в области **Значения** для поля **Выручка** автоматически устанавливается тип операции – **Сумма**. Если для поля **Выручка** нужна иная операция, тогда, щелкнув по полю **Сумма по полю Выручка**, выбирается команда **Параметры полей значений**. В открывшемся диалоговом окне выбирается из списка необходимая операция.

Вид и форма сводной таблицы изменяется на рабочем листе автоматически после выбора любого параметра в окне **Список полей сводной таблицы**.

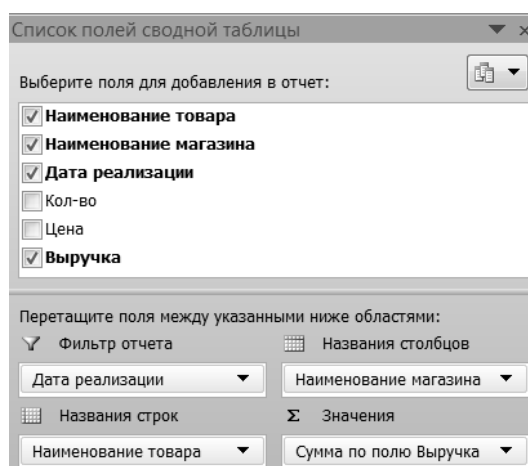


Рисунок 42 – Настройка полей сводной таблицы

## СИСТЕМА КОМПЬЮТЕРНОЙ АЛГЕБРЫ MATHCAD

MathCAD является математическим редактором, позволяющим проводить разнообразные научные и инженерные расчеты, начиная от элементарной арифметики и заканчивая сложными реализациями численных методов. Отличительной чертой MathCAD от большинства других современных математических приложений является его построение по принципу "что вы видите, то и получите".

Важно помнить, что по умолчанию вычисления в документе производятся в режиме реального времени (сверху вниз и слева на право), т. е. как только пользователь вводит в формулу оператор численного или символического равенства, MathCAD пытается вычислить это выражение (и все остальные формулы, находящиеся ниже по тексту).

### Панель инструментов Математика

Помимо элементов управления, характерных для типичного текстового редактора, MathCAD снабжен дополнительными средствами для ввода и редактирования математических символов, одним из которых является отображенная на рисунке 43 панель инструментов **Math** (Математика) – **View/Toolbars** (Вид/Панели инструментов).

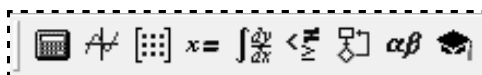


Рисунок 43 – Панель инструментов Математика


Панель **Математика** предназначена для вызова на экран еще девяти панелей, с помощью которых, собственно, и происходит вставка математических операций в документы. Чтобы вызвать какую-либо из них, нужно нажать соответствующую кнопку на панели **Математика**.




Перечислим назначение математических панелей:

- **Calculator** (Калькулятор) – служит для вставки основных математических операций, получила свое название из-за схожести набора кнопок с кнопками типичного калькулятора;
- **Graph** (График) – для вставки графиков;
- **Matrix** (Матрица) – для вставки матриц и матричных операторов;
- **Evaluation** (Выражения) – для вставки операторов управления вычислениями;
- **Calculus** (Вычисления) – для вставки операторов интегрирования, дифференцирования, суммирования;
- **Boolean** (Булевы операторы) – для вставки логических (булевых) операторов;
- **Programming** (Программирование) – для программирования средствами MathCAD;
- **Greek** (Греческие символы) – для вставки греческих символов;
- **Symbolic** (Символика) – для вставки символьных операторов.

### Ввод и редактирование выражений

Во время ввода и редактирования выражений необходимо обращать внимание на следующие элементы интерфейса MathCAD:


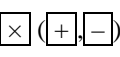
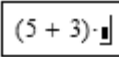


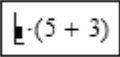







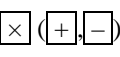
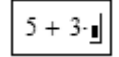


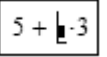






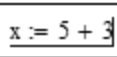
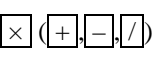
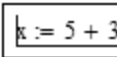
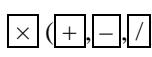
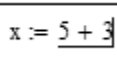
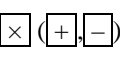
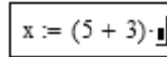


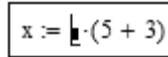
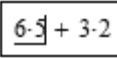

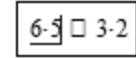
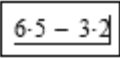

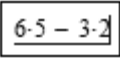
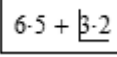

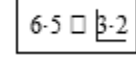
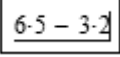

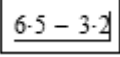
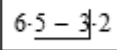

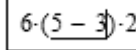
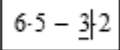

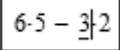
-  – *визир* – крестик красного цвета, который отмечает пустое место в документе, куда можно вводить текст или формулу;

-  – курсор ввода – горизонтальная и вертикальная линии синего цвета, выделяющие в тексте или формуле определенную часть;
-  – маркер символа – черный прямоугольник;
-  – маркер оператора – черная прямоугольная рамка.

Ввести математическое выражение можно в любом пустом месте документа MathCAD. Для этого поместите визир в желаемое место документа, щелкнув в нем мышью, и просто начинайте вводить выражение, нажимая клавиши на клавиатуре. При этом в документе создается математическая область, которая обрамляется тонкой сплошной черной линией.

Покажем некоторые ситуации при вводе выражений.

Таблица 4 – Последовательность операций при вводе и редактировании математических выражений

Исходная ситуация	Операнд	Результат	Исходная ситуация	Операнд	Результат
					
					
					
					
		нет			нет
					
					
					
					

При вводе выражения нажатие клавиши <Ins> меняет направление ввода символов слева направо (уголок курсора ввода справа) или справа налево (уголок курсора ввода слева). Попеременное нажатие на клавишу <Space> приводит к изменению объема обрамленной части выражения курсором ввода.

### Операторы численного и символьного вывода

Для того чтобы показать численное значение выражения необходимо:

- Ввести выражение;
- Нажать знак численного равенства « $\Rightarrow$ » (панель **Калькулятор**).

При нажатии клавиши  $\Rightarrow$  знак численного равенства появится в конце выражения вне зависимости от того где в этот момент в выражении находился курсор ввода. В случае присутствия в выражении неизвестных величин на экране появится сообщение об ошибке (рис. 44).

Для того чтобы показать аналитическое (символьное) значение выражения необходимо:

- Ввести выражение;
- Нажать знак символьного равенства « $\rightarrow$ » (панель **Вычисления**).

При нажатии комбинации клавиш <Ctrl>+<. > знак символьного равенства появится в конце выражения вне зависимости от того где в этот момент в выражении находился курсор ввода.

Символьное вычисление используется при определении интеграла или производной, элементы которых находятся на панели **Вычисления**, а также других операций, операторы которых находятся на панели **Символика**. Один из них, например оператор **Solve**, находит корни уравнения в символьном виде (рис. 45).

Рисунок 44 – Примеры использования операторов вывода

Рисунок 45 – Примеры символьного вывода

## Форматирование результатов

MathCAD вычисляет все выражения с точностью до 20 знаков, но выводит на экран не все значащие цифры.

Установив указатель мыши на нужном численном результате расчета, откройте окно форматирования чисел **Result/Format** (Формат/Результат) на закладке **Number Format** (Формат числа). В этом окне можно выбрать следующие форматы (рис. 46):

- **General** (Основной) – принят по умолчанию. Число знаков после запятой определяется в пункте **Number of decimal pieces** (Число десятичных знаков).
- **Decimal** (Десятичный) – десятичное представление чисел с плавающей точкой.
- **Scientific** (Научный) – числа отображаются только с порядком.
- **Engineering** (Инженерный) – числа отображаются только с порядком, кратным 3.
- **Fraction** (Дробный) – числа отображаются в виде правильной или неправильной дроби.

Рисунок 46 – Примеры форматов численных результатов

Можно установить выбранный формат для всех результатов в документе. Для этого необходимо перед настройкой формата визир установить на свободном месте документа. Важно помнить, что выбранный формат применится только для тех результатов, для которых не применялось форматирование, описанное предыдущим способом.

## Присвоение значений переменным

Значения переменным присваиваются с помощью оператора присваивания «:=», по следующему правилу:

**имя переменной := выражение**

Оператор «:=» удобнее набирать с клавиатуры, нажав сочетание клавиш <Shift> + <:=>.

Имена переменных могут состоять из латинских, русских, греческих и других букв и цифр, знаков подчеркивания «\_», штриха «'», символа процента «%», вводимых с клавиатуры (рис. 47).

Рисунок 47 – Примеры использования оператора присваивания

Имена переменных не могут начинаться с цифры, знака подчеркивания, штриха, символа процента, не могут включать в себя пробелы.

Если вместо непрерывной переменной используется ряд чисел, выстроенных в порядке возрастания или убывания, такая переменная называется дискретной. Определение дискретной переменной имеет вид:

**имя переменной := первое значение . . последнее значение**

**имя переменной := первое значение, второе значение . . последнее значение**

Дискретная переменная (примеры использования которой приведены на рисунке 48) может задавать как целые, так и дробные значения переменной, но обязательно равноотстоящие друг от друга.

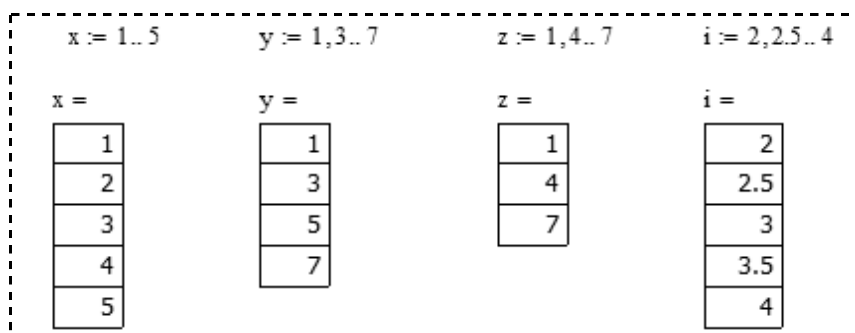


Рисунок 48 – Примеры создания дискретной переменной

Двоеточие – знак диапазона нельзя набирать с клавиатуры, нажимая два раза клавишу <.> (точка). Надо на клавиатуре нажать клавишу <.> (точка с запятой) или в математической панели **Матрица** выбрать значок  $m..n$ .

### Встроенные функций и функции пользователя

MathCAD позволяет использовать в расчетах как встроенные функции, так и функции, создаваемые пользователем.

Получить доступ к встроенным функциям MathCAD можно путем нажатия на стандартной панели кнопки  $f(x)$ . В левой части открывшегося окна находятся группы (категории) функций. Щелкнув мышью на любой из групп функций, вы увидите справа перечень функций, входящих в эту группу.

Вставить встроенную функцию в выражение можно через панель **Калькулятор**, через главное меню **Вставка/Функция** или набрать вручную с клавиатуры.

Функция пользователя создается вручную. При этом необходимо придерживаться следующего правила:

**имя функции(аргументы) := выражение**

$$f(x) := \sin(x) + 3 \quad w_1(x,y) := x^2 + 5 \quad y12(a,b,c) := 2 \cdot a^2 + 5 \cdot b + c$$

Рисунок 49 – Примеры создания пользовательских функций

Имена функций создаются по тем же правилам, что и имена переменных.

Функции пользователя можно создавать как на основе встроенных функций, так и уже созданных пользовательских функций либо их комбинаций (рис. 49).

Например, для создания разветвляющихся функций используется встроенная функция «if» (аналог функции ЕСЛИ() – Excel) (таблица 5).

Таблица 5 – Примеры создания разветвляющихся функций с помощью встроенной функции «if»

Разветвляющаяся функция	Листинг в MathCAD
$f(x) = \begin{cases} 2 \cdot x, & x < 0 \\ x^2, & x \geq 0 \end{cases}$	$f(x) := \text{if}(x < 0, 2 \cdot x, x^2)$
$f(x) = \begin{cases} 2 \cdot x, & x < 0 \\ x^2, & 0 \leq x \leq 3 \\ e^{2 \cdot x}, & x > 3 \end{cases}$	$f(x) := \text{if}(x < 0, 2 \cdot x, \text{if}(x \leq 3, x^2, e^{2 \cdot x}))$

### Построение плоского графика

Для построения плоского графика функции следует:

- установить визир в то место, где должен появиться график;
- на математической панели щелкнуть мышью по кнопке **График**. На панели **График – X-Y Plot** (График X-Y);

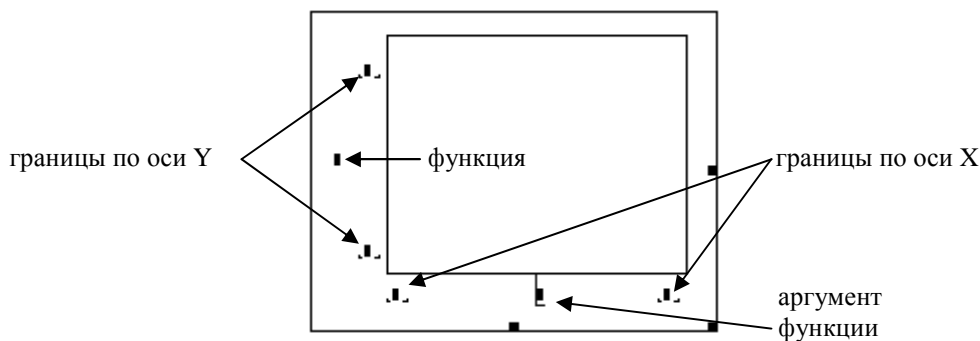


Рисунок 50 – Шаблон для построения плоского графика функции

- в появившемся на месте визира шаблоне плоского графика (рис. 50) ввести на оси абсцисс имя аргумента, на оси ординат – имя функции;
- щелкнуть мышью вне шаблона графика.

Если диапазон значений аргумента не задан, по умолчанию график будет построен в диапазоне значений аргумента от  $-10$  до  $10$ .

Если ввести на осях ординат и абсцисс имена двух функций одного аргумента, например  $x(t)$  и  $y(t)$ , то будет отображен параметрический график функции.

Чтобы на одном шаблоне разместить несколько графиков, надо, набрав на оси ординат имя первой функции, нажать клавишу запятой (уголок курсора при этом обязательно должен находиться в конце имени функции). В появившемся маркере впишите имя второй функции и т. д.

Если две функции имеют разные аргументы, например,  $f1(x)$  и  $f2(y)$ , то на оси ординат нужно ввести (через запятую) имена обеих функций, а на оси абсцисс (также через запятую) – имена обоих аргументов,  $x$  и  $y$ . Тогда первый график будет построен для первой функции по первому аргументу, а второй график – для второй функции по второму аргументу.

Покажем пример построения график функции одной переменной  $\sin(x)$  в диапазоне  $x$  от  $0$  до  $4$  и дополнительно на том же шаблоне отобразим две точки принадлежащие функции  $\sin(x)$  при  $x=2$  и  $x=3$ .

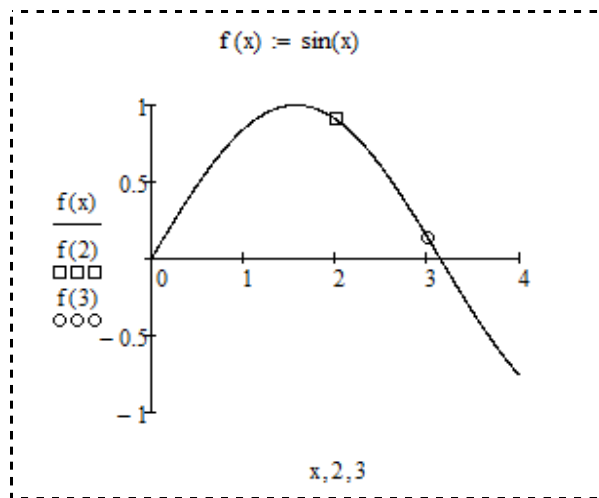


Рисунок 51 – Пример создания нескольких функций на одном шаблоне

При реализации такого примера (рис. 51) необходимо изменить настройки графиков функций. Для этого сделайте двойной щелчок мышью в поле шаблона – откроется окно форматирования графиков.

На первой закладке **Оси X, Y** выберите опцию **Отображение осей** – по центру.

На второй закладке **Трассировка** для трех функций установите следующие настройки, показанные на рисунке 52:

Обозначение в легенде	Частота символов	Символ	Ширина символа	Линия	Толщина линии	Цвет	Тип	Ось Y
кривая 1	1		1	—	1		линии	Y
кривая 2	1	□	1		1		точки	Y
кривая 3	1	○	1		1		точки	Y

Рисунок 52 – Фрагмент закладки **Трассировка** диалогового окна **Форматирования графиков**

## Действия над матрицами

Чтобы определить вектор или матрицу, следует:

- записать имя матрицы, ввести оператор присваивания «:=»;
- на панели **Математика** нажать кнопку с изображением матрицы. Откроется панель **Матрица**, на которой нужно вновь нажать кнопку с изображением матрицы. На этот раз откроется диалоговое окно, в котором надо ввести число строк и число столбцов матрицы и нажать кнопку **ОК**. На экране появится шаблон матрицы. То же действие вызывается нажатием комбинации клавиш <Ctrl>+<m>;
- каждый маркер символа в шаблоне заполнить числами или буквенными выражениями.

Доступ к любому элементу матрицы можно получить, задав имя матрицы с двумя индексами. Первый индекс обозначает номер строки, второй – номер столбца. Произвольный элемент вектора задается одним индексом.

$$\begin{array}{l}
 i := 0..2 \quad j := 0..2 \\
 A_{0,j} := 3 \quad A = (3 \ 3 \ 3) \quad B_{i,i} := 2 \quad B = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} \\
 A_{1,j} := 2 \quad A = \begin{pmatrix} 3 & 3 & 3 \\ 2 & 2 & 2 \end{pmatrix} \quad B_{i,j} := 3 \quad B = \begin{pmatrix} 3 & 3 & 3 \\ 3 & 3 & 3 \\ 3 & 3 & 3 \end{pmatrix} \\
 A_{2,j} := 1 \quad A = \begin{pmatrix} 3 & 3 & 3 \\ 2 & 2 & 2 \\ 1 & 1 & 1 \end{pmatrix} \quad B_{i,i} := 2 \quad B = \begin{pmatrix} 2 & 3 & 3 \\ 3 & 2 & 3 \\ 3 & 3 & 2 \end{pmatrix} \\
 A_{i,1} := 4 \quad A = \begin{pmatrix} 3 & 4 & 3 \\ 2 & 4 & 2 \\ 1 & 4 & 1 \end{pmatrix}
 \end{array}$$

Рисунок 53 – Пример использования индексов массивов

Для набора нижнего индекса можно щелкнуть на кнопке **Subscript** (Индекс) на панели **Матрица**, но лучше использовать клавишу <[>, так как при работе с матрицами ставить нижний индекс приходится очень часто.

Нумерация элементов массива (вектора или матрицы) может начинаться с 0, 1 или с любого другого числа (положительного или отрицательного). Порядком нумерации элементов массива управляет встроенная переменная **ORIGIN**. По умолчанию **ORIGIN** = 0. Это означает, что первый элемент массива имеет номер 0.

Чтобы нумерация элементов векторов и матриц начиналась, как обычно принимается в математике, с 1, нужно перед вводом матрицы, а лучше в начале документа, напечатать **ORIGIN:=1** (все буквы прописные).

$$\begin{array}{l}
 A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad \text{ORIGIN} := 0 \quad A_{1,1} = 5 \quad A_{2,1} = 8 \\
 \text{ORIGIN} := 1 \quad A_{1,1} = 1 \quad A_{2,1} = 4
 \end{array}$$

Рисунок 54 – Пример использования системной переменной **ORIGIN**

Операции сложения, вычитания и умножения выполняются тем же способом, что и над обычными переменными. Для операций транспонирования, обращения и вычисления определителя матрицы есть специальные элементы, расположенные на панели **Матрица**.

Используя символ **Столбец матрицы** можно получить доступ к указанному столбцу массива. Чтобы получить доступ к строке матрицы необходимо предварительно транспонировать матрицу, а затем использовать символ **Столбец матрицы**.

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad A^{<1>} = \begin{pmatrix} 2 \\ 5 \\ 8 \end{pmatrix} \quad (A^T)^{<1>} = \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix} \quad (A^T)^{<1>T} = (4 \ 5 \ 6)$$

Рисунок 55 – Пример использования символа **Столбец матрицы** с панели **Матрица**

## Функции для работы с массивами

Таблица 6 – Примеры использования встроенных функций при работе с массивами

Функция	Описание	Пример (листинг MathCAD)
augment(A,B)	формирует матрицу, где в первых столбцах располагаются элементы матрицы A, а в последних матрицы B	$A := \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad B := \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix} \quad \text{stack}(A,B) = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{pmatrix}$ $\text{augment}(A,B) = \begin{pmatrix} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \end{pmatrix}$
stack(A,B)	формирует матрицу, где в первых строках располагаются элементы матрицы A, а в последних матрицы B	
submatrix(A,ir,jr,ic,jc)	формирует матрицу, которая является блоком матрицы A, расположенным в строках от ir до jr и столбцах ic от jc	$A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad \text{ORIGIN} := 1$ $\text{submatrix}(A, 1, 2, 1, 2) = \begin{pmatrix} 1 & 2 \\ 4 & 5 \end{pmatrix}$ $\text{min}(A) = 1 \quad \text{submatrix}(A, 3, 3, 1, 2) = (7 \ 8)$ $\text{max}(A) = 9 \quad \text{submatrix}(A, 1, 3, 2, 2) = \begin{pmatrix} 2 \\ 5 \\ 8 \end{pmatrix}$ $\text{mean}(A) = 5 \quad \text{submatrix}(A, 1, 3, 2, 2) = \begin{pmatrix} 2 \\ 5 \\ 8 \end{pmatrix}$ $\text{rows}(A) = 3 \quad \sum_{i=1}^3 A_{i,1} = 12 \quad \prod_{j=1}^3 A_{2,j} = 120$ $\text{cols}(A) = 3$
min(A)	определяет минимальный элемент массива	
max(A)	определяет максимальный элемент массива	
mean(A)	определяет среднее арифметическое элементов массива	
rows(A), cols(A)	определяет количество строк или столбцов массива	
$\sum_{i=0}^n$ $\prod_{k=1}^n$	определяет сумму или произведение n элементов аргумента	

## Решение систем линейных алгебраических уравнений (СЛАУ)

Последовательность действий при решении СЛАУ с помощью блока **Given..Find**:

- задать начальные приближения для всех неизвестных, входящих в систему;
- напечатать ключевое слово **Given** (Дано). Убедитесь, что при печати вы не находитесь в текстовой области. Если нажать клавишу пробела, то математическое выражение становится текстовой областью и слово **Given** перестает восприниматься как ключевое;
- ввести уравнения и неравенства. Между левой и правой частями уравнения должен стоять знак логического равенства. Ввести его можно с панели **Булевы операторы** либо используя комбинацию клавиш <Ctrl>+<=>;
- введите любое выражение, содержащее функцию **Find**.

Пример:

Условие	Решение (листинг MathCAD)
$\begin{cases} 2 \cdot x + 3 \cdot y = 15 \\ -9 \cdot x + 6 \cdot y = 18 \end{cases}$	$x := 1 \quad y := 1$ <p><b>Given</b></p> $2 \cdot x + 3 \cdot y = 15 \quad -9 \cdot x + 6 \cdot y = 18$ $\begin{pmatrix} x \\ y \end{pmatrix} := \text{find}(x, y) \quad \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0.923 \\ 4.385 \end{pmatrix}$

Последовательность действий при решении СЛАУ с помощью обратной матрицы (рис. 56) совпадает с последовательностью действий при решении СЛАУ в Excel.



$$A := \begin{pmatrix} 2 & 3 \\ -9 & 6 \end{pmatrix} \quad b := \begin{pmatrix} 15 \\ 18 \end{pmatrix}$$

$$|A| = 39 \quad \begin{pmatrix} x \\ y \end{pmatrix} := A^{-1} \cdot b \quad \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0.923 \\ 4.385 \end{pmatrix}$$

Рисунок 56 – Пример решения СЛАУ с помощью обратной матрицы

### Определение нулей и экстремумов функции

Для нахождения нулей функции в MathCAD используется функция **root()**:

**Root(функция, начальное приближение [ , начало отрезка, конец отрезка] )**

Перед началом решения желательно построить график функции, чтобы проверить, есть ли корни, т.е. пересекает ли график ось абсцисс. Начальное приближение лучше всего выбрать по графику поближе к значению корня.

В функции **root()** достаточно использовать два первых параметра. В этом случае необходимо предварительно задавать начальное приближение.

Если указывать в функции **root()** все четыре параметра, тогда начальное приближение предварительно можно не задавать. Необходимо помнить, что внутри указанного интервала должен быть только один нуль функции.

$$f(x) := \sin(x)$$

$$x := 3$$

$$\text{root}(f(x), x) = 3.142 \quad \text{или} \quad \text{root}(f(x), x, 3, 4) = 3.142 \quad \text{или} \quad \text{root}(f(x), x, 3, 4) \rightarrow \pi$$

Рисунок 57 – Пример использования функции **root()**

Для нахождения локальных экстремумов в MathCAD используются функции **maximize()** и **minimize()**:

**maximize(имя функции, аргументы)**

**minimize(имя функции, аргументы)**

Использование функций **maximize()** и **minimize()** аналогично функции **find()**.

Ключевое слово **Given** можно опустить – оно необходимо только при наличии ограничений (рис. 58).

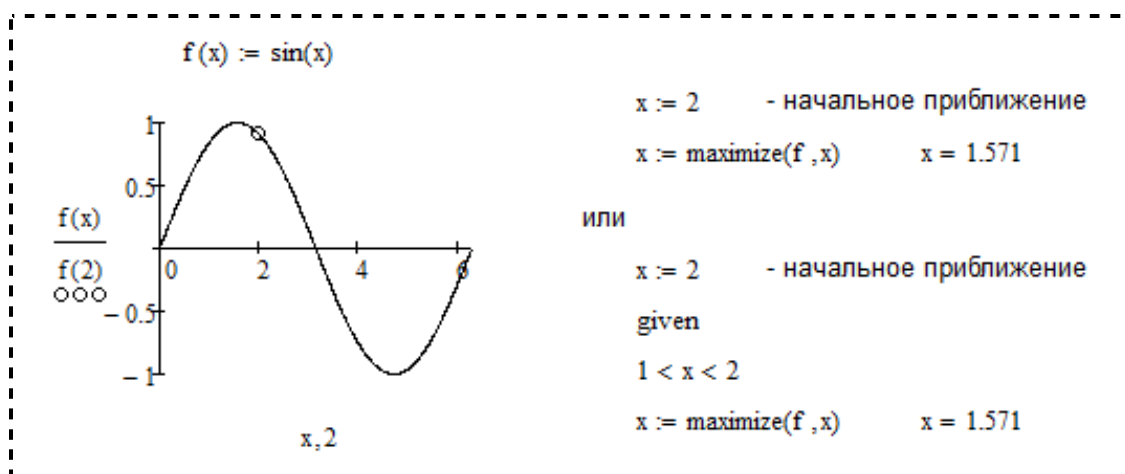


Рисунок 58 – Пример использования функции **maximize()**

Visual Basic for Applications (VBA) – язык программирования, встроенный в линейку приложений Microsoft Office и предназначенный для их автоматизации. Благодаря его сравнительной лёгкости освоения, офисные приложения могут создавать даже пользователи, не программирующие профессионально.

Обычно программный код VBA организуется в виде совокупности процедур (макросов), каждая из которых предназначена для решения определенной задачи. Наряду с процедурами, в языке VBA также могут быть созданы и собственные функции, которые затем будут использоваться в процедурах.

### Безопасность макросов

Чтобы включить макросы в Excel 2003, необходимо выбрать пункт главного меню **Сервис/Макрос/Безопасность**. В открывшемся окне **Безопасность** установить переключатель **Уровень безопасности** в положение **Средняя** или **Низкая**.

В Excel 2007+ параметры безопасности макросов можно изменять так:

- На вкладке **Разработчик** в группе **Код** нажать кнопку **Безопасность макросов**. Если вкладка **Разработчик** не отображается, необходимо нажать кнопку **Microsoft Office**, щелкнуть **Параметры Excel**, а затем в категории **Основные** в разделе **Основные параметры работы с Excel** выбрать параметр **Показывать вкладку «Разработчик»** на ленте.

- В категории **Параметры макросов** в разделе **Параметры макросов** выбрать вариант **Отключить все макросы с уведомлением** (аналог Excel 2003 – **Средняя**) или **Включить все макросы** (аналог Excel 2003 – **Низкая**).

Если установлена опция **Средняя** (Excel 2003), то при открытии файла в диалоговом окне необходимо нажать кнопку **Не отключать макросы**.

Если установлена опция **Отключить все макросы с уведомлением** (Excel 2007+), то при открытии рабочих книг, содержащих макросы, макросы отключаются и Excel выводит предупреждение системы безопасности о том, что макрос отключен. Если есть необходимость включить макросы, то надо щелкнуть сначала на кнопке **Параметры** в строке предупреждения. Затем в открывшемся диалоговом окне **Параметры безопасности Microsoft Office** установить переключатель **Включить это содержимое** и щелкнуть на кнопке **ОК**.

Очень важно после этого закрыть и снова открыть данный файл. Без этого действия макросы будут по-прежнему отключены. Установленный уровень безопасности распространяется на все файлы Excel.

### Открытие и сохранение макросов VBA

Для открытия редактора макросов VBA в Excel 2003 необходимо выбрать пункт меню **Сервис/Макрос/Редактор Visual Basic**, в Excel 2007+: вкладку **Разработчик** в группе **Код** команду **Visual Basic**. В обеих системах эти действия можно заменить комбинацией клавиш **Alt+F11**.

Рабочие книги Excel 2003, содержащие макросы, сохраняются в файлах с расширением **XLS**, а рабочие книги Excel 2007+: в файлах с расширением **XLSM**. Если такая рабочая книга Excel 2007+ сохраняется в формате **XLSX**, заданном по умолчанию, то Excel сохранит рабочую книгу без макросов. Если надо сохранить макросы, то нажимается кнопка **Microsoft Office**, выбирается пункт **Сохранить как** и в диалоговом окне **Сохранение документа** выбирается тип файла **Книга Excel с поддержкой макросов (\*.xlsm)**.

### Создание процедур и макросов

Перед тем как вводить код программы, необходимо вставить модуль в рабочую книгу. Если рабочая книга уже имеет лист модуля, его можно использовать для нового макроса.

Для вставки нового модуля, надо выполнить следующие действия:

- выбрать в окне **Project** рабочую книгу, которая используется в текущий момент;
- выбрать команду **Insert/Module**. В рабочей книге появится новый (пустой) модуль.

Каждый элемент в окне **Project** имеет определенное количество свойств. Для модификации свойств используется окно **Properties**. Чтобы отобразить окно **Properties**, выбирается команда **View / Properties Window** или нажимается клавиша **F4**. В окне **Properties** отображается список свойств выбранного элемента.

Для изменения имени модуля необходимо в окне **Project** выбрать нужный модуль, а затем в окне **Properties** в свойстве **Name** указать его новое имя. Чтобы удалить ненужный модуль, нужно щелкнуть по нему правой клавишей мыши в окне **Project** и в контекстном меню выбрать пункт **Remove Module...** Далее в диалоговом окне нажать на кнопку **Нет**.

Процедура может быть двух типов: **подпрограммой** и **функцией**.

**Процедура-подпрограмма** – это нечто вроде новой команды, которая может быть выполнена либо пользователем, либо другим макросом. В рабочей книге Excel может содержаться произвольное число подпрограмм.

Подпрограммы всегда начинаются с ключевого слова **Sub**, после которого следует имя макроса (у каждого макроса должно быть уникальное имя), а затем – пара круглых скобок. (В этих скобках задаются аргументы, но

если у подпрограммы нет аргументов, они остаются пустыми.) Оператор **End Sub** говорит об окончании подпрограммы. Строки, заключенные между этими двумя операторами, составляют тело процедуры или текст макроса.

Вторым типом процедуры VBA является **процедура-функция**. Функция всегда возвращает единственное значение (так же, как и обычная функция рабочей таблицы), которое внутри процедуры присваивается имени функции. Функции подобны подпрограммам, но начинаются ключевым словом **Function** и заканчиваются оператором **End Function**.

Таблица 7 – Пример создания процедур в VBA

Тип процедуры	Синтаксис	Пример
Функция	[Public   Private] <b>Function</b> имя ([список_арг]) [As тип] [операторы] имя = выражение End Function	Public Function Fun_F(x) 'Процедура функции  Fun_F = 2 * x + 3 End Function
Подпрограмма	[Public   Private] <b>Sub</b> имя ([список_арг]) [операторы] End Sub	Public Sub Tab_F() 'Процедура подпрограммы  End Sub

**Замечание:** Служебные слова Private и Public задают область видимости процедур и функций:

- **Private** делает объект доступным только внутри данного модуля;
- **Public** делает объект доступным из другого модуля.

Примеры процедур, показанные в таблице 7, содержат комментарии (например, «Процедура функции»). Комментарии – это заметки для того, кто составляет код процедуры, VBA их игнорирует. Строка комментариев начинается с апострофа. Комментарий можно поместить после любого оператора. Другими словами, если VBA встречает апостроф, он игнорирует остальной текст этой строки.

Создать шаблон процедуры типа **Sub** или **Function** можно автоматически:

- установить курсор в окне **Code**;
- выбрать пункт меню **Insert/Procedure...**;
- в появившемся диалоговом окне **Add Procedure** в поле **Name** указать имя процедуры (без пробелов и скобок), а в поле **Type** указать тип процедуры **Sub** (рис. 59) или **Function** (рис. 60).

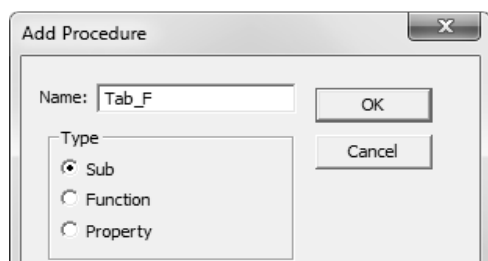


Рисунок 59 – Форма для создания процедуры **Sub**

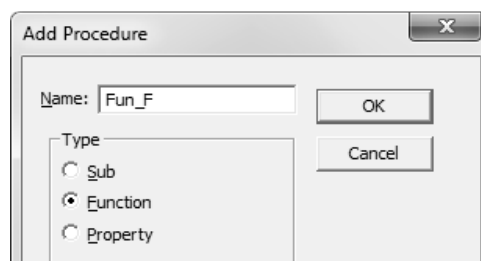



Рисунок 60 – Форма для создания процедуры **Function**

В процессе работы с MS Excel часто приходится многократно выполнять однотипные действия. В этом случае разумно создавать макросы с помощью макрорекордера (Macro Recorder), т.е. специального средства для записи макросов. При этом все необходимые действия записываются программой записи макроса и преобразуются автоматически в команды языка VBA. Работа макрорекордера во многом напоминает запись с помощью обычного магнитофона (диктофона). При этом перемещение курсора по ленте и рабочей книге не включается в записанные команды. Перед созданием макроса с помощью макрорекордера важно знать:

- Все шаги и команды, выполняемые макросом, должны быть спланированы перед записью или написанием макроса. Если при записи макроса была допущена ошибка, сделанные исправления также будут записаны.
- VBA хранит каждый записанный макрос в отдельном модуле, присоединенном к книге.

Порядок записи макроса:

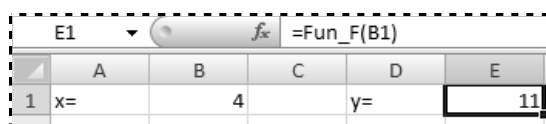
1. Выбрать команду **Начать запись** (в Excel 2003: п.м. **Сервис/Макрос**, в Excel 2007+: вкладка **Разработчик** в группе **Код**).
2. В появившемся диалоговом окне **Запись макроса**:
  - в поле **Имя макроса** ввести имя для макроса, которое не должно начинаться с цифры, не должно иметь пробелов и символов пунктуации;

- в поле Сочетание клавиш для того чтобы запускать макрос с помощью сочетания клавиш, ввести букву (допускается использование сочетаний CTRL+ буква (для строчных букв) или CTRL+SHIFT+буква (для прописных букв), где буква – любая буквенная клавиша на клавиатуре);
  - в поле Сохранить в: выбрать книгу, в которой требуется сохранить макрос (Личная книга макросов, Новая книга, Эта книга);
  - в поле Описание можно, но необязательно, ввести необходимый текст для краткого описание макроса.
3. После нажатия кнопки ОК, Excel запустит макрорекордер и отобразит панель "Остановить запись".
  4. Выполнить макрокоманды, которые нужно записать (для того чтобы прервать выполнение макроса, надо нажать кнопку ESC).
  5. Нажать кнопку  на панели инструментов "Остановить запись".

## Выполнение процедур и макросов

Перед использованием *процедуры-функции* обязательно нужно задать в скобках значения всех аргументов. Так, функцию VBA (Fun\_F(x)) в таблице 7) можно использовать:


- вызвав из другой процедуры VBA (например,  $y = \text{Fun\_F}(4)$ );
- ввести в ячейку рабочего листа точно так же, как и любую встроенную функцию Excel, только находящуюся в категории "Определенные пользователем" (рис. 61);
- вызвав из окна **Immediate**.



	A	B	C	D	E
1	x=	4		y=	11

Рисунок 61 – Использование функции Fun\_F(x) на рабочем листе

*Процедуры-подпрограммы* запускаются другими способами:

1. Запуск из рабочего листа Excel. В этом диалоговом окне **Макрос** выбрать из списка имя нужной подпрограммы и щелкнуть на кнопке **Выполнить**. Для открытия диалогового окна **Макрос** можно воспользоваться комбинацией клавиш **ALT + F8** или выполнить команду:
  - ✓ Excel 2003: **Макросы...** из пункта главного меню **Сервис/Макрос**.
  - ✓ Excel 2007+: **Макросы** на вкладке **Разработчик** в группе **Код**.
2. Запуск из окна **Code** редактора VBA. Поместить курсор где-нибудь в коде макроса и выполнить одно из предложенных действий:
  - ✓ выбрать пункт главного меню **Run/Run Sub**;
  - ✓ нажать на пиктограмму **Run Sub** панели инструментов **Standard** – ;
  - ✓ нажать клавишу **F5**.
3. Вызов из другой процедуры VBA (например, Call Tab\_F или просто Tab\_F).
4. Запуск из окна **Immediate** (см. ниже **Диалоговое окно Immediate**).
5. Процедуры (в том числе и макросы) можно закрепить:
  - ✓ за кнопкой на рабочем листе;
  - ✓ за пиктограммой на имеющейся панели инструментов или на новой (для Excel 2003);
  - ✓ за командой в любом пункте меню (для Excel 2003);
  - ✓ в виде кнопок на ленте или панели быстрого доступа (Excel 2007+).

## Пошаговое выполнение макросов

При отладке программы часто возникает необходимость проследить за тем, как выполняются отдельные операторы в отлаживаемой программе или же какие значения принимают те или иные переменные на различных этапах выполнения программы. Для достижения этих целей в редакторе VBA реализован режим прерывания.

Перевести программу в режим прерывания в некоторый, заранее известный момент ее выполнения можно с помощью установки **точки останова**. Чаще всего **точку останова** помещают в строке программного кода непосредственно перед оператором, который, по вашему предположению, является причиной возникновения ошибки. Как только по ходу выполнения программы достигается оператор, которому назначена **точка останова**, работа программы приостановится (шаг 3 таблицы 8). Теперь можно будет проверить текущие значения любых переменных (шаг 6 таблицы 8). Последующее нажатие **F8** позволяет выполнять процедуру по шагам, в порядке естественного выполнения операторов (шаги 4 и 5 таблицы 8). Если очередной оператор является вызовом процедуры типа **Sub** или **Function**, то применение **F8** приведет к открытию текста вызываемой процедуры в окне редактирования кода и позволит пройти эту процедуру по шагам, увидев результаты всех выполняемых в ней действий.

Для размещения или отмены точки останова в программном коде необходимо выполнить одно из следующих действий (шаг 2 таблицы 8):

- щелкнуть на серой полоске, расположенной в левой части окна редактирования кода, напротив требуемой строки программного кода;
- поместить курсор в нужную строку кода и нажать **F9**.

Таблица 8 – Элементы пошагового выполнения макроса

Шаг 1 – Создали процедуру My_T() типа Sub	Шаг 2 – Установили точку останова с помощью F9	Шаг 3 – Запустили процедуру с помощью F5
<pre>Public Sub My_T()   x = 3   y = 8 + x   Debug.Print y End Sub</pre>	<pre>Public Sub My_T()   x = 3   y = 8 + x   Debug.Print y End Sub</pre>	<pre>Public Sub My_T()   x = 3   y = 8 + x   Debug.Print y End Sub</pre>
Шаг 4 – Перешли к следующему оператору с помощью F8	Шаг 5 – Перешли к следующему оператору с помощью F8	Шаг 6 – Удерживаем указатель мыши над переменной y
<pre>Public Sub My_T()   x = 3   y = 8 + x   Debug.Print y End Sub</pre>	<pre>Public Sub My_T()   x = 3   y = 8 + x   Debug.Print y End Sub</pre>	<pre>Public Sub My_T()   x = 3   y = 8 + x   Debug.Print y End Sub</pre>

Количество размещаемых точек останова неограниченно, поэтому в разных строках программы можно разместить столько точек останова, сколько сочтете необходимым. Единственное условие – нельзя размещать точки останова в строках комментариев и строках с операторами, которые VBA на самом деле не выполняет, например, в строках с объявлениями переменных.

В редакторе VBA реализован механизм автоматической подсказки, которая позволяет увидеть текущее значение любой переменной в программе. Для этого, когда программа находится в режиме прерывания, необходимо задержать на секунду указатель мыши на имени интересующей вас переменной, и на экране появится окно подсказки – небольшой прямоугольник с именем и текущим значением переменной в нем (шаг 6 таблицы 8).

При выполнении макросов могут возникнуть ошибки, отличные от синтаксических, – ошибки выполнения. Такие ошибки можно обнаружить только в процессе выполнения процедуры. Обычно такие ошибки выполнения приводят к остановке процедур VBA. При этом отображается диалоговое окно с сообщением, в котором указывается номер ошибки и краткое ее описание (рис. 62).

В этом окне кнопка **End** прекращает выполнение процедуры. Щелчок на кнопке **Debug** переводит выполняющуюся процедуру в режим прерывания, что позволяет в окне кода редактора VBA внести исправления в тот оператор, где возникла данная ошибка выполнения, а затем продолжить либо завершить (**Run/Reset** или на **Reset** панели инструментов) выполнение программы.

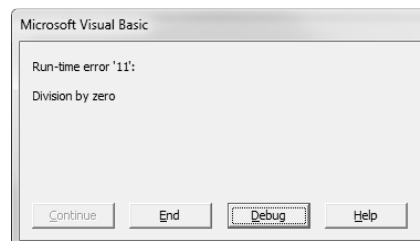


Рисунок 62 – Сообщение об ошибке

### Ввод-вывод данных на рабочий лист

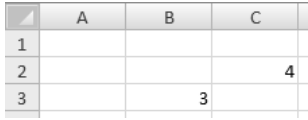
VBA – это объектно-ориентированный язык программирования. Это означает, что основными его элементами являются объекты – рабочие книги (**Workbook**), рабочие листы (**Worksheet**), ячейки или диапазоны ячеек (**Range, Cells**), диаграммы (**Chart**) и др. Эти объекты расположены в иерархическом порядке.

Набор однотипных объектов называется коллекцией. Например, коллекция всех объектов рабочих листов (**Worksheet**) называется **Worksheets**. Обратиться к отдельному объекту коллекции можно, используя порядковый номер или ссылки. Например, если в рабочей книге есть три рабочих листа: **Лист1**, **Лист2** и **Лист3**, то к объекту коллекции рабочих листов **Лист2** можно обратиться так – **Worksheets("Лист2")**.

Для выбора объекта, с которым предстоит дальнейшая работа, необходимо использовать метод **Select**. Метод – это действие, примененное к объекту. Так, например, чтобы выбрать рабочий лист **Лист2** (сделать его активным), необходимо записать в макросе следующую команду – **Worksheets("Лист2").Select**.

Для обращения к определенной ячейке рабочего листа необходимо использовать свойства **Range** ("адрес ячейки") или **Cells**(номер строки ячейки, номер столбца ячейки). С помощью этих свойств можно передавать значения переменной в ячейку и из ячейки – переменной. Примеры приведены в таблице 9.

Таблица 9 – Примеры использования свойств **Range** и **Cells**

<i>Условие:</i>	Присвоить переменной <i>f</i> значение из ячейки <b>C2</b> рабочего листа	Вывести на рабочий лист в ячейку <b>B3</b> результат выражения $2^2-1$
	$f = \text{Range}("C2")$ или $f = \text{Cells}(2,3)$	$\text{Range}("B3") = 2^2 - 1$ или $\text{Cells}(3,2) = 2^2 - 1$
<i>Результат:</i>	$f = 4$	В ячейке <b>B3</b> число 3

При использовании в выражении без указания координат свойство **Cells** объекта **Worksheet** определяет диапазон, включающий все ячейки данного рабочего листа. Совместное действие **Cells** и метода **Clear** позволяет очистить содержимое всех ячеек активного рабочего листа. Например, **Cells.Clear**.

### Диалоговые окна

В VBA есть две функции – **InputBox** и **MsgBox**, которые позволяют отображать простые диалоговые окна. Эти диалоговые окна можно видоизменить несколькими способами, но, конечно, они не могут содержать всех тех опций, которые доступны в созданных пользователем диалоговых окнах.

В краткой форме функция VBA **InputBox**, предназначенная для ввода значений переменным, имеет следующий синтаксис:

`InputBox(сообщение [,заголовок] [,по_умолчанию])`

Назначение аргументов:

- ✓ *сообщение* – текст, отображаемый в окне (обязательный аргумент);
- ✓ *заголовок* – текст, который появляется в строке заголовка окна (необязательный аргумент);
- ✓ *по\_умолчанию* – значение, отображаемое в окне ввода по умолчанию (необязательный аргумент).

В следующем примере функция **InputBox** предлагает пользователю ввести значение переменной *x*.

`x = InputBox("введите значение x", "ввод значений")`

При выполнении этого оператора VBA Excel выводит на экран диалоговое окно, показанное на рисунке 63. Следует заметить, что в данном примере использованы только два первых аргумента, параметр *по\_умолчанию* не указан. Когда пользователь введет некоторое значение и щелкнет на кнопке ОК, это значение будет присвоено переменной *x*. Функция **InputBox** всегда возвращает строку (текст), поэтому может возникнуть необходимость в преобразовании результата в число. Для этого необходимо воспользоваться функцией **Val**, которая преобразует текст в число:

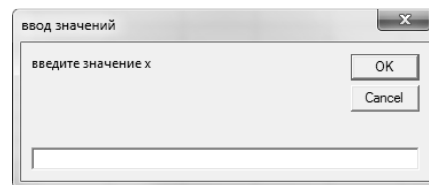


Рисунок 63 – Диалоговое окно функции **InputBox**

`x = Val(InputBox("введите значение x", "ввод значений"))`

Если в качестве параметра *по\_умолчанию* задать значение 3.5 (т.е.  $x=3.5$ ), то код будет выглядеть так:

`x = Val(InputBox("введите значение x", "ввод значений", "3.5"))`

Функция VBA **MsgBox** – это весьма удобное средство для того, чтобы отобразить на экране информацию и попросить пользователя сделать выбор, щелкнув на одной из предложенных кнопок. Эта функция используется для того, чтобы вывести значение переменной. Краткий синтаксис функции **MsgBox** выглядит следующим образом:

`MsgBox(сообщение [,кнопки] [,заголовок])`

Описание аргументов:





- ✓ *сообщения* – текст, отображаемый в окне сообщения (обязательный аргумент);
- ✓ *кнопки* – коды кнопок, которые отображаются в окне сообщения (необязательный аргумент);
- ✓ *заголовок* – текст, который появляется в строке заголовка окна сообщения (необязательный аргумент).

Функцию **MsgBox** можно вызывать в виде отдельного оператора, не присваивая ее результат какой-нибудь переменной. Если функция вызывается самостоятельно, то аргументы не нужно заключать в круглые скобки.

В следующем примере функция **MsgBox** выводит на экран сообщение о переменной  $x = 3.5$  (рис. 64):

`MsgBox "значение переменной x = " + Str(x)`

Таблица 10 – Константы в **MsgBox**

<i>Константа</i>	<i>Знак</i>
<code>vbCritical</code>	
<code>vbQuestion</code>	
<code>vbExclamation</code>	
<code>vbInformation</code>	

Здесь в тексте *сообщения* использованы знак «+» для соединения (склеивания) двух частей текста и функция **Str**, которая преобразует числовое значение переменной *x* в текст.

Настраивать окна сообщений можно с помощью соответствующих встроенных констант VBA, которые используются в качестве аргументов **MsgBox**.

Встроенные константы VBA, отвечающие за тип пиктограммы в диалоговом окне сообщений, перечислены в таблице 10.

В следующем примере в операторе **MsgBox** заданы все три аргумента:

`MsgBox "значение переменной" + vbCr + "x = " + Str(x), vbInformation, "Вывод результата"`

Результат приведенного оператора отображен на рисунке 65. В первом аргументе *сообщение* использована встроенная константа **vbCr**, которая позволяет разбить текст на две части и перенести вторую часть на следующую строку. Если в качестве аргумента используется текст, то он заключается в кавычки.

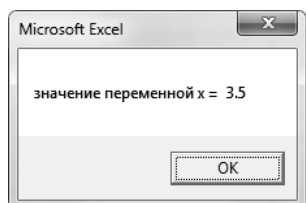


Рисунок 64 – Диалоговое окно оператора **MsgBox**

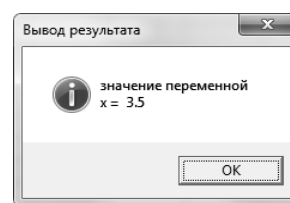


Рисунок 65 – Диалоговое окно оператора **MsgBox**

## Оператор **If**

Оператор **If...Then** – один из важнейших управляющих операторов. Он организует выполнение заданного блока операторов, если при вычислении указанного условного выражения будет получено значение ИСТИНА, и не делает ничего, если будет получено значение ЛОЖЬ. Такая конструкция **If...Then** записывается в одной строке, и завершающий оператор **End If** для нее не требуется (таблица 11).

Однако в операторе **If...Then** может выполняться не только один оператор, но и блок последовательно следующих операторов. В этом случае проверяемое условие и выполняемые операторы записываются в отдельных строках. Если условное выражение имеет значение ИСТИНА, выполняемая группа операторов начинается с первого оператора, стоящего после ключевого слова **Then**, и включает все последующие операторы, вплоть до оператора завершения **End If** (таблица 11).

Таблица 11 – Примеры краткой формы разветвляющей структуры

Блок-схема	Оператор	однострочный	многострочный
	<b>Синтаксис</b>	<code>If условие Then оператор</code>	<code>If условие Then операторы End If</code>
	<b>Пример</b>	<code>If x &gt; 5 Then MsgBox "x больше 5"</code>	<code>If x &gt; 5 Then MsgBox "x больше 5" End If</code>

Для более сложных ситуаций, когда на основании некоторого условия необходимо выбрать одну из двух различных последовательностей операторов, используется оператор **If...Then...Else**.

Как и оператор **If...Then**, конструкция **If...Then...Else** может быть однострочной (таблица 12) и многострочной (таблица 13).

Таблица 12 – Пример полной формы разветвляющей структуры (однострочный оператор **If**)

Блок-схема	Оператор	однострочный
	<b>Синтаксис</b>	<code>If условие Then оператор 1 Else оператор 2</code>
	<b>Пример</b>	<code>If x &gt; 5 Then MsgBox "x больше 5" Else MsgBox "x не больше 5"</code>

Таблица 13 – Пример полной формы разветвляющей структуры (многострочный оператор **If**)

Блок-схема	Оператор	Синтаксис	Пример
	многострочный	<pre>If условие Then     оператор 1 Else     оператор 2 End If</pre>	<pre>If x &gt; 5 Then     MsgBox "x больше 5" Else     MsgBox "x не больше 5" End If</pre>

Часто при написании программ возникают такие ситуации, когда приходится проверять не одно, а два или больше условий. В этом случае можно поместить один оператор **If...Then** внутри другого оператора **If...Then** (или оператор **If...Then...Else** внутри оператора **If...Then...Else**), что называется вложением операторов.

Вложенные операторы следует использовать тогда, когда для принятия решения необходимо проверить дополнительное условие, но при этом первое условие должно оказаться истинным. Если же первое условие не выполнено, то будут выполняться операторы, следующие за ключевым словом **Else**.

Таблица 14 – Пример многострочного оператора **If** с вложенным оператором **If**

Блок-схема	Синтаксис	Пример
	<pre>If условие 1 Then     оператор 1 Else     If условие 2 Then         оператор 2     [Else         оператор 3 ]     End If End If</pre>	<pre>If x &gt; 5 Then     MsgBox "x больше 5" Else     If x &lt; 5 Then         MsgBox "x меньше 5"     Else         MsgBox "x равно 5"     End If End If</pre>

Эквивалентом вложенных операторов является оператор **If...Then...ElseIf**, в котором используется ключевое слово **ElseIf**. С помощью ключевого слова **ElseIf** в одном условном операторе можно проверить несколько условий. Синтаксис этого условного оператора выглядит так, как показано в таблице 15.

Ключевое слово **Else** является необязательным, но если оно все же присутствует, то должно быть записано в конструкции оператора последним.

Таблица 15 – Пример многострочного расширенного оператора **If**

Блок-схема	Синтаксис	Пример
	<pre>If условие 1 Then     оператор 1 ElseIf условие 2 Then     оператор 2 [Else     оператор 3] End If</pre>	<pre>If x &gt; 5 Then     MsgBox "x больше 5" ElseIf x &lt; 5 Then     MsgBox "x меньше 5" Else     MsgBox "x равно 5" End If</pre>

Ключевое слово **Elseif** может повторяться в конструкции сколько угодно раз. При этом выполнение заданных условий проверяется строго последовательно, в том порядке, в каком они записаны. И как только истинное условие (ИСТИНА) будет найдено, выполняется соответствующий ему блок операторов, после чего выполнение данного условного оператора прекращается.



## ЗАПИСЬ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ В EXCEL, VBA И MATHCAD

### Математические константы

<i>мат. запись</i>	<i>Excel</i>	<i>MathCAD</i>	<i>VBA</i>
число $\pi$	ПИ()	$\pi$	4·Atn(1)
число $e^1$	EXP(1)	$e^1$	Exp(1)

### Математические функции

<i>мат. запись</i>	<i>Excel</i>	<i>MathCAD</i>	<i>VBA</i>	<i>мат. запись</i>	<i>Excel</i>	<i>MathCAD</i>	<i>VBA</i>
$ x $	ABS(x)	$ x $	Abs(x)	$\sqrt{x}$	КОРЕНЬ(x)	$\sqrt{x}$	Sqr(x)
$\lg x$	LOG10(x)	log(x)	Log(x) / Log(10)	$e^x$	EXP(x)	$e^x$	Exp(x)
$\log_n x$	LOG(x;n)	log(x,n)	Log(x) / Log(n)	$\ln x$	LN(x)	ln(x)	Log(x)
ctg x	1/ TAN(x)	cot(x)	1/ Tan(x)	cos x	COS(x)	cos(x)	Cos(x)
	COT(x)*			sin x	SIN(x)	sin(x)	Sin(x)
arctg x	ATAN(x)	atan(x)	Atn(x)	tg x	TAN(x)	tan(x)	Tan(x)
arcctg x	ПИ()/2 – ATAN(x)	acot(x)	4·Atn(1)/2 – Atn(x)	arccos x	ACOS(x)	acos(x)	–
	ACOT(x)*			arcsin x	ASIN(x)	asin(x)	–

Примечание: \* начиная с версии Excel 2013

### Арифметические операции

<i>мат. запись</i>	<i>Excel</i>	<i>MathCAD</i>	<i>VBA</i>
$-a^n$	– (a^n)	$-a^n$	– (a^n)
$\sin^2 x$	SIN(x)^2	$\sin(x)^2$	Sin(x)^2
$\cos^2 x^3$	COS(x^3)^2	$\cos(x^3)^2$	Cos(x^3)^2
$\sqrt[n]{a}$	a^(1/n)	$\sqrt[n]{a}$	a^(1/n)
$\frac{a+b}{c \cdot d}$	(a+b)/(c*d)	$\frac{a+b}{c \cdot d}$	(a+b)/(c*d)

### ОСНОВНЫЕ ЭЛЕМЕНТЫ БЛОК-СХЕМ

	Начало или конец алгоритма		Проверка условия
	Ввод или вывод информации		Организация циклов с параметрами
	Обработка данных (вычисление)		Порядок выполнения действий

## ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ

-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----

A series of horizontal dotted lines for writing.

## Лабораторная работа № 1

### ТЕМА. Знакомство с табличным процессором Excel. Работа с таблицами и объектами в текстовом редакторе Word.

#### Задание:

1. Создать таблицу в **Excel** для решения прямой продуктовой задачи (для известного объема сырья на входе необходимо найти объем продукта на выходе, полученного путем многоступенчатого технологического процесса его обработки) по предлагаемому образцу и сохранить её в папку **R:\Информатика\ЛабРаб1** под именем **labrab\_1.xls**.

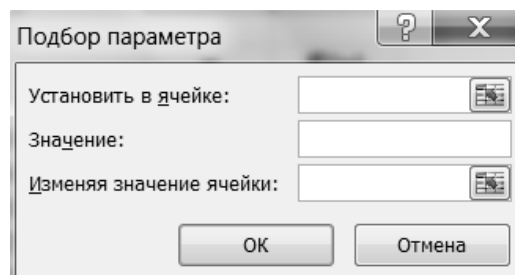
Замечание: Десятичные дроби переписываются с тремя знаками после запятой.

		Объем сырья на входе ( $P_{вх}$ ):					
Этап (i)	Коэффициент потерь, % ( $K_i$ )		Объем потерь ( $\Delta_i$ )		Объем продукта на выходе ( $P_{вых}$ )		
	Значения	Формулы	Значения	Формулы	Значения	Формулы	
1							
2							
3							
4							
5							
					Значения	Формулы	
Среднее значение коэффициента потерь:							
Максимальное значение объема потерь:							
Общее значение объема потерь:							

2. Решить задачу обратного продуктового расчета (для заданного объема продукта на выходе найти объем потребного для него сырья) с помощью инструмента «Подбор параметра».

**Объем продукта на выходе ( $P_{вых}$ ) = \_\_\_\_\_**

**Объем сырья на входе ( $P_{вх}$ ) = \_\_\_\_\_**



3. Создать составной документ в текстовом редакторе **Word** по предлагаемому **ОБРАЗЦУ** и сохранить его в папку **R:\Информатика\ЛабРаб1** под именем **Отчет\_labrab\_1.doc**.

4. В документе должны присутствовать следующие объекты:

- заголовок оформлен с помощью дополнительного модуля **WordArt** (2003: **Вставка/Рисунок**; 2007+: **Вставка/Текст/WordArt**);
- рисунок оформлен в виде **таблицы** без обрамления (5 столбцов и 3 строки). Картинки добавить можно двумя способами: через буфер обмена (*копированием*)

из графического редактора, в котором они предварительно открыты), либо как внедренный объект (2003: Вставка/Рисунок/Из файла/кнопка **Обзор** ; 2007+: Вставка/Текст/Объект/Создание из файла). Файлы с картинками можно просмотреть по ссылке:

U:\VT&PM\MS\_F\МАПП\Информатика\Лабораторные работы\ЛабРаб1\Рисунки

- после заголовка добавлено поле «Дата и Время» (2003: Вставка/Дата; 2007+: Вставка/Текст/Дата и время) с форматом «1 августа 2017 г.» и режимом  **Обновлять автоматически**.
  - внедрена таблица из Excel как **связанный файл** (2003: Вставка/Объект/Создание из файла/кнопка **Обзор** ,  Связь с файлом; 2007+: Вставка/Текст Объект/Создание из файла);
  - набраны математические формулы с помощью дополнительного модуля *Microsoft Equation*.
5. В колонтитулы добавить текст «ФИО, группа».

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

### Задания на защиту лабораторной работы № 1

**Задание.** Для заданного объема продукта на выходе найти объем потребного для него сырья:

**Объем продукта на выходе ( $P_{вых}$ )= \_\_\_\_\_, Объем сырья на входе ( $P_{вх}$ )= \_\_\_\_\_.**

выдано: \_\_\_\_\_  
(дата) (подпись)

### Лабораторная работа № 2 (1)

**ТЕМА. Исследование функции на отрезке в вычислительных средах.  
Табулирование функций и поверхностей, построение их графиков  
в ЭТ Excel + СКМ MathCad.**

**Задание:** 1. На рабочем листе «Непрерывная функция» в *Excel* задать отрезок, определить шаг табулирования и получить таблицу значений непрерывной функции  $y(x)$ . По таблице значений построить график.

**Условие:**

$y(x) =$  \_\_\_\_\_  $a =$  \_\_\_\_\_  $b =$  \_\_\_\_\_  $n =$  \_\_\_\_\_

		А	В
n →		a=	
n+1 →		b=	
n+2 →		n=	
n+3 →		h=	

Формулы в ячейках:

\_\_\_\_\_ = \_\_\_\_\_;  
(B, n+3)

\_\_\_\_\_ = \_\_\_\_\_;  
(A, n+9)

\_\_\_\_\_ = \_\_\_\_\_  
(B, n+9)

2. Задать функцию  $y(x)$  в СКМ *MathCad*, построить на заданном отрезке  $[a, b]$  её график и вывести его на печать:

MathCAD:  
график X-Y  
отображение осей (Axis Stile) – пересечение (Crossed)  
в легенде: —  $y(x)$   
● нули функции  
■ локальные максимумы  
◆ локальные минимумы

ориентировочный размер графика –  $9 \times 16$  см.

3. Протабулировать непрерывную функцию  $y(x)$  в СКМ *MathCad*.

**Листинг решения (в MathCad):**

4. На рабочем листе «Кусочно-непрерывная функция» в *Excel* задать отрезок, определить шаг табулирования и получить таблицу значений кусочно-непрерывной функции  $y1(x)$ . По таблице значений построить график.

**Условие:**

$y1(x) =$

$a =$  \_\_\_\_\_

$b =$  \_\_\_\_\_

$n =$  \_\_\_\_\_

		<b>A</b>	<b>B</b>
$n \rightarrow$		$a =$	
$n+1 \rightarrow$		$b =$	
$n+2 \rightarrow$		$n =$	
$n+3 \rightarrow$		$h =$	

Формулы в ячейках:

\_\_\_\_\_ = \_\_\_\_\_;  
(B, n+3)

\_\_\_\_\_ = \_\_\_\_\_;  
(A, n+10)

\_\_\_\_\_ = \_\_\_\_\_.  
(B, n+10)

5. Задать функцию  $y1(x)$  в *СКМ MathCad*, построить на заданном отрезке  $[a, b]$  её график.

**Листинг решения (в MathCad):**



6. На рабочем листе «Поверхность» в *Excel* задать отрезки разбиений по осям «X» и «Y», определить для каждой оси шаг табулирования и получить таблицу значений поверхности  $Z(x,y)$ . Построить по таблице значений график поверхности.

**Условие:**

$Z(x, y) =$  \_\_\_\_\_

$a_x =$  \_\_\_\_\_  $b_x =$  \_\_\_\_\_  $n_x =$  \_\_\_\_\_

$a_y =$  \_\_\_\_\_  $b_y =$  \_\_\_\_\_  $n_y =$  \_\_\_\_\_

	A	B	C	D
n →	$a_x =$		$a_y =$	
n+1 →	$b_x =$		$b_y =$	
n+2 →	$n_x =$		$n_y =$	
n+3 →	$h_x =$		$h_y =$	
n+4 →				
n+5 →	X	Y		
n+6 →				
n+7 →				

Формулы в ячейках:

\_\_\_\_\_ = \_\_\_\_\_;  
(B, n+3)

\_\_\_\_\_ = \_\_\_\_\_;  
(D, n+3)

\_\_\_\_\_ = \_\_\_\_\_;  
(A, n+7)

\_\_\_\_\_ = \_\_\_\_\_;  
(C, n+5)

\_\_\_\_\_ = \_\_\_\_\_;  
(D, n+7)

7. Задать поверхность  $Z(x,y)$  в *СКМ MathCad*, построить её график.

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

### Лабораторная работа № 2 (2)

**ТЕМА. Исследование функции на отрезке в вычислительных средах.  
Поиск нулей и экстремумов функции одной переменной  
в ЭТ Excel + СКМ MathCad.**

**Задание:** 1. На рабочем листе «Непрерывная функция» в *Excel* по таблице значений функции  $y(x)$  определить отрезки, содержащие:

• нули функции:

1) 


 2) 


 3) 


• локальный минимум функции:


• локальный максимум функции:


Замечание: Десятичные дроби переписываются с пятью знаками после запятой.

2. Уточнить нули функции  $y(x)$  в *Excel* с помощью средства «Подбор параметра».

		<b>Значение нуля</b>	<b>Значение функции</b>
	1-ый нуль		
	2-ой нуль		

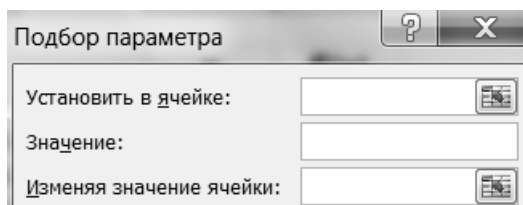
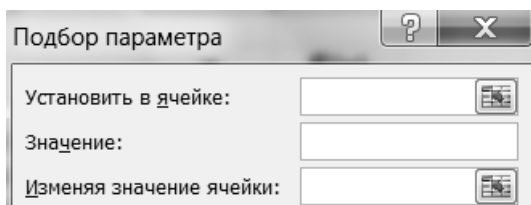
Формула функции: \_\_\_\_\_ =  
(адрес ячейки)

\_\_\_\_\_  
\_\_\_\_\_



Для 1-го нуля: начальное приближение в ячейке \_\_\_\_\_ = \_\_\_\_\_

Для 2-го нуля: начальное приближение в ячейке \_\_\_\_\_ = \_\_\_\_\_



3. Уточнить **крайний правый** нуль функции  $y(x)$  в *СКМ MathCad* с помощью функции **root()**.

**Листинг решения (в MathCad):**

4. На рабочем листе «Непрерывная функция» в *Excel* уточнить локальные экстремумы функции  $y(x)$  с помощью средства «Поиск решения»:

• локальный минимум функции:

• локальный максимум функции:

[ \_\_\_\_\_ ; \_\_\_\_\_ ]

[ \_\_\_\_\_ ; \_\_\_\_\_ ]

5. Уточнить локальные экстремумы функции  $y(x)$  в *СКМ MathCad* с помощью блока **Given / Maximize (Minimize)**.

**Листинг решения (в MathCad):**

6. Найти наибольшее и наименьшее значения функции на отрезке.

**В Excel:**

• наименьшее:

• наибольшее:



**Листинг решения (в MathCad):**

7. Сравнить результаты, полученные в разных вычислительных средах.

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

## Задания на защиту лабораторной работы № 2

**Задание 1.** Определить шаг табулирования на участке [a=\_\_\_\_; b=\_\_\_\_; n=\_\_\_\_] и получить таблицу значений непрерывной функции  в Excel  в MathCad.  $f(x)=$ \_\_\_\_\_.

По полученной таблице значений  $f(x)$ :  
 построить:  график функции.  
 определить отрезки содержащие:  нули функции;  локальные минимумы функции;  локальные максимумы функции.  
 найти:  наименьшее и  наибольшее значение функции.

1) 


 2) 


 3) 


 4) 


выдано: \_\_\_\_\_  
 (дата) (подпись)

**Задание 2.** Определить шаг табулирования, получить таблицу значений и построить график кусочно-непрерывной функции на отрезке [a=\_\_\_\_; b=\_\_\_\_] при n=\_\_\_\_\_.

	a=	
	b=	
	n=	
	h=	
...		
	<b>x</b>	<b>f(x)</b>

Расчетные формулы:

= \_\_\_\_\_  
 \_\_\_\_\_  
 = \_\_\_\_\_

выдано: \_\_\_\_\_  
 (дата) (подпись)

**Задание 3.** Построить таблицу значений и график поверхности  $Z(x,y)$ :

$Z(x,y) =$  \_\_\_\_\_

$a_x =$ \_\_\_\_  $b_x =$ \_\_\_\_  $n_x =$ \_\_\_\_  
 $a_y =$ \_\_\_\_  $b_y =$ \_\_\_\_  $n_y =$ \_\_\_\_

Формула поверхности в крайней левой верхней ячейки таблицы значений:

\_\_\_\_\_ = \_\_\_\_\_

выдано: \_\_\_\_\_  
 (дата) (подпись)

### Лабораторная работа № 3 (1)

#### ТЕМА. Обработка массивов в вычислительных средах. Операции над матрицами в ЭТ Excel + СКМ MathCad.

**Задание:** 1. В ЭТ *Excel* для заданных матриц **A** и **B**, вектора **d** и числа **k** (две последние цифры логического имени / 10) вычислить:

- используя команды с клавиатуры, соответствующие арифметическим операциям:  $A + B$ ,  $A - B$ ,  $k * B$ ,  $k * d$ ,  $A / k$ ;
- используя встроенные функции:  $A * B$ ,  $B * A$ ,  $B * d$ ,  $A^2$ ,  $\Delta A$ ,  $B^{-1}$ ,  $d^2$ ;
- выполнить транспонирование матрицы **A** с помощью встроенной функции, а матрицы **B** с помощью специальной вставки.

**Условие:**

Матрица <b>A</b>	Матрица <b>B</b>	Вектор <b>d</b>	Число <b>k</b>
_____	_____	_____	_____
(диапазон ячеек)	(диапазон ячеек)	(диапазон ячеек)	(адрес ячейки)

2. В ЭТ *Excel* вычислить значение матричного выражения **M** по частям.

**Условие:** Матрица **M** = \_\_\_\_\_

Часть матричного выражения	Диапазон ячеек на листе в Excel	Формула в ячейках

Результаты:


Замечание: Десятичные дроби переписываются с двумя знаками после запятой.

3. В ЭТ *Excel* вычислить значение матричного выражения **M** одной формулой.

Результаты: \_\_\_\_\_ = \_\_\_\_\_  
(диапазон ячеек) (формула)

4. В СКМ *MathCad* сформировать матрицу **C**, вставив \_\_\_\_\_

Полученное выражение: \_\_\_\_\_ Результат: \_\_\_\_\_

C:= \_\_\_\_\_

\_\_\_\_\_ C = \_\_\_\_\_

5. В СКМ *MathCad* сформировать вектор  $v:=(v_1 \ v_2 \ v_3 \ v_4 \ v_5)$ , где

Условие	Выражение
	v1:=
	v2:=
	v3:=
	v4:=
	v5:=

Ответ:  $v = ( \underline{\hspace{1cm}} \ \underline{\hspace{1cm}} \ \underline{\hspace{1cm}} \ \underline{\hspace{1cm}} \ \underline{\hspace{1cm}} )$

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

### Задания на защиту лабораторной работы № 3 (1)

**Задание 1.** Вычислить в Excel матричное выражение **M** = \_\_\_\_\_

по частям;  одной формулой.

Часть матричного выражения	Диапазон ячеек на листе в Excel	Формула в ячейках

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 2.** Сформировать в СКМ MathCad матрицу  $C$  по заданному условию:

---



---

Полученное выражение:

Результат:

$C :=$  \_\_\_\_\_

$C =$

---



---

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 3.** Сформировать вектор  $v := (v_1 \ v_2 \ v_3 \ v_4)$ , где

**Условие**

**Выражение**

	$v_1 :=$	
	$v_2 :=$	
	$v_3 :=$	
	$v_4 :=$	

**Ответ:**  $v = ($  \_\_\_\_\_ \_\_\_\_\_ \_\_\_\_\_ \_\_\_\_\_  $)$

выдано: \_\_\_\_\_  
(дата) (подпись)

### Лабораторная работа № 3 (2)

**ТЕМА.** *Решение систем линейных алгебраических уравнений (СЛАУ) в ЭТ Excel + СКМ MathCad.*

**Задание:** 1. Решить СЛАУ в ЭТ *Excel* с четырьмя неизвестными как матричное уравнение.

**Условие:**

Система  
уравнений:



<p>а) ввести матрицу коэффициентов <b>A</b> и вектор свободных членов <b>b</b>:</p>	<table border="1" style="display: inline-table; margin-right: 20px;"> <tr><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td></tr> </table> <table border="1" style="display: inline-table;"> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>																																													
<p>б) вычислить определитель матрицы <b>A</b>:</p>	$ A  =$ _____ _____ = _____ (адрес ячейки) (формула)																																													
<p>в) вычислить обратную матрицу <b>A<sup>-1</sup></b>:</p>	_____ = _____ (диапазон ячеек) (формула)																																													
<p>г) найти вектор-решение по формуле <b>x = A<sup>-1</sup> · b</b> :</p>	_____ = _____ (диапазон ячеек) (формула)																																													
<p>Результат:  <i>Замечание:</i> Десятичные дроби переписываются с тремя знаками после запятой.</p>	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> </table>																																													
<p>д) выполнить проверку найденного решения по норме (длине) вектора невязки <b>A · x - b</b>.</p>	_____ = _____ (диапазон ячеек) (формула вектора невязки) _____ = _____ (адрес ячейки) (формула длины вектора невязки) _____ = _____ (результат)																																													

2. Решить СЛАУ в ЭТ *Excel* методом Крамера (методом определителей).

а) вычислить определитель матрицы **A** и определители вспомогательных матриц, полученных заменой *i*-го столбца столбцом свободных членов заданной СЛАУ:

$|A| =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

$|A1| =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

$|A2| =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

$|A3| =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

$|A4| =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

б) найти вектор-решение по формулам Крамера  $x_i = |A_i| / |A|$

$x_1 =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

$x_2 =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

$x_3 =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

$x_4 =$  \_\_\_\_\_, формула в ячейке \_\_\_\_\_ = \_\_\_\_\_

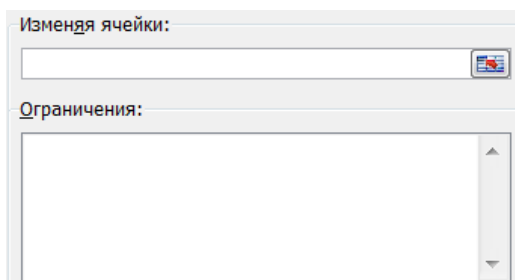
3. Решить СЛАУ в ЭТ *Excel* с помощью встроенных возможностей (надстройки «Поиск решения»).

		<b>x1</b>	<b>x2</b>	<b>x3</b>	<b>x4</b>				
Исходные значения:	<b>x=</b>								
		<b>A</b>					<b>A·x</b>	<b>=</b>	<b>b</b>
	<i>ур1</i>							<b>=</b>	
	<i>ур2</i>							<b>=</b>	
	<i>ур3</i>							<b>=</b>	
	<i>ур4</i>							<b>=</b>	

Формулы в ячейках:

\_\_\_\_\_ = \_\_\_\_\_  
 \_\_\_\_\_ = \_\_\_\_\_  
 \_\_\_\_\_ = \_\_\_\_\_  
 \_\_\_\_\_ = \_\_\_\_\_

Диалоговое окно «Поиск решения»:



Результат:


*Замечание:* Десятичные дроби переписываются с тремя знаками после запятой.

4. Решить СЛАУ в СКМ *MathCad* как матричное уравнение.

а) ввести матрицу коэффициентов **A** и вектор свободных членов **b**:

$$A: = \left( \begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array} \right) \quad b: = \left( \begin{array}{c} \\ \\ \\ \end{array} \right)$$

б) найти вектор-решение

$$x: = \underline{\hspace{2cm}} = \text{(выражение)}$$

$$\left( \underline{\hspace{1cm}}, \underline{\hspace{1cm}}, \underline{\hspace{1cm}}, \underline{\hspace{1cm}} \right)^T$$

*Замечание:* Десятичные дроби переписываются с тремя знаками после запятой.

5. Решить СЛАУ в СКМ *MathCad* с помощью встроенных возможностей (блока **Given / Find**) и выполнить проверку найденного решения по норме (длине) вектора невязки.

6. Сравнить результаты, полученные в разных вычислительных средах.

### Листинг решения (в MathCad):

```
' Начальные значения переменных
x1 := x2 := x3 := x4 :=
' Определение блока Given ... Find
Given

( ) := Find( ) ' Поиск решения

( ) = ( ) ' Результат

' Определение вектора невязки

' Вычисление нормы вектора невязки
```

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

### Задания на защиту лабораторной работы № 3 (2)

**Задание 1.** Решить СЛАУ как матричное уравнение  $\square$  в Excel (одной формулой)  $\square$  в MathCad.

Система уравнений: {

Результат (в Excel):


*Замечание:* Десятичные дроби переписываются с тремя знаками после запятой.

### Листинг решения (в MathCad):

Формула в диапазоне ячеек

\_\_\_\_\_ = \_\_\_\_\_

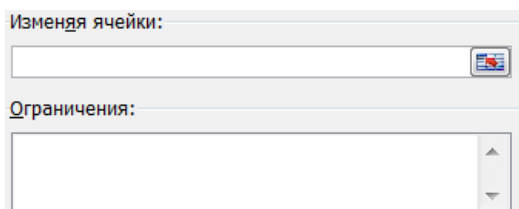
выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)



**Задание 2.** Решить СЛАУ в *Excel* с помощью встроенных возможностей.

Система уравнений: {

Диалоговое окно «Поиск решения»:



Результат:


Замечание: Десятичные дроби переписывать с тремя знаками после запятой.

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

**Задание 3.** Решить СЛАУ в СКМ *MathCad* с помощью встроенных возможностей:

Система уравнений:

$$A: = \left( \begin{array}{c} \\ \\ \end{array} \right) \quad b: = \left( \begin{array}{c} \\ \\ \end{array} \right)$$

Листинг решения (в *MathCad*):



Замечание: Десятичные дроби переписываются с тремя знаками после запятой.

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

## Лабораторная работа № 4

### ТЕМА. Работа со списками в ЭТ Excel. Использование временных функций и функций поиска в Excel-таблице.

#### Задание:

1. Открыть файл с исходной базой данных (далее – БД) **СписокЛР4.xls** и сохранить его на рабочий диск R:\Информатика\...
2. На рабочем листе «**Исходный\_список**» добавить в список 5 новых записей с произвольными данными с помощью **формы**.
3. Создать на отдельных рабочих листах (с соответствующими именами) справочные таблицы, в которых всем столбцам данных (без заголовков) присвоить имена для дальнейшего их использования в формулах.

<i>Имя диапазона</i>	<i>Диапазон ячеек на листе в Excel</i>	<i>Имя диапазона</i>	<i>Диапазон ячеек на листе в Excel</i>

4. Заполнить таблицу формулами для расчета. Для извлечения данных из справочных таблиц использовать функции ПРОСМОТР( ), ВПР( ).

<i>Название столбца</i>	<i>Адрес ячейки первой строки</i>	<i>Формула в ячейке</i>

5. Скопировать в текущей рабочей книге рабочий лист «Исходный\_список» 6 раз. Каждому листу присвоить новое имя: **Сортировка, Автофильтр1, Автофильтр2, Расш\_фильтр, Итоги, Сводная таблица**.
6. На рабочем листе «**Сортировка**» отсортировать список по указанным полям. Выполнить условное форматирование данных таблицы по варианту.

**Условие:** \_\_\_\_\_

Форматировать значения, для которых следующая формула является истинной:

Образец:

7. На рабочем листе «Автофильтр1», используя возможности Автофильтра, выбрать записи по критериям для одного поля таблицы БД.

*Условие: Выбрать перевозки*

---



---



---



---



---

Пользовательский автофильтр

Показать только те строки, значения которых:

и  или

8. На рабочем листе «Автофильтр2», используя возможности Автофильтра, выбрать записи по критериям для нескольких полей таблицы БД.

*Условие: Выбрать* \_\_\_\_\_

---



---

Пользовательский автофильтр

Показать только те строки, значения которых:

и  или

Пользовательский автофильтр

Показать только те строки, значения которых:

и  или

9. На рабочем листе «Расш\_фильтр» (в отдельных диапазонах) используя возможности расширенного фильтра, сделать выборки:

9.1. по критериям из п. 5 (автофильтра1)


9.2. по критериям из п. 6 (автофильтра2)

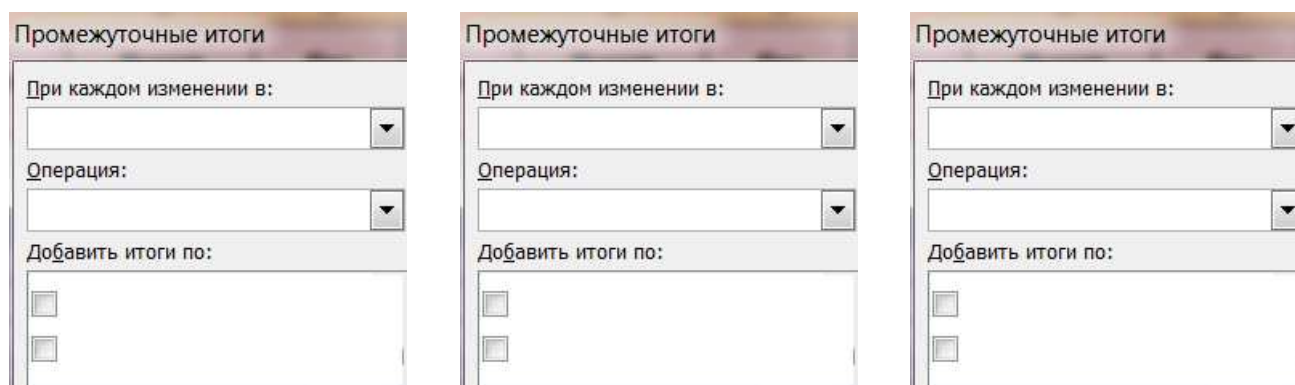

9.3. по вычисляемому критерию

**Условие:** Выбрать \_\_\_\_\_


Вычисляемый критерий: \_\_\_\_\_ = \_\_\_\_\_.  
(ячейка) (формула)

10. На рабочем листе «Итоги» подвести промежуточные итоги.

**Условие:** Подсчитать \_\_\_\_\_



11. На рабочем листе «Сводная таблица» построить сводную таблицу.

**Условие:**

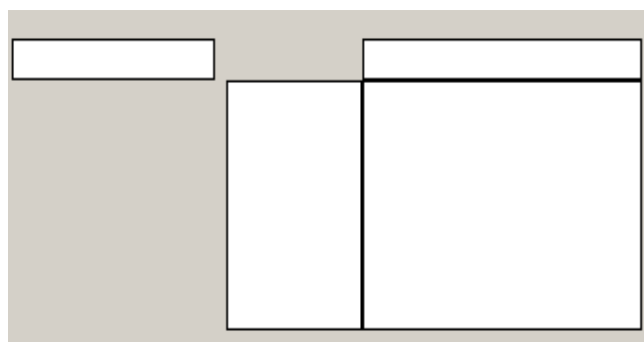
Страница в разрезе: \_\_\_\_\_

Макет: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_



12. Настроить параметры таблицы на рабочем листе «Исходный\_список» перед печатью: разместить документ на одном листе формата А4 альбомной ориентации, уменьшить размер полей, оформить колонтитулы: в верхний добавить тему лабораторной работы, в нижний – ФИО, № группы.

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

## Задания на защиту лабораторной работы № 4

**Задание 1.** Выбрать записи по заданным критериям используя возможности

Автофильтра,  Расширенного фильтра.

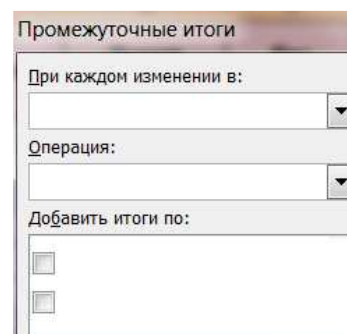
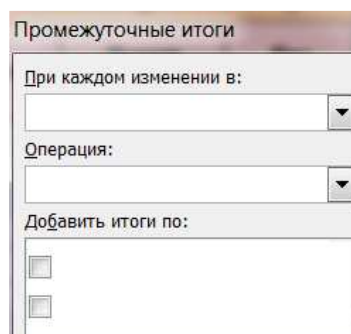
**Условие:** \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_


Вычисляемый критерий: \_\_\_\_\_ = \_\_\_\_\_  
(ячейка) (формула)

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 2.** Подвести промежуточные итоги.

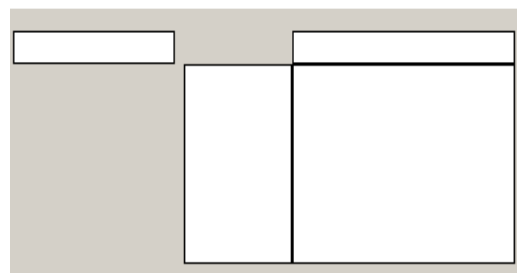
**Условие:** \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_



выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 3.** Построить сводную таблицу.

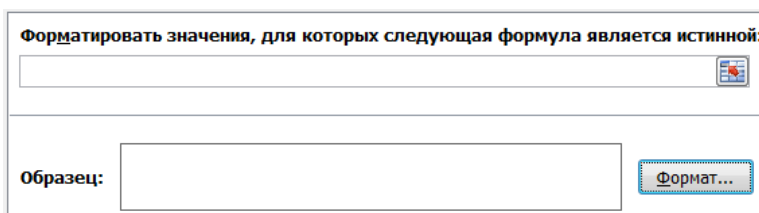
**Условие:**  
 Страница в разрезе: \_\_\_\_\_  
 Макет: \_\_\_\_\_  
 \_\_\_\_\_



выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 4.** Выполнить условное форматирование данных таблицы.

**Условие:** \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_



выдано: \_\_\_\_\_  
(дата) (подпись)

## Лабораторная работа № 5

ТЕМА. Основы работы в VBA. Создание процедур с линейной структурой.  
Работа с макросами.

### ЧАСТЬ I. Процедуры-функции.

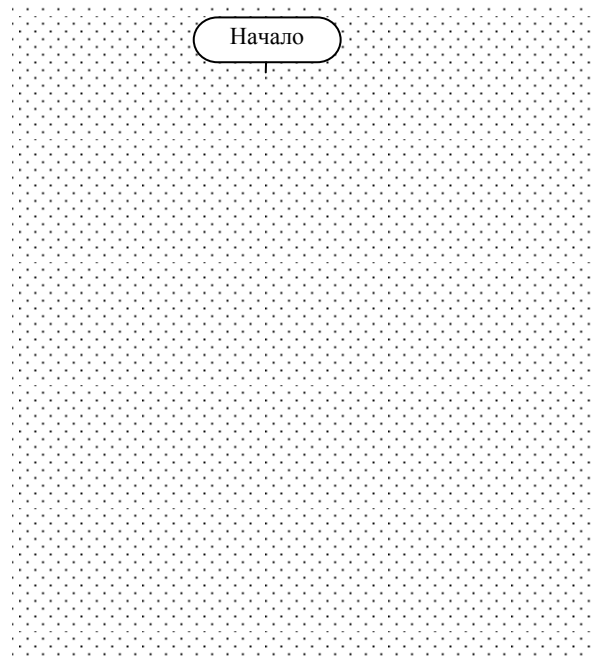
#### Задание:

1.1. Разработать линейную программу в виде процедуры-функции для вычисления значения выражения, зависящего от произвольных значений переменных *a*, *b*, *c*. Разбить выражение на простейшие части. Составить блок-схему алгоритма.

#### Условие:

#### Public Function

```
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----
```



#### End Function

1.2. Вычислить в Excel значение полученной пользовательской функции, задав значения входным переменным в любых ячейках рабочего листа.

1.3. Получить значение заданного выражения обычной формулой в Excel.

	Параметры:	a=		Результат выражения (функция VBA):	
		b=			
		c=		Результат выражения (формула Excel):	

Замечание: Десятичные дроби переписываются с тремя знаками после запятой.

Формула Excel: \_\_\_\_\_ = \_\_\_\_\_  
(ячейка) (формула)

1.4. Вычислить заданное выражение в СКМ MathCAD.

**Ответ** = \_\_\_\_\_

## ЧАСТЬ II. Процедуры-подпрограммы.

**Задание:** Разработать линейный алгоритм в виде процедуры-подпрограммы для вычисления геометрической формулы. Составить блок-схему алгоритма. Исходные данные ввести в окно InputBox, а результат вывести в окно MsgBox. Закрепить выполнение процедуры за указанным элементом интерфейса.

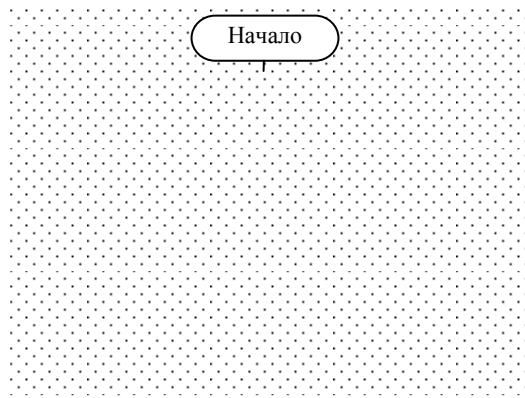
*Условие:*

*Расчетная формула:*


**Public Sub**

-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----

**End Sub**



*Значение исходных данных:*

*Результат:*

--	--

## ЧАСТЬ III. Макросы.

**Задание:** Записать макрос с помощью макрорекодера для форматирования ячейки: изменения размера и начертания шрифта, обрамления и заполнения любым цветом фона (желательно бледным оттенком). Закрепить макрос за комбинацией клавиш. Выполнить макрос для форматирования ячеек со значениями параметров и результатами вычислений выражения в ЧАСТИ I лабораторной работы.

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

### Задания на защиту лабораторной работы № 5

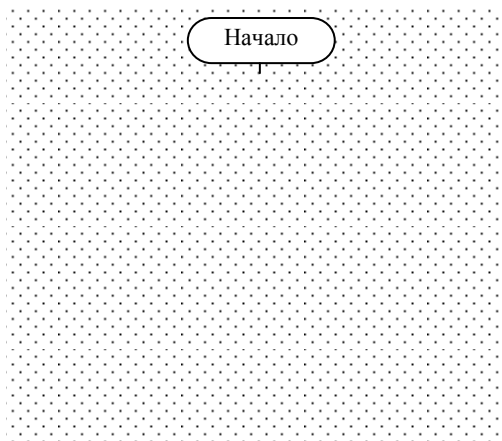
**Задание 1.** Разработать линейную программу в виде процедуры-функции для вычисления значения выражения. Составить блок-схему алгоритма.

*Условие:*

**Public Function**

-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----

**End Function**



выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 2.** Разработать процедуру-подпрограмму для вычисления формулы. Составить блок-схему алгоритма. Исходные данные ввести в окно InputBox, а результат вывести в окно MsgBox. Закрепить выполнение процедуры за указанным элементом интерфейса.

*Условие:*

*Расчетная формула:*


**Public Sub**

-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----

**End Sub**

*Значение исходных данных:*

*Результат:*

выдано: \_\_\_\_\_  
(дата) (подпись)

### Лабораторная работа № 6

#### ТЕМА. Создание процедур с разветвляющейся структурой в Excel + VBA + СКМ MathCad.

**Задание:**

1. Получить значения корней квадратного уравнения  $_____x^2 + _____x + _____ = 0$  в Excel с помощью функции **ЕСЛИ**( ) – в случае отрицательного значения дискриминанта вывести в результирующих ячейках текст «нет корней».
2. Проверить значения корней уравнения в Excel с помощью средства «Подбор параметра».



	A	B	C	D	E	F	G
1	Коэффициенты:			Дискриминант:		D=	
2	a=						
3	b=			Корни (функция ЕСЛИ):			
4	c=			x <sub>1</sub> =		x <sub>2</sub> =	
5							
6	Корни ("Подбор параметра"):						
7		x <sub>1</sub> =			x <sub>2</sub> =		
8							
		Корни (пользовательская функция)				Корни (процедура-подпрограмма)	
		x <sub>1</sub>	x <sub>2</sub>			x <sub>1</sub>	x <sub>2</sub>

Замечание: Десятичные дроби переписываются с тремя знаками после запятой.

H1 = \_\_\_\_\_  
(формула)

E4 = \_\_\_\_\_  
(формула)

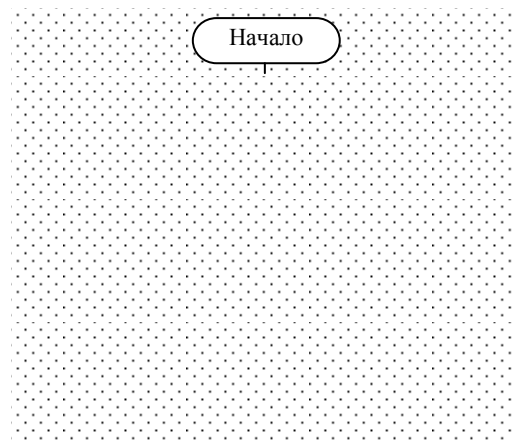
G4 = \_\_\_\_\_  
(формула)

D7 = \_\_\_\_\_, G7 = \_\_\_\_\_  
(формула) (формула)

3. Создать процедуру-функцию для решения квадратного уравнения с использованием простого многострочного оператора условного перехода **If...End If**. Составить блок-схему алгоритма.

**Public Function**

-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----



**End Function**

4. Создать процедуру-подпрограмму для решения квадратного уравнения с использованием расширенного многострочного оператора условного перехода **If...End If**. Коэффициенты уравнения считать с ячеек рабочего листа, полученные значения корней отобразить в отдельных ячейках листа. Закрепить полученную процедуру за кнопкой на рабочем листе.

Public Sub

-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----

End Sub

5. Выполнить решение уравнения в СКМ MathCAD с помощью функции *root*( ).

**Листинг решения (в MathCad):**

-----

Работа выполнена верно: (дата) \_\_\_\_\_ (подпись) \_\_\_\_\_

**Задания на защиту лабораторной работы № 6**

**Задание 1.** Найти корни квадратного уравнения  $_____x^2 + _____x + _____ = 0$  в СКМ MathCAD с помощью функции *root*( ).

**Листинг  
решения  
(в MathCad):**

-----

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 2.** Создать пользовательскую функцию для вычисления значений кусочно-непрерывной функции при разных значениях аргумента (попадающих в разные интервалы) с использованием:

- простого однострочного оператора **If**;
- простого многострочного оператора **If**;
- расширенного многострочного оператора **If**.

**Условие:**

$$f(x) = \left\{ \begin{array}{l} \end{array} \right.$$

**Public Function**

-----  
-----  
-----  
-----  
-----  
-----  
-----  
-----

**End Function**

		$x$	$f(x)$
	на 1-ой ветке		
	на 2-ой ветке		
	на 3-ей ветке		

(формула)

Замечание: Десятичные дроби переписываются с тремя знаками после запятой.

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 3.**

Вычислить значения разветвляющейся функции в СКМ MathCAD при разных значениях аргумента (попадающих в разные интервалы) с помощью функции *if* ( ).

**Условие:**

$$f(x) = \left\{ \begin{array}{l} \end{array} \right.$$

**Листинг решения (в MathCad):**

-----

выдано: \_\_\_\_\_  
(дата) (подпись)

## ОГЛАВЛЕНИЕ

Общие указания	3
Отметки о защите лабораторных работ	3
Методические указания к выполнению лабораторных работ	4
ЭЛЕКТРОННАЯ ТАБЛИЦА MICROSOFT EXCEL	4
СИСТЕМА КОМПЬЮТЕРНОЙ АЛГЕБРЫ MATHCAD	18
ЯЗЫК ПРОГРАММИРОВАНИЯ VISUAL BASIC FOR APPLICATIONS В MICROSOFT EXCEL	26
ЗАПИСЬ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ В EXCEL, VBA И MATHCAD	33
ОСНОВНЫЕ ЭЛЕМЕНТЫ БЛОК-СХЕМ	33
ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ	34
Лабораторная работа № 1	36
ЗАДАНИЯ НА ЗАЩИТУ ЛАБОРАТОРНОЙ РАБОТЫ № 1	37
Лабораторная работа № 2 (1)	37
Лабораторная работа № 2 (2)	40
ЗАДАНИЯ НА ЗАЩИТУ ЛАБОРАТОРНОЙ РАБОТЫ № 2	42
Лабораторная работа № 3 (1)	43
ЗАДАНИЯ НА ЗАЩИТУ ЛАБОРАТОРНОЙ РАБОТЫ № 3 (1)	44
Лабораторная работа № 3 (2)	45
ЗАДАНИЯ НА ЗАЩИТУ ЛАБОРАТОРНОЙ РАБОТЫ № 3 (2)	48
Лабораторная работа № 4	50
ЗАДАНИЯ НА ЗАЩИТУ ЛАБОРАТОРНОЙ РАБОТЫ № 4	53
Лабораторная работа № 5	54
ЗАДАНИЯ НА ЗАЩИТУ ЛАБОРАТОРНОЙ РАБОТЫ № 5	55
Лабораторная работа № 6	56
ЗАДАНИЯ НА ЗАЩИТУ ЛАБОРАТОРНОЙ РАБОТЫ № 6	58

Учебное издание

*Составители:*

*Гучко Ирина Михайловна  
Кофанов Валерий Анатольевич  
Рамская Людмила Константиновна*

# ЛАБОРАТОРНЫЙ ПРАКТИКУМ

по дисциплине «Информатика»

для студентов специальности

«Машины и аппараты пищевых производств»

машиностроительного факультета

дневной формы обучения

*2-е издание, исправленное и переработанное*

*Текст печатается в авторской редакции,  
пунктуации и орфографии*

Ответственный за выпуск: Гучко И.М.

Редактор: Боровикова Е.А.

Компьютерная вёрстка: Гучко И.М.

---

Подписано в печать 02.09.2019 г. Формат 60x84 <sup>1</sup>/<sub>8</sub>. Бумага «Performer».  
Гарнитура «Times New Roman». Усл. печ. л. 7,44. Уч. изд. л. 8,0. Заказ № 1088. Тираж 12 экз.  
Отпечатано на ризографе учреждения образования «Брестский государственный  
технический университет». 224017, г. Брест, ул. Московская, 267.