

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Кафедра интеллектуальных информационных технологий

**АВТОМАТИЗАЦИЯ РАБОТЫ С ДОКУМЕНТАМИ
ПРИ ИСПОЛЬЗОВАНИИ ТЕКСТОВОГО РЕДАКТОРА
MICROSOFT WORD**

Методические указания

к выполнению лабораторных работ по дисциплине

“Программные средства персональных компьютеров”

для студентов специальности

53 01 02 «Автоматизированные средства обработки информации»

УДК 681.3

В методических указаниях приведены необходимые теоретические сведения о средствах автоматизации создания документов в текстовом редакторе Word, таких как использование полей и издательских возможностей Word, создание макросов на языке Visual Basic. Содержатся указания к выполнению трех лабораторных работ по указанной тематике (постановка задачи, содержание отчета и контрольные вопросы для проверки).

Методические указания предназначены для использования студентами специальности 53 01 02 в ходе выполнения лабораторных работ по дисциплине «Программные средства персональных компьютеров» и при оформлении курсовых и дипломных проектов, а также могут быть полезны инженерам – программистам и опытным пользователям для создания удобной рабочей среды текстового редактора Microsoft Word.

Составитель: Л.А. Горбашко, ст.преподаватель

Рецензент: Муравьев Г.Л. директор Брестского филиала учреждения образования «Институт современных знаний», к.т.н.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
ЛАБОРАТОРНАЯ РАБОТА № 1 АВТОМАТИЗАЦИЯ ЗАПОЛНЕНИЯ ДОКУМЕНТОВ С ПОМОЩЬЮ ПОЛЕЙ WORD	6
КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ.....	6
ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ.....	9
СОДЕРЖАНИЕ ОТЧЕТА	10
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	10
ЛАБОРАТОРНАЯ РАБОТА № 2 УПРАВЛЕНИЕ БОЛЬШИМИ ДОКУМЕНТАМИ.....	12
КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ.....	12
ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ.....	17
СОДЕРЖАНИЕ ОТЧЕТА	19
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	19
ЛАБОРАТОРНАЯ РАБОТА №3 СОЗДАНИЕ МАКРОСОВ WORD С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА VISUAL BASIC FOR APPLICATION.....	20
КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ.....	20
ПРИМЕРЫ ПРОГРАММ НА ЯЗЫКЕ VBA	24
ВАРИАНТЫ ЗАДАНИЙ	26
СОДЕРЖАНИЕ ОТЧЕТА	28
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	28
ЛИТЕРАТУРА	28

ВВЕДЕНИЕ

В настоящее время наиболее широко распространенным текстовым редактором является редактор Microsoft Word, входящий в комплект интегрированного пакета Microsoft Office. На сегодняшний день это наиболее мощный текстовый редактор, который предназначен как для простого создания и оформления текста, так и использования его в качестве настольной издательской системы.

Вопросы редактирования и форматирования текста входят в программу школьного курса дисциплины «Информатика». Для совершенствования использования широких возможностей Word студентами специальности АСОИ предложена проблема автоматизации создания документов, которая позволяет ускорить обработку документов, избавить пользователя от рутинной работы, упростить сложные приемы обработки текста для неквалифицированного пользователя.

Текстовый редактор Word содержит множество функций автоматизации обработки документов, таких как использование стилей, списков, полей Word; создание макросов; подготовка текста к изданию (создание оглавления, алфавитного указателя, сносок, примечаний); работа над текстом совместно нескольких пользователей, генерация документов методом слияния, использование шаблонов и др.

Материал трех лабораторных работ, представленных далее, включает в себя наиболее значимые, часто используемые и сложные для освоения методы автоматизации. Полностью средства автоматизации можно изучить, используя специальную литературу (например, [1]).

ЛАБОРАТОРНАЯ РАБОТА № 1

АВТОМАТИЗАЦИЯ ЗАПОЛНЕНИЯ ДОКУМЕНТОВ С ПОМОЩЬЮ ПОЛЕЙ WORD

Цель работы. Изучить категории существующих полей Word и научиться их применять для заполнения документов.

Краткие теоретические сведения

Виды полей

Поле – инструкция (команда) обработки текста. Обычно коды полей скрыты от пользователя. Выполнение этих инструкций и дает результат в виде выполненного действия.

По выполняемым функциям все поля WORD делятся на 9 категорий:

1. автоматизация документа;
2. дата и время;
3. нумерация;
4. оглавление и указатели;
5. сведения о документе;
6. сведения о пользователе;
7. связи и ссылки;
8. слияние;
9. формулы.

В каждую из категорий включено несколько полей. Функции полей различны.

Все поля можно сгруппировать следующим образом:

- ♦ к категории 1 и 7 относятся поля, выполняющие определенные действия (например, INCLUDEPICTURE – вставить рисунок);
- ♦ к категориям 3, 5, 6 – с помощью этих полей можно поместить информацию из памяти компьютера или из одного документа в другой;
- ♦ к категории 8 – запрос и ввод информации в документ;
- ♦ к категории 4 – поля-маркеры, не выводят никаких значений и не производят никаких действий; они определяют элементы, включаемые в оглавление и алфавитный указатель.

Состав и структура полей

Поле состоит из четырех основных частей:

- ♦ символов поля;

- ◆ имени поля;
- ◆ инструкций (аргументов);
- ◆ вспомогательных элементов.

Символы поля — { }. Обозначают начало и конец поля. Их можно вставить ТОЛЬКО с помощью команды меню Вставка-Поле (или Ctrl-F9).

Имя поля вводится вручную или выбирается из списка диалогового окна. Может представлять из себя знак равенства =.

Инструкции (аргументы) - текст или числа, управляющие работой поля. Если аргумент представляет собой строку (состоит более чем из одного слова), его требуется заключать в кавычки. Пример аргумента:

FILLIN	"Введите фамилию" ... }
имя поля	аргумент

Если текст не заключен в кавычки, то в качестве аргумента используется только первое слово.

{FILLIN Введите фамилию ... } - аргументом является только слово Введите.

Вспомогательные части поля:

Закладка - помечает выделенный текст. Может хранить информацию.

Ключ - необязательные инструкции, включают\выключают некоторые результаты полей (например, ключи форматирования -/MERGEFORMAT).

Режимы отображения полей

Поля существуют в двух режимах:

- ◆ в виде кодов { };
- ◆ в виде значений (*значение поля*- результат выполнения инструкций, представленный на экране).

Обычно используется режим значений полей; коды полей не видны. Для отображения полей на экране следует перейти в режим кодов полей. Переключение режимов отображения кодов полей можно выполнять следующими способами:

- ◆ все поля : Сервис-Параметры-Вид-флажок Коды полей или Alt-F9;
- ◆ одно поле : 1) контекстное меню поля - коды\значения полей;
2) курсор в нужном поле- Shift - F9.

Замечание. При этом способе коды некоторых полей не выводятся, а автоматически формируются как скрытый текст (например, поле XE). Для отображения кодов эти поля нужно установить флажок Скрытый текст (Сервис-Параметры- Вид).

Обновление полей

Обновление – изменение значения полей в результате подстановки новых данных. При обновлении поля меняется его результат. Поля, не имеющие видимых значений (например, поле ASK), изменяют результаты других полей.

Внимание! Автоматически поля НЕ обновляются.

Поля можно обновить по одному и в совокупности все имеющиеся в документе.

Способы обновления полей вручную:

◆ Контекстное меню поля- Обновить поле

Клавиша F9 для текущего поля.

(Для обновления нескольких полей нужно сначала их выделить (Правка – Выделить все)).

Способы автоматического обновления полей:

- ◆ при печати : установить обновление полей (Сервис – Параметры-Печать – Обновлять поля);
- ◆ при открытии документа (например, поле DATE);
- ◆ при печати слиянием (поля категории Слияние, например, NEXT);
- ◆ при разбивке на страницы;
- ◆ автоматически (поле AUTONUM- автономерация).

Ключи в кодах полей

Некоторые поля могут иметь ключи. Ключи позволяют расширить возможности полей. Чаще всего они используются для форматирования результатов заполнения полей.

Каждый ключ имеет свое назначение. Ключ состоит из обратной наклонной черты \ и одного или нескольких символов. Поле может содержать до 10 ключей. Существуют универсальные ключи и ключи для одного конкретного поля.

Некоторые ключи вводятся автоматически при вставке поля. Ключи можно добавлять и вручную, редактируя коды полей. При вставке поля можно использовать диалоговое окно для добавления ключей (кнопка Параметры).

Закладки

Для создания закладки следует установить курсор в позицию, которую нужно отметить, или выделить нужный текст. Затем выполнить Вставка- Закладка- Ввести имя- Добавить.

Общее назначение закладок – быстро перейти к нужному месту в книге (Правка- Перейти), помеченному ранее закладкой..

Средство Закладка может широко применяться для различных целей. Например, закладки могут использоваться в кодах полей для обмена информацией между полями (аналог переменной).

Для пометки текста закладкой его нужно выделить и выполнить команду Вставка-Закладка, ввести имя закладки. При этом в диалоговом окне отображается список уже существующих закладок.

При вставке закладок в коды полей поступают по-другому: нужно просто ввести с клавиатуры имя закладки в код поля. При выполнении инструкции, содержащейся в поле, новая закладка автоматически попадает в список закладок.

ПОДРОБНАЯ ИНФОРМАЦИЯ О КОДАХ И КЛЮЧАХ ПОЛЕЙ ПРИВЕДЕНА В РАЗДЕЛЕ Дополнительные сведения Справки текстового редактора Word.

Порядок выполнения работы

Создать приведенный на рисунке 1 документ «Анонс научной конференции». Для заполнения данных, обведенных рамкой, использовать поля WORD.

Правила заполнения полей приведены ниже.

XI - номер конференции вводится с клавиатуры в виде десятичного числа. В документе номер отображается римскими цифрами.

22 ноября 2000 - текущая дата. Заполняется автоматически. Формат даты должен соответствовать указанному отображению.

78,1 % - число вводится с клавиатуры. В документе отображается в процентном формате с одним знаком после запятой.

21,9 % - вычисляется (100-предыдущее число). Формат такой же.

Конференция будет великолепна! – Текст вставляется автоматически в документ в зависимости о значения процента участников конференции. Если в конференции участвуют не менее 90% заявленных участников, то печатается текст 'Конференция будет великолепна', в противном случае - 'Процент участия невелик'. Формат отображения соответствует указанному.

40 – выбирается макс. число из таблицы (столбец Количество экспонатов)

75 – сумма всех чисел из таблицы. (столбец Количество экспонатов)

высшей .- Автоматически определяется категория конференции и вставляется соответствующий текст. Категория определяется по следующему правилу:

Общее количество экспонатов	Категория
150 и более	высшая
80-149	первая
менее 80	вторая

1 – ссылка на страницу, на которой содержится число- общее количество экспонатов.

47 – количество слов в данном документе (статистические данные)

слов – текст должен склоняться по необходимым падежам в зависимости от количества слов.

Пример: 47 слов, 42 слова.

Иванов А.А. - автор документа.

Содержание отчета

1. Определение поля . Состав полей . Вспомогательные элементы поля.
2. Режимы отображения полей. Способ переключения между режимами (любой), способ обновления полей (любой).
3. Используемые ключи полей.
4. Распечатка документа в режиме кодов полей либо переписанные от руки ВСЕ коды полей со вспомогательными элементами (каждое поле - с новой строки).

Контрольные вопросы

1. Определение поля . Категория полей.
2. Состав полей .
3. Режимы отображения полей. Способ переключения между режимами.
4. Методы обновления полей.
5. Форматирование результатов заполнения поля.
6. Назначение и использование закладок.

АНОНС НАУЧНОЙ КОНФЕРЕНЦИИ

XI конференция состоится 22 ноября 2000 г.

В конференции принимают участие 78,1 % научно-производственных объединений (НПО).

Остальные 21,9 % не участвуют.

Конференция будет великолепна!

Название НПО	Количество экспонатов
Маяк	10
Окунь	20
им. Ленина	5
Электросила	40

Наибольшее количество экспонатов, представленных одним НПО - 40

Общее количество экспонатов - 75.

Наша конференция принадлежит к категории высшей лиги.

(Категория определяется по количеству экспонатов, указанному на стр. 1)

В данном документе 47 слов. Исполнитель Иванов А.А.

Рисунок 1

ЛАБОРАТОРНАЯ РАБОТА № 2

УПРАВЛЕНИЕ БОЛЬШИМИ ДОКУМЕНТАМ

Цель работы. Научиться применять текстовый редактор WORD для оформления документов по правилам издательской системы.

Краткие теоретические сведения

Документ WORD может быть представлен для просмотра в нескольких режимах, предоставляющих пользователю разные возможности обработки документа: режимы

Режим структуры

Переход в любой режим осуществляется через меню Вид.

Режим структуры документа предусмотрен в качестве специального режима, в котором заголовки различных уровней связаны с подзаголовками и текстом. Для работы в этом режиме обязательно должны быть использованы стандартные стили заголовков различного уровня, принятые для обычного режима. В структуре могут использоваться все девять видов встроенных заголовков различного уровня от первого до девятого. При этом для всех уровней заголовков используется один уровень текста.

Использование режима структуры обеспечивает следующие удобства при работе над документом:

- ◆ Если установить сворачивание документа до уровня заголовков с выведением первой строки раздела, можно использовать режим структуры для быстрого просмотра больших документов
- ◆ Можно легко перемещаться по документу, сворачивая и разворачивая отдельные разделы
- ◆ Режим структуры удобно использовать для редактирования. Выделяя отдельные заголовки и используя кнопки панели структуры, их можно перемещать вместе со связанным с ними текстом

Стили

Форматирование - придание тексту определенного вида. Форматирование состоит в задании параметров форматирования. Различают форматирование символов и форматирование абзацев. Параметры форматирования символов: шрифт, кегль, эффекты. Параметры форматирования абзацев: отступы, выравнивание, межстрочный интервал, интервал до и после абзаца и некоторые другие.

Стиль - это совокупность параметров форматирования, имеющая свое название.

Представьте себе документ, отформатированный одним шрифтом, без отступов, табулирующий, заголовков. Такой текст трудно читать. Теперь представьте, что вам придет-

ся отформатировать каждый из элементов вручную. Если документ имеет достаточно большой объем, то задача становится неразрешимой при условии, что все элементы одного типа (например, подписи и заголовки соответствующих уровней) должны быть оформлены единообразно.

С помощью стилей можно установить несколько параметров форматирования за один шаг, включая форматирование символов и форматирование абзацев. При этом для каждого из элементов оформления можно установить собственный стиль, который будет иметь уникальное наименование.

В Word стиль может быть применен как к целому абзацу, так и к нескольким символам. Принципиальное отличие заключается в том, что абзац, благодаря признаку конца абзаца, можно не выделять, достаточно установить в нем курсор, а стиль символов можно применить к фрагменту текста любой величины, но без знака абзаца, поэтому требуемый фрагмент обязательно должен быть выделен. Кроме того, при применении стиля к произвольному фрагменту текста устанавливаются только параметры форматирования символов, в то время как при задании стиля абзаца устанавливается полный набор параметров форматирования.

Собственно назначение стилей состоит в удобстве их применения. Для применения стиля следует установить курсор в пределах абзаца, стиль которого следует поменять, открыть список стилей стандартной панели инструментов и выбрать нужный стиль. Для создания нового стиля следует выполнить команду меню **Формат-Стиль-Создать**.

Автоматическая подпись названий

Названия и перекрестные ссылки облегчают составление и редактирование больших документов. Это удобный инструмент для автоматической нумерации рисунков, таблиц, формул.

Название – подписи к рисункам, заголовки таблиц, формул, могут включать нумерацию.

Перекрестная ссылка – ссылка за дополнительной информацией к указанному элементу (например, к номеру указанного объекта или к его названию).

Названия и перекрестные ссылки существуют в документе в виде кодов полей. Для обновления номеров указанных элементов при редактировании текста следует обновить поля.

Для подписи объекта после его создания следует выполнить пункт меню **Вставка-Название**. При этом устанавливается фиксированная часть подписи, формат номера и вводится изменяемая часть подписи. Подпись названия осуществляется стилем «Название объекта».

Для создания перекрестной ссылки на объект следует выполнить пункт меню **Вставка – Перекрестная ссылка**.

Сноски

Сноски и концевые сноски применяются для объяснений, комментариев или организации ссылок на первоисточники в документе. В одном и том же документе вы можете одновременно использовать как сноски, так и концевые сноски, (например, сноски применять для подробных комментариев, а концевые сноски — для ссылок на первоисточники).

Обычные сноски располагаются в нижней части страницы под чертой, отделяющей их от основного текста. Концевые сноски располагаются в конце документа.

Сноска любого вида состоит из двух частей: маркера ссылки на сноску и сопутствующего ему текста сноски. Поэтому вставка сносок выполняется в два этапа:

- 1) обозначается место, к которому относится сноска. В этом месте устанавливается маркер сноски (как правило — это просто порядковый номер сноски)
- 2) вводится текст сноски.

Отображение и изменение разделителей сносок выполняется только в режиме Обычный.

Оглавления и указатели

Алфавитный указатель предоставляет возможность отыскать нужные элементы в документе по номерам страниц. Содержится обычно в конце документа (книги).

Создание указателей.

При создании указателей прежде всего следует отметить, какие именно элементы следует внести в алфавитный указатель. Для этого следует выделить и пометить элемент указателя в тексте (Вставка- Оглавление и указатели- Пометить).

После того, как все необходимые указатели помечены, следует создать алфавитный указатель (кнопка ОК).

При выполнении операции "Пометить указатель" в текст вставляется специальное поле WORD XE- элемент-указателя. Поле XE форматируется как скрытый текст и не отображается в документе. Поле XE вставляется для определения элемента, помещаемого в указатель, и располагается после выделенного элемента. С помощью этого средства можно редактировать элемент указателя, вводить новые элементы. Совокупность полей элементов XE формирует поле всего алфавитного указателя INDEX.

Для отображения скрытого текста следует снять флажок Скрытый текст (Сервис- Параметры).

Многоуровневый указатель

От простого алфавитного указателя многоуровневый отличается тем, что вслед за главным элементом в нем следуют элементы второго уровня. (Например, первый уровень -язык, второй уровень- английский, русский). созданный указатель будет выглядеть следующим образом:

Простой указатель		Многоуровневый	
Язык	78,100	Язык	
		англ.	78
		рус	100

При создании многоуровневого указателя выделенный текст попадает в поле Основной элемент. В поле Дополнительный вводится элемент второго уровня.

Автоматическая пометка элементов указателя.

При работе с большими документами пометка элементов представляет собой достаточно утомительную задачу. Альтернативное средство пометки указателей – файл соответствий.

Файл соответствий создается путем создания нового документа. В документе вносятся информация в два столбца. В первом столбце – слова, которые следует включить в алфавитный указатель, во втором – соответствующие элементы указателя, которые должны быть созданы на основании информации, содержащейся в первом столбце. Для оформления информации в виде двух столбцов создается таблица. Слова из первого столбца помечаются в тексте как указатели, в алфавитный же указатель вместо них включаются слова второго столбца. Файл сохраняется как обычный документ.

На основании файла соответствий по всему тексту автоматически помечаются элементы указателя.

Оглавление.

Оглавление содержит заголовки в той последовательности, в которой они встречаются в документе, независимо от нумерации. Заголовок обычно определяется по использованному стилю.

Создание оглавления.

Перед созданием оглавления необходимо установить для всех заголовков, включаемых в оглавление, стили соответственно уровню заголовка. Для создания оглавления следует выполнить команду Вставка-Оглавление_и_указатели-Оглавление-ОК.

Если для заголовков использованы стандартные стили, то оглавление будет помещено в текущую позицию курсора.

Если для заголовков использованы собственные стили, не основанные на стандартных, следует вызвать параметры оглавления. В списке существующих стилей установить требуемый для оглавления уровень заголовков.

Следует учитывать, что при редактировании текста оглавление автоматически НЕ обновляется. Для внесения изменений в оглавление следует сгенерировать его еще раз.

Режим Главный документ

При подготовке большого документа группой исполнителей возникают сложности. С помощью Главного документа можно работать вместе с соавторами. Каждый автор создает свой отдельный файл, которые затем просматриваются как единый документ. Средство Главный документ предназначено для облегчения работы над большими документами, публикациями, состоящими из нескольких частей (глав) и обеспечивает:

- быстрый поиск и перемещение к любой части документа
- возможность знакомиться с выполненными частями работы, но в то же время присутствует запрет на изменения чужих глав без разрешения автора.
- соблюдение единого стиля оформления в большом документе, части которого готовятся разными людьми.
- печать документа целиком без необходимости открывать разные файлы
- возможность создания сквозных списков (нумерация страниц, указателей), сквозного оглавления.

Главный документ имеет специальный режим работы — режим Главного документа (ГД), который содержит функции режима Структуры документа, дополненные специальными функциями главного документа.

При работе в режиме ГД каждый документ представляет из себя главу. Любой документ может быть главным либо вложенным в него. Каждый вложенный документ хранится в отдельном файле.

Любой документ считается главным при переходе в режим ГД. К нему можно добавлять вложенные документы с помощью инструментальной панели.

Можно создать новый вложенный документ, который на экране отделяется рамкой, сохраняется в отдельном файле с именем, составленным из начала текста вложенного файла.

Можно вставить вложенный документ из уже существующего файла.

Можно объединить вложенные документы. Для этого:

- в режиме ГД расположите документы один за другим в том порядке, в котором хотите объединить их в новый вложенный документ;
- выделите заголовок всех объединяемых документов;
- нажмите кнопку Объединить;
- сохраните файл.

При этом все выделенные документы объединяются в один, который записывается под именем первого документа.

Для ограничения доступа к текущему документу пользуйтесь кнопкой Блокировка панели инструментов.

Порядок выполнения работы

Для выполнения задания использовать текст в файле Full.doc. Для удобства выполнения задания разделены на категории по темам.

1) СТИЛЬ.

Создать собственный стиль на основе стандартного:

Заг1 – Times New Roman, кегль 14, все буквы заглавные, с новой страницы, выравнивание по левому краю с абзаца 1,5, интервал после абзаца – 2 межстрочных, формат нумерации – арабская цифра и пробел.

Заг2 – Times New Roman, кегль 14, интервал до и после абзаца – 3 межстрочных; Текстовый – кегль 14, межстрочный интервал- 1,2, абзацный отступ 1,5 см, выравнивание по левому краю;

Применить собственные стили к заданному тексту.

2) АВТОМАТИЧЕСКИЕ НАЗВАНИЯ.

Добавить в документ три рисунка: один в первом разделе и два во втором. Подписать эти рисунки, используя названия. Рисунки нумеровать в пределах раздела через точку, номер от подписи отделяется пробелом. Стиль для подписей должен иметь кегль 12, выравнивание по центру, интервал после абзаца – 1,5. Пример подписи рисунка:



Рисунок 3.1 - Структура иерархии модулей

Добавить рисунок между вторым и третьим и подписать его. Как изменяется нумерация названий?

Удалить второй рисунок и его подпись. Как изменяется нумерация названий? Выполнить действия для обновления номеров рисунков (какие?)

Добавить в документ текст в виде ссылок на рисунки, например, «Описанное взаимодействие иллюстрирует рисунок 2.1». Слова «рисунок 2.1» должны быть организованы в виде перекрестной ссылки.

Добавить рисунок между первым и вторым. Как изменилась нумерация рисунков и ссылок?

Удалить второй рисунок. Какие существуют ошибки перекрестных ссылок?

Провести обновление ссылок. Как изменились нумерация ссылок и ошибки ссылок?

3) СНОСКИ

Создать 2 обычные сноски для слов НИОКР и ПРОГНОЗИРОВАНИЕ. Самостоятельно объяснять эти понятия.

Создать одну концевую сноску для слова СКАНЕР.

Изменить формат маркеров сносок на а, в, с.

Удалить одну сноску.

Изменить вид разделителя сносок на такой ♥♥♥♥♥♥♥♥.

4) АЛФАВИТНЫЙ УКАЗАТЕЛЬ.

Создать в конце текста алфавитный указатель основных понятий в одну колонку.

В него включить слова НИОКР, КОНФЕРЕНЦИЯ.

Проверить возможность скрытия и отображения поля ХЕ на экране.

Создать многоуровневый алфавитный указатель для словосочетания УНИВЕРСИТЕТ и всех возможных названий университетов, имеющихся в тексте, для 2-го уровня.

Создать файл соответствий, в который внести 2 указателя по желанию. Добавить данные указатели в алфавитный указатель.

Удалить из текста указатель КОНФЕРЕНЦИЯ. Как изменится алфавитный указатель?

5) ОГЛАВЛЕНИЕ

Создать в начале текста оглавление документа, включающее два уровня иерархии с заполнителем строки _____

Добавить в текст новый раздел. Вносятся ли автоматически изменения в оглавление?

Обновить оглавление и алфавитный указатель документа в соответствии с внесенными изменениями.

Обновить оглавление таким образом, чтобы заголовки первого и второго уровня поменялись в иерархии структуры. (без внесения изменений в текст)

[РЕЖИМ СТРУКТУРЫ.

Перейти в режим структуры

Ознакомиться с назначением кнопок панели инструментов "Структура".

Отобразить на экране: а) заголовки только 2-х уровней; б) весь текст

Переместить список монографий в раздел "Внедрение результатов НИОКР"

Удалить подраздел "Монографии"

7) ГЛАВНЫЙ ДОКУМЕНТ.

Создать главный документ из текущего документа

Создать 2 вложенных документа:

один уже существующий вложить в раздел «Хоздоговорны...»;

пустой документ вложить во вновь созданный раздел «Исследовательская работа студентов»

Внести информацию в пустой документ. Обязательно использовать хотя бы один заголовок

Объединить два вложенных документа в один

Заблокировать полученный вложенный документ от внесения изменений

Создать оглавление главного документа. Как отражаются в нем заголовки вложенных документов?

Содержание отчета

1. Определение стиля. Имя созданного стиля.
2. Виды сносков. Нумерация и пояснения созданных сносков.
3. Типы элементов, из которых составляются оглавление и алфавитный указатель. Кол-во данных элементов в заданном тексте.
4. При выполнении какого действия обновляются оглавление и алфавитный указатель.
5. Имя и содержание файла соответствий.
6. Назначение и возможности режима Структура.
7. Назначение одной любой кнопки на панели инструментов "Структура".
8. Назначение и возможности режима Главный документ.
9. Имена файлов: главного документа, двух вложенных, полученного вложенного после объединения.

Контрольные вопросы

1. Чему равен одинарный межстрочный интервал в пунктах?
2. Что такое форматирование, стиль, сноска, оглавление, алфавитный указатель?
3. Два способа создания алфавитного указателя.
4. Два способа создания оглавления.
5. Наименования полей, помечающих элементы алфавитного указателя и оглавления?
6. Назначение режима структуры
7. Сходство и отличие режимов Структуры и Главный документ.
8. Как сгенерировать оглавление составного большого документа, разделы которого находятся в разных файлах?
9. Какие поля Word используются при создании автоматического названия?
10. Какой стиль используется для автоматического названия? Как изменить стиль подписи рисунка?

ЛАБОРАТОРНАЯ РАБОТА №3 СОЗДАНИЕ МАКРОСОВ WORD С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА VISUAL BASIC FOR APPLICATION

Краткие теоретические сведения

Создание макросов

Макрос (макрокоманда) – последовательность команд, имеющая имя. С помощью макросов можно автоматизировать действия, часто выполняемые пользователем. Для написания макросов используется язык Visual Basic for Application (VBA), встроенный в пакет Microsoft Office.

Редактор Visual Basic доступен во всех приложениях Microsoft Office.

Макрос может быть создан в автоматическом режиме и вручную в редакторе Visual Basic. Для создания макроса в автоматическом режиме следует выполнить пункт меню Сервис- Макрос- Начать запись. Уже существующий макрос можно отредактировать или удалить, используя диалоговое окно Сервис- Макросы.

Макросы сохраняются вместе с документом.

Структура проекта

Проект – это основная программа, которая объединяет и инициализирует все подключаемые модули, библиотеки и т.п. Обычно содержит описания объектов и интерфейса, вызовы процедур из подключенных модулей. В состав проекта входят следующие компоненты:

- 1) модули – содержат процедуры макросов;
- 2) модули класса – содержат описания пользовательских классов;
- 3) формы – содержат описания форм и процедур обработки событий; позволяют организовать диалог с пользователем;
- 4) объекты Microsoft Word – делают доступными другие документы;
- 5) ссылки – подключение библиотек; позволяют сделать объект другого приложения доступным в программе.

Наиболее часто используемые компоненты – это модули и формы.

В VBA существует 2 типа проектов :

- 1) Normal – проект сохранен вместе с шаблоном Normal.dot; содержащиеся в нем макросы доступны для любого документа Word;
- 2) Project – хранит только макросы данного документа.

При написании макроса с использованием форм следует учитывать, какой тип проекта выбран. Если создается проект типа Normal, то и форма должна быть создана в

ЛАБОРАТОРНАЯ РАБОТА №3

СОЗДАНИЕ МАКРОСОВ WORD С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА VISUAL BASIC FOR APPLICATION

Краткие теоретические сведения

Создание макросов

Макрос (макрокоманда) – последовательность команд, имеющая имя. С помощью макросов можно автоматизировать действия, часто выполняемые пользователем. Для написания макросов используется язык Visual Basic for Application (VBA), встроенный в пакет Microsoft Office.

Редактор Visual Basic доступен во всех приложениях Microsoft Office.

Макрос может быть создан в автоматическом режиме и вручную в редакторе Visual Basic. Для создания макроса в автоматическом режиме следует выполнить пункт меню Сервис- Макрос- Начать запись. Уже существующий макрос можно отредактировать или удалить, используя диалоговое окно Сервис- Макросы.

Макросы сохраняются вместе с документом.

Структура проекта

Проект – это основная программа, которая объединяет и инициализирует все подключаемые модули, библиотеки и т.п. Обычно содержит описания объектов и интерфейса, вызовы процедур из подключенных модулей. В состав проекта входят следующие компоненты:

- 1) модули – содержат процедуры макросов;
- 2) модули класса – содержат описания пользовательских классов;
- 3) формы – содержат описания форм и процедур обработки событий; позволяют организовать диалог с пользователем;
- 4) объекты Microsoft Word – делают доступными другие документы;
- 5) ссылки – подключение библиотек; позволяют сделать объект другого приложения доступным в программе.

Наиболее часто используемые компоненты – это модули и формы.

В VBA существует 2 типа проектов :

- 1) Normal – проект сохранен вместе с шаблоном Normal.dot; содержащиеся в нем макросы доступны для любого документа Word;
- 2) Project – хранит только макросы данного документа.

При написании макроса с использованием форм следует учитывать, какой тип проекта выбран. Если создается проект типа Normal, то и форма должна быть создана в

проекте Normal, иначе объекты формы будут недоступны в данном проекте. Содержащиеся модули и макросы в проекте наглядно отображаются в окне проекта, находящемся слева вверху в редакторе VBA.

Организация диалога

Организация диалога в VBA реализуется через пользовательские формы. Для создания новой формы используется кнопка панели инструментов Вставить UserForm.

Создание диалога состоит из двух этапов: настройка формы и добавление элементов управления.

Настройка формы осуществляется через окно свойств объекта. Для формы рекомендуется использовать следующие свойства объекта UserForm:

Свойство	Значение	Пояснение
Name	Myform	Имя
Caption	Шрифт	подпись формы в окне
Width	112	Ширина
Height	205	Высота

Возможные элементы управления находятся на панели инструментов формы. Они являются стандартными для диалога ОС Windows. Доступ к элементам управления осуществляется через семейство формы Controls. Это семейство включает в себя все элементы управления, установленные на форму. Каждому элементу управления присваивается номер в семействе, начиная с 0, в порядке установки на форму.

Например, установим на форму элемент управления Переключатель (объект типа CheckBox). Через окно свойств объекта инициализируем свойства:

Свойство	Значение
Name	C1
Caption	Жирный

Установить используемый переключатель (флажок) программным путем можно так:

`Myform.C1=True`

или

`Myform.Controls("C1")=True`

Использование событий

Функции объектно-ориентированной программы обычно активизируются при наступлении какого-либо события операционной системы. Примеры событий ОС Windows:

Событие	Имя
Щелчок мышью	Click
Получение фокуса	Enter
Потеря фокуса	Exit

Событие Получение фокуса наступает в момент активизации объекта, т.е. объект становится текущим.

При наступлении события активизируется процедура со строго определенным названием, которая называется процедурой обработки события или обработчиком события. Например, обработчик события нажатия кнопки с именем K1 выводит в поле с именем P1 текущей формы слово «Привет»:

```
Sub K1_Click()  
    P1.Text= " Привет "  
End Sub
```

Имя обработчика события формируется автоматически при выполнении двойного щелчка по соответствующему событию в окне событий объекта. Оно состоит из имени объекта (K1), знака подчеркивания и имени события(Click).

Для отображения формы на экране рекомендуется использовать метод Show объекта UserForm, а для скрытия формы после выбора пользователя - метод Hide.

Основные конструкции языка Basic

Язык Visual Basic построен на синтаксисе языка Basic и поддерживает все его конструкции.

Компилятор языка нечувствителен к регистру клавиатуры. Каждый оператор записывается с новой строки. При необходимости записать два оператора на одной строке их разделяют двоеточием. Строки комментариев обозначаются командой REM или одинарной кавычкой.

Описание и типы переменных

Допускается неявное описание переменных, т.е. задание типа переменной по присвоенному ей значению. Например, оператор присваивания

```
X=4,5
```

при неявном описании указывает компилятору, что тип переменной X – вещественный.

При явном описании можно использовать следующие типы переменных:

Integer, Byte, Long – целые,

Boolean – логический,

Single, Double – вещественные,

String – строковый.

Для описания переменной используется оператор DIM:

```
DIM X as Single
```

```
DIM A(20) as Integer
```

Описание массива .

Условный оператор:

IF логическое условие Then

 Операторы

 [Else в квадратных скобках - необязательная часть

 Операторы]

End IF

Оператор выбора:

Select Case переменная- переключатель

 Case Значение1

 Операторы 1

 Case Значение2

 Операторы 2

 [Else

 Операторы]

End Select

Операторы циклов:

FOR переменная цикла= начальное значение TO конечное значение [STEP шаг]

 Операторы

NEXT переменная цикла

While логическое условие

 Операторы

Wend

Методические рекомендации

При написании программы рекомендуется создать макрос в автоматическом режиме (Сервис- Макрос- Начать запись). Затем просмотреть сгенерированный текст в редакторе Visual Basic для определения объектов, методов и свойств, которые следует использовать при написании программы. При необходимости дополнительную информацию можно получить из справки в разделе Справка по Visual Basic.

Внимание! При использовании коллекции слов Words в состав слова включаются пробелы после него до следующего слова или знака препинания.

Рекомендуется использовать следующие объекты:

ОБЪЕКТ	СВОЙСТВО или МЕТОД	ПОЯСНЕНИЕ
ActiveDocument	PageSetup	Параметры страницы активного документа
	Words	Коллекция слов активного документа
Selection	Start	выделенный текст ; если текст не выделен, то текущая позиция курсора.
	End	
	Startof	
	Endof	
Range	Start	любой заданный диапазон текста, который необходимо обработать
	End	
	Startof	
	Endof	
Bookmark	Start	закладка
	End	
	Name	
	Delete	
UserForm	Show	форма – диалоговое окно для ввода-вывода данных
	Hide	

Примеры программ на языке VBA

Пример 1.

Весь документ представить в виде одного абзаца путем удаления символов признака конца абзаца ¶

Текст программы.

```
Set r = ActiveDocument.Range(Start:=0, End:=0)
```

```
r.WholeStory
```

```
Selection.StartOf unit:=wdStory
```

```
k = 1
```

```
i = 0
```

```
While k < r.End - i
```

```
Selection.EndKey unit:=wdLine
```

```
Selection.Delete unit:=wdCharacter, Count:=1
```

```
k = Selection.start
```

```
i = i + 1
```

```
Wend
```


Текст программы с комментариями

```
REM создание объектной переменной r типа диапазон, начальная и конечная  
REM позиции которого равны 0, т.е. диапазон пустой
```

```
Set r = ActiveDocument.Range(Start:=0, End:=0)
```

```
REM задание значения диапазону r – весь документ
```

r.WholeStory

```
REM выделения текста нет, поэтому объект Selection представляет из себя  
REM текущую позицию курсора, начало и конец этого объекта совпадают;  
REM установить начальную позицию – начало документа (единица измерения  
REM перемещений – весь документ)
```

```
Selection.StartOf unit:=wdStory
```

```
k = 1
```

```
i = 0
```

```
REM цикл выполняется до тех пор, пока k меньше разности номера последнего  
REM символа в документе (позиция конца диапазона r) и i
```

While k < r.End - i

```
REM переместить текущую позицию курсора в конец строки
```

```
Selection.EndKey unit:=wdLine
```

```
REM удалить один символ в текущей позиции курсора
```

```
Selection.Delete unit:=wdCharacter, Count:=1
```

```
REM переменной k присвоить значение начала выделения,
```

```
REM т.е. в данном случае текущей позиции курсора
```

```
k = Selection.start
```

```
i = i + 1
```

Wend

Пример 2.

В выделенном фрагменте текста после фамилии Петров добавить его инициалы.

```
Set col = Selection.Words
```

```
For Each a In col
```

```
    If a = "Петров" Then a.InsertAfter "B.B. "
```

```
Next
```

Пример 3.

В выделенном фрагменте текста изменить размер шрифта на 14 или 20 по выбору пользователя.

В шаблон Normal добавить объект UserForm и дать ему имя MyForm, используя свойство Name инспектора объектов.

На форму установить два компонента CheckBox и один CommonButton, дать им имена

соответственно Ch1, Ch2, B1.

Создать процедуру обработки события нажатия кнопки:

```
Private Sub B1_Click()  
REM Свойство Tag формы используется для передачи значения,  
REM выбранного пользователем.  
If ch1 = True Then Tag = 1 Else Tag = 0  
myform.hide  
REM форма невидима на экране  
End Sub
```

Текст программы макроса:

```
myform.Show  
REM показать форму  
If myform.Tag = 1 Then Selection.Font.Size = 14 Else Selection.Font.Size = 20
```

Варианты заданий

1. Составить программу перекодировки выделенного текста из кодировки ANSI (Windows) в KOI8. Соответствие символов следующее:

Кодировка	Исходная строка	Результат
ANSI-KOI8	Абвгдежзийклмнопр- стуфхцчшщъьзюя	юабидефгхийкљмно- пјрстужвѣыѝзшщѣ
KOI8- ANSI	Абвгдежзийклмнопр- стуфхцчшщъьзюя	бвцздеѣѝкљмноп- ртѣфжжигпояѝшщѣас

Программа должна проверять, выделен ли фрагмент текста, затем в диалоговом окне запрашивать у пользователя вид преобразования путем установки переключателя в группе из двух переключателей.

Для обработки строк рекомендуется использовать встроенные функции Basic: ASC, RIGHT; CHR, INSTR; информация в Справочнике по Visual Basic.

2. Составить программу правильной расстановки пробелов в тексте. В основе алгоритма лежат следующие правила редактирования текста:

- между словами ставится только один пробел;
- перед знаком препинания пробел не ставится;
- после знака препинания пробел обязательно ставится;

Макрокоманда должна обрабатывать выделенный фрагмент текста или весь текст при отсутствии выделения.

3. Составить программу удаления из текста переносов, расставленных вручную. Программа должна работать в двух режимах:

1 - автоматическое удаление переносов. При этом обнаруженные дефисы между словами сразу удаляются;

2 - запрос на удаление переноса. При обнаружении в тексте слова, разделенного дефисом на части, на экран выводится диалоговое окно с найденным словом. При этом пользователь выбирает одно из трех действий - удалить перенос, оставить перенос или завершить выполнение макрокоманды.

4. Составить программу печати документа книжкой. Для этого надо выполнить настройки параметров страницы таким образом, чтобы после распечатки документа листы можно было посередине сшить. Перед началом печати пользователем в диалоговом окне вводится желаемая ширина всех 4-х полей для одной стр., т.е. половины листа. Использовать вид элементов управления формы для ввода - поле увеличения-уменьшения (поле со счетчиком), предусмотреть значение полей по умолчанию.

Основа алгоритма печати следующая:

а) установить альбомную ориентацию страницы;

б) организовать 4 цикла печати. Для каждого цикла задаются свои размеры полей печати. В каждом цикле печатается 1/4 от общего количества страниц документа.

Например, в исходном файле 8 стр. Для распечатки его нужно 2 листа - по 2 стр. на лист с двух сторон. Длина листа формата А4 в альбомной ориентации - 297 мм, половина листа - 148 мм.

Если краевое поле каждой части стр. должно составлять 15 мм, а среднее - 30 мм, то ширина одной стр. текста - $148 - 30 - 15 = 103$ мм.

1 цикл - печать страниц 1 и 3. Поля: левое - 15 мм, правое - $297 - 103 - 15 = 179$ мм (см. рис. 2)

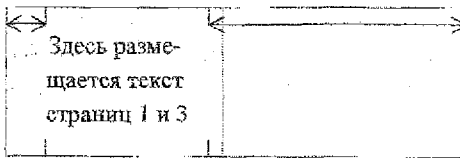


Рисунок 2 - Левое и правое поля для 1 цикла печати

2 цикл - печать 6 и 8 стр. на тех же сторонах двух листов. Поля: левое - $148 + 15 = 163$ мм, правое - 30 мм (см. рис.3).

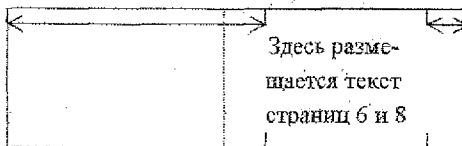


Рисунок 3 - Левое и правое поля для 2 цикла печати

Настройки 3 и 4 циклов аналогичны.

Если число страниц в документе не кратно 4, то в конец документа добавляются пустые страницы.

5. Составить программу сортировки разделов текста в алфавитном порядке заголовков. Текст при этом перемещается вместе со своим заголовком. Название стилей, примененных к заголовкам и к остальному тексту, вводится пользователем в диалоговом окне. Для ввода названия стилей в форме диалогового окна используется поле с раскрывающимся списком, содержащее все текущие стили документа.

6. Составить программу изменения начертания слов в выделенном фрагменте документа, написанных латинскими буквами. Вид начертания выбирается пользователем в диалоговом окне в виде установки любого количества флажков (Ж, К, Ч).

7. Составить программу обработки текста, который был создан в другом текстовом редакторе, при этом в конце каждой строки остался символ с кодом 13, который в Word является признаком конца абзаца и мешает форматированию текста. Макрокоманда должна удалять из текста лишние признаки конца абзаца. Начало нового абзаца определяется по абзацному отступу в виде пробелов (>3) или табуляции.

Содержание отчета

1. Цель работы
2. Задание
3. Схема алгоритма
4. Текст программы

Контрольные вопросы

1. Что такое макрос?
2. Структура проекта VBA.
3. Область действия объектов в зависимости от типа проекта.
4. Иерархия объектов документа.

ЛИТЕРАТУРА

1. Ильина М. Word 97. – М.: Бинком, 1998.
2. Персон Р., Поуз К. Microsoft Word 97 в подлиннике. – СПб.: BHV – Санкт-Петербург, 1997.
3. Культип Н.Б. Макрокоманды в MS WORD. – СПб.: BHV, 1998.
4. СТ БГТУ 01-2002. Оформление материалов курсовых и дипломных проектов (работ), отчетов по практике. Общие требования и правила оформления. - Брест, УО «БГТУ», 2002.

Создать оглавление главного документа. Как отражаются в нем заголовки вложенных документов?

Содержание отчета

1. Определение стиля. Имя созданного стиля.
2. Виды сноска. Нумерация и пояснения созданных сноска.
3. Типы элементов, из которых составляются оглавление и алфавитный указатель. Кол-во данных элементов в заданном тексте.
4. При выполнении какого действия обновляются оглавление и алфавитный указатель.
5. Имя и содержание файла соответствий.
6. Назначение и возможности режима Структура.
7. Назначение одной любой кнопки на панели инструментов "Структура".
8. Назначение и возможности режима Главный документ.
9. Имена файлов: главного документа, двух вложенных, полученного вложенного после объединения.

Контрольные вопросы

1. Чему равен одинарный межстрочный интервал в пунктах?
2. Что такое форматирование, стиль, сноска, оглавление, алфавитный указатель?
3. Два способа создания алфавитного указателя.
4. Два способа создания оглавления.
5. Наименования полей, помечающих элементы алфавитного указателя и оглавления.
6. Назначение режима структуры.
7. Сходство и отличие режимов Структуры и Главный документ.
8. Как сгенерировать оглавление составного большого документа, разделы которого находятся в разных файлах?
9. Какие поля Word используются при создании автоматического названия?
10. Какой стиль используется для автоматического названия? Как изменить стиль подписи рисунка?

УЧЕБНОЕ ИЗДАНИЕ

Составитель: Горбашко Лариса Анатовна

**АВТОМАТИЗАЦИЯ РАБОТЫ С ДОКУМЕНТАМИ
ПРИ ИСПОЛЬЗОВАНИИ ТЕКСТОВОГО РЕДАКТОРА
MICROSOFT WORD**

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению лабораторных работ
по дисциплине «Программные средства персональных компьютеров»
для студентов специальности 53 01 02

Ответственный за выпуск: Горбашко Л.А.

Редактор: Строчак Т.В.

Корректор: Никитчик А.Д.

Подписано к печати 26.01.04 г. Бумага писчая. Формат 60x84 1/16. Усл. печ. л. 1,6.
Уч.изд.л. 1,75. Тираж 100 экз. Заказ № 104. Отпечатано на ризографе учреждения
образования «Брестский государственный технический университет».
224017, г.Брест, ул. Московская, 267.