

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Программное обеспечение информационных технологий»

ИНФОРМАТИКА

*Методические рекомендации к лабораторным работам
для студентов направления подготовки
15.03.06 «Мехатроника и робототехника»
очной формы обучения*



Могилев 2022

УДК 004.65
ББК 32.973.26-0.18.2
Б17

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Программное обеспечение информационных технологий» «31» января 2022 г., протокол № 7

Составитель: ст. преподаватель А. С. Сидоренко

Рецензент: канд. техн. наук, доц. В. М. Ковальчук

Методические рекомендации содержат базовые сведения об основах информационных технологий, задания для выполнения лабораторных работ и примеры решения задач.

Учебно-методическое издание

ИНФОРМАТИКА

Ответственный за выпуск	В. В. Кутузов
Корректор	И. В. Голубцова
Компьютерная верстка	М. М. Дударева

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 21 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский
университет, 2022

Содержание

1 Лабораторная работа № 1. ТП Word. Основные навыки работы. Редактирование и форматирование текста.....	4
2 Лабораторная работа № 2. ТП Word. Создание, редактирование и форматирование таблиц.....	7
3 Лабораторная работа № 3. ТП Word. Слияние.....	12
4 Лабораторная работа № 4. ТП Word. Форма.....	15
5 Лабораторная работа № 5. ТП Word. Макросы.....	17
6 Лабораторная работа № 6. ЭТ Excel. Элементы окна. Настройка панели инструментов. Ввод и редактирование данных.....	19
7 Лабораторная работа № 7. ЭТ Excel. Использование встроенных функций...	20
8 Лабораторная работа № 8. ЭТ Excel. Диаграммы. Задача о траектории движения тела.....	22
9 Лабораторная работа № 9. ЭТ Excel. Решение уравнений и систем уравнений. Задачи оптимизации.....	23
10 Лабораторная работа № 10. ЭТ Excel. Методы работы с базами данных в приложении.....	26
11 Лабораторная работа № 11. VBA. Элементы управления.....	28
12 Лабораторная работа № 12. VBA. Оператор условного перехода. Объекты Excel.....	31
13 Лабораторная работа № 13. VBA. Оператор выбора With и Select Case.....	34
14 Лабораторная работа № 14. VBA. Цикл с параметром (For...Next).....	35
15 Лабораторная работа № 15. VBA. Циклы с предусловием и постусловием	37
16 Лабораторная работа № 16. VBA. Пользовательские функции.....	39
17 Лабораторная работа № 17. VBA. Пользовательские процедуры.....	40
18 Лабораторная работа № 18. Графический интерфейс пользователя. Простейшие вычисления Matlab.....	42
19 Лабораторная работа № 19. Операции с векторами и матрицами в системе Matlab.....	44
20 Лабораторная работа № 20. Построение графиков в Matlab.....	46
Список литературы.....	47

1 Лабораторная работа № 1. ТП Word. Основные навыки работы. Редактирование и форматирование текста

Цель работы: приобретение навыков создания документов в Microsoft Word.

Лабораторная работа состоит из двух частей, каждая из которых содержит теоретический раздел и практическое задание.

Результаты выполнения заданий сохранить в файлах, для размещения которых использовать рабочие папки (создаются в папке группы на сервере) или на собственные носители.

Для защиты работы представить отчет (в рукописном виде)

Порядок работы в текстовом редакторе Word.

СОЗДАНИЕ ДОКУМЕНТА.

Новый документ создается командой меню Файл→Создать. В результате на экране появляется диалоговое окно СОЗДАТЬ, в котором пользователю можно выбрать шаблон документа и нажать кнопку ОК.

УСТАНОВКА ПАРАМЕТРОВ СТРАНИЦЫ ДОКУМЕНТА.

Перед набором текстового документа можно установить параметры страницы документа (по умолчанию используются параметры, заданные в шаблоне).

Для этого надо использовать команду Файл→Параметры страницы (Разметка страницы). В результате появится диалоговое окно с вкладками: Поля, Размер бумаги, Источник бумаги, Макет.

Используя вкладку Поля, пользователь задает границы области текста в документе. С помощью вкладки Размер бумаги определяются размер бумажного листа и его ориентация. Редактор Word позволяет создавать документы на листах формата А5 (размер соответствует почтовой карточке), формата А4 (обычный печатный лист), формата А3 или установить собственный размер листа.

УСТАНОВКА ПАРАМЕТРОВ ФОРМАТИРОВАНИЯ АБЗАЦА (способ выравнивания текста, отступы и интервалы).

Для достижения этой цели необходимо установить курсор ввода в нужный абзац и выполнить команду Формат→Абзац (Главная→Абзац). В диалоговом окне установить требуемые значения в полях Выравнивание, Отступ, Интервал.

ПРОВЕРКА ПРАВОПИСАНИЯ.

Производится в документе командой Сервис→Правописание. (Рецензирование→Правописание).

СОХРАНЕНИЕ ДОКУМЕНТА.

Сохранение осуществляется выбором команды Файл→Сохранить (сохранить под старым именем) или Файл→Сохранить как... (сохранить с новыми параметрами – указываются новые папка, имя, тип и т. д.).

В результате на экране появляется диалоговое окно Сохранение документа. Это окно содержит поля, в которых задается имя файла и маршрут к нему (его адрес), указывается тип сохраняемого файла.

ЗАДАНИЕ.

1 Создайте новый документ (не вводя пока текста), имеющий следующие параметры страницы:

- размер бумаги – А4;
- верхнее поле – 2,5 см;
- нижнее поле – 2,5 см;
- левое поле – 2 см;
- правое поле – 1,5 см.

Для отображения границ зоны текста выполните команду Сервис→Параметры и на вкладке Вид включите флажок границы текста.

2 Сохраните созданный документ в своей папке (имя файла определите сами, например Lab_W_1).

3 Скопируйте в новый документ исходный текст с шестую страницы данного файла.

Для этого его надо выделить, скопировать в буфер и вставить в Ваш файл.

4 Отформатируйте текст в Вашем документе в соответствии с образцом на стр. 5.

5 Создайте элемент автотекста, включивший в него содержание третьего абзаца текста.

6 Создайте элемент автозамены, который при наборе Ваших инициалов вставляет в текст Вашу полную фамилию (например, ввод ИАН).

Название “**электронная вычислительная машина**” определено изначальной областью применения ЭВМ – выполнение инженерно-технических расчетов. Однако для современных ЭВМ больше подходит другое определение: *программно управляемая искусственная (инженерная) среда, предназначенная для восприятия, хранения обработки и передачи информации*

Устройства ввода и вывода обеспечивают ЭВМ связь с внешним миром.

Устройства ввода (например, клавиатура) осуществляют считывание информации и представлении считанной информации электрическими сигналами, воспринимаемые памятью.

Устройства вывода преобразуют электрические сигналы, поступающие из памяти и несущие в себе информацию о результатах обработки данных, в форму машинописного текста, графика, других изображений на экране, бумаге и т.п.

Память ЭВМ содержит программы, исходные данные, промежуточные результаты и другую информацию, необходимую ЭВМ для совершения различных (требуемых) операций и для взаимной связи между отдельными частями устройства. Вычисления, заданные программой, реализуются **центральный процессором**, или (когда термин не вызывает неопределенности) просто **процессором**. Он выбирает команды из памяти и выполняет действия, предписанные ими.

Редактор формул.**Основные положения.**

Запись и редактирование формул в Word ведется с помощью меню ВСТАВКА→УРАВНЕНИЕ. Процедура позволяет вставлять в документ математические формулы, включающие знаки операций, выражения, дроби, степени, интегралы и т. п.

При записи формулы используется автоматическое применение соответствующих стилей для различных ее компонентов (уменьшенный размер шрифта для показателей степени, курсив для переменных и т. п.).

Освоить работу с редактором сравнительно несложно, хотя сам процесс записи формул достаточно трудоемок. С другой стороны, следует отметить простоту изменения размеров готовых формул

Создать новый документ: «Lab_W_2», который должен содержать формулу из приложения 1 (вариант выбирается по согласованию с преподавателем).

$1. \int_0^{\pi/2} \sqrt{1 - \frac{1}{4} \sin^2 x} dx \cdot (\epsilon = \frac{1}{2} \cdot 10^{-4}). \square$	$2. \int_0^{\pi/2} \frac{\sin x}{x} dx \cdot (\epsilon = \frac{1}{2} \cdot 10^{-4}). \square$
$3. \int_0^{\pi/2} \sqrt{1 + \sin^2 x} dx \cdot (\epsilon = \frac{1}{2} \cdot 10^{-2}). \square$	$4. \int_1^9 \sqrt{6x - 5} dx \cdot (n = 8). \square$
$5. \int_{0,7}^{1,3} \frac{dx}{\sqrt{2x^2 + 0,3}} \cdot (n = 17). \square$	$6. \int_4^{5,2} \ln x dx \cdot (n = 6). \square$
$7. \int_0^{1,2} \ln(1 + x^2) dx \cdot (n = 6). \square$	$8. \int_0^1 e^{-x^2} dx \cdot (n = 10). \square$
$9. \int_0^1 \frac{\sin x}{x} dx \cdot (n = 10). \square$	$10. \int_0^1 \sin x^2 dx \cdot (n = 10). \square$
$11. \int_0^1 \cos x^2 dx \cdot (n = 10). \square$	$12. \int_0^1 \frac{x}{1+x} dx \cdot (n = 10). \square$
$13. \int_0^1 e^{x^2} dx \cdot (n = 10). \square$	$14. \int_2^6 \lg(x^2 + 3,5) dx \cdot (n = 8). \square$
$15. \int_0^{\pi/2} \sqrt{1 - \frac{1}{2} \sin^2 x} dx \cdot (n = 6). \square$	$16. \int_0^1 \frac{dx}{1+x} \cdot (\epsilon = \frac{1}{2} \cdot 10^{-4}). \square$

Рисунок 1.2 – Задания

Все подробности приведены в электронных методических рекомендациях.

Контрольные вопросы

- 1 Как заменить отображение страницы с книжной на альбомную?
- 2 Какие режимы отображения документа Вы знаете?
- 3 Как настроить масштаб отображения документа?
- 4 Как изменить количество документов в списке, предназначенном для их быстрого открытия?
- 5 Для чего предназначена функция быстрого сохранения файла?
- 6 Как настроить параметры автосохранения?
- 7 Как включить автоматическую замену «прямых» кавычек парными?
- 8 Как изменить единицы измерения для настройки параметров документа с сантиметров на миллиметры?
- 9 Как пронумеровать страницы?
- 10 Как отредактировать колонтитул?
- 11 Как разбить документ на разделы?
- 12 Каким образом можно соединить два раздела (найдите сами или используйте справку по Word)?

2 Лабораторная работа № 2. ТП Word. Создание, редактирование и форматирование таблиц

Цель работы: приобретение практических навыков по работе с таблицами в редакторе Microsoft Word.

Основные положения.

Таблицы предназначены для упорядочивания данных и создания интересных макетов страницы с последовательно расположенными столбцами текста или графики. Для работы с таблицами используется пункт меню Таблица.

Наиболее быстрый путь создания простой таблицы – например такой, которая имеет одинаковое количество строк и столбцов – с помощью команды Добавить таблицу.

Создание простой таблицы.

- 1 Выберите место создания таблицы.
- 2 Выберите команду Добавить таблицу из меню Таблица.
- 3 Установите нужные параметры (число столбцов и строк, ширина столбцов, автоформат таблицы).

Добавление строк и столбцов в таблицу.

Чтобы добавить строку в конец таблицы, щелкните последнюю ячейку последней строки, а затем нажмите TAB.

Чтобы добавить столбец справа от последнего столбца в таблице, щелкните за пределами самого правого столбца. Выберите команду Выделить столбец в меню Таблица, а затем нажмите команду Добавить столбцы.

Ввод с клавиатуры и перемещение по таблице.

Таблица 2.1 – Добавление строк и столбцов

ЧТОБЫ	НАЖМИТЕ КЛАВИШУ
Перейти в соседнюю ячейку	ТАВ (если курсор в последней ячейке таблицы, то нажатие ТАВ добавляет новую строку внизу таблицы)
Перейти в предыдущую ячейку	SHIFT+ТАВ
Перейти на предыдущую или следующую строку	СТРЕЛКА ВВЕРХ или СТРЕЛКА ВНИЗ
Перейти в первую ячейку в строке	ALT+HOME
Перейти в последнюю ячейку в строке	ALT+END
Перейти в первую ячейку в столбце	ALT+PAGE UP
Перейти в последнюю ячейку в столбце	ALT+PAGE DOWN

ССЫЛКА

Изменение ширины столбца или высоты строки таблицы.

Чтобы для нескольких столбцов (строк) установить одинаковую ширину (высоту), выделите нужные столбцы (строки), а затем выберите команду Выровнять ширину столбцов (Выровнять высоту строк) в меню Таблица.

Можно также изменить ширину столбцов (высоту строк) при помощи перетаскивания границ столбца (строки) в самой таблице или перетаскиванием маркеров границ столбцов (строк) таблицы на горизонтальной (вертикальной) линейке.

Отображение или скрытие сетки в таблице.

Выберите команду Отображать сетку или Скрыть сетку в меню Таблица.

Удаление из таблицы ячеек, строк или столбцов.

Выделите ячейки, строки или столбцы для удаления. При удалении ячеек включите режим отображения символов ячейки. При удалении строк включите режим отображения символов строк.

Выберите команду Удалить ячейки, Удалить строки или Удалить столбцы в меню Таблица.

При удалении ячеек выберите нужный параметр (со сдвигом влево или вправо, удалить всю строку или весь столбец).

Задание 1

Создайте расписание занятий своей группы на три дня недели в соответствии с вариантом в таблице 2.2 и указаниями к выполнению задания.

Результаты сохраните в рабочей папке в файле «Lab_W_2_1»

Таблица 2.2 – Варианты заданий

номер варианта	День недели		
1	Понедельник	Вторник	Среда
2	Пятница	Понедельник	Вторник
3	Понедельник	Вторник	Пятница
4	Четверг	Вторник	Среда
5	Понедельник	Среда	Пятница
6	Четверг	Пятница	Среда
7	Вторник	Среда	Четверг
8	Пятница	Вторник	Среда
9	Вторник	Среда	Понедельник
10	Пятница	Среда	Четверг
11	Вторник	Четверг	Понедельник
12	Вторник	Пятница	Понедельник
13	Среда	Четверг	Пятница
14	Пятница	Понедельник	Среда
15	Среда	Четверг	Вторник
16	Четверг	Пятница	Вторник
17	Среда	Пятница	Вторник
18	Среда	Понедельник	Вторник
19	Четверг	Пятница	Понедельник
20	Пятница	Понедельник	Четверг

Указания к выполнению задания 1.

1 Вставьте таблицу, состоящую из семи столбцов и семи строк, используя кнопку на панели инструментов или воспользовавшись командой Таблица → Добавить таблицу.

2 В первой строке объедините ячейки, принадлежащие соответственно второму и третьему, четвертому и пятому, шестому и седьмому столбцам, используя команду Объединить ячейки.

3 Аналогичным образом объедините ячейки первого столбца, принадлежащие первой и второй строкам.

4 Заполните расписание занятий в своей группе в соответствующие дни недели.

Часы	Понедельник		Вторник		Четверг	
	Нечетная неделя	Четная неделя	Нечетная неделя	Четная неделя	Нечетная неделя	Четная неделя
8.30-10.05	Математика		Иностранный язык			Белорусский язык
10.25-12.00	Геодезия		Экономика	ОИиВТ	Логика	История
12.20-13.55	Физкультура		ОИиВТ	Экономика	Начертательная геометрия	
14.15-15.50			Математика		Физкультура	
16.00-17.35						

Рисунок – 2.1 Исходная таблица

5 Ширину первого столбца грубо отрегулируйте, используя мышь при нажатой клавише {Ctrl} и точно отрегулируйте двойным щелчком на правой границе столбца.

6 Если занятия проводятся и в четную, и в нечетную неделю, то следует объединить соответствующие ячейки.

7 Не вводите по несколько раз названия повторяющихся предметов. Копируйте их.

8 Используя команду Автоформат, установите формат, указанный преподавателем.

9 Используя команду Границы и заливка меню Формат, введите в ячейки с наименованиями предметов фоновое оформление, указанное преподавателем.

10 Создайте таблицу изображенную на рисунке 2.1. Используя команды Удалить строки и Добавить строки, введите в таблицу переменую и вставьте столбец с номером пары, как показано на рисунке 2.2.

№ пары	Часы	Понедельник		Вторник		Четверг	
		Нечетная неделя	Четная неделя	Нечетная неделя	Четная неделя	Нечетная неделя	Четная неделя
1	8.30-10.05	Математика		Иностранный язык			Белорусский язык
Перемена!!!							
2	10.25-12.00	Геодезия		Экономика	ОИиВТ	Логика	История
Перемена!!!							
3	12.20-13.55	Физкультура		ОИиВТ	Экономика	Начертательная геометрия	
Перемена!!!							
4	14.15-15.50			Математика		Физкультура	
5	16.00-17.35						

Рисунок – 2.2 Измененная таблица

11 Скопируйте таблицу созданную изначально – рисунок 2.1. Выделите несколько столбцов, а затем удалите их, используя команду Удалить столбцы. В результате этой операции должно остаться расписание занятий одного дня. Пример показан на рисунке – 2.3 Занятие одного дня.

Часы	Понедельник	
	Нечетная неделя	Четная неделя
8.30-10.05	Математика	
10.25-12.00	Геодезия	
12.20-13.55	Физкультура	
14.15-15.50		
16.00-17.35		

Рисунок – 2.3 Занятие одного дня

Задание 2. Вычисления в таблицах

Основные сведения об использовании формул.

Ячейки таблицы имеют адреса, образованные именем столбца (А, В, С...) и номером строки (1, 2, 3...), например, А1, С4 и т.д.

Word позволяет выполнять вычисления, записывая в отдельные ячейки таблицы формулы с помощью команды Таблица\Формула... (Работа с таблицами – Макет ...). Формула задается как выражение, в котором использованы:

– абсолютные ссылки на ячейки таблицы в виде списка (разделяемые знаком ";") – А1; В5; Е10 и т.д.) или блока (начало и конец блока ячеек – А1:А10);

– ключевые слова для ссылки на блок ячеек:

а) LEFT – ячейки, расположенные в строке левее ячейки с формулой;

б) RIGHT – ячейки, расположенные в строке правее ячейки с формулой;

в) ABOVE – ячейки, расположенные в столбце выше ячейки с формулой;

г) BELOW – ячейки, расположенные в столбце ниже ячейки с формулой;

– константы – числа, текст в двойных кавычках;

– закладки, которым соответствует определенный текст документа (например, числа), созданный с помощью команды Правка\Закладка;

– встроенные функции Word, например, SUM(), AVERAGE();

– знаки операции (+, -, *, /, %, ^, =, <, >, <=, >=).

Практическое задание

Создайте таблицу, содержащую сведения об успеваемости.

Результаты сохраните в рабочей папке в файле «Lab_W_3».

Как показано на рисунке 2.4 – Сведения об успеваемости студентов произведите вычисления успеваемости студентов. Для облегчения адресации ячеек зеленым цветом указаны имена столбцов и номера строк.

Сведения об успеваемости студентов в 2021/2022 учебном году.

	А	В	С	Д	Е	Ф	Г	Н	І	Ј
1	№ п п	Учебная дисциплина	Группа	Средний балл	Всего сдало	Отл. .	Хор. .	Удовл. .	Неудовл. .	Неяк. и
2	1	Высшая математика	141			8	8	8	3	2
3	2	Информатика	142			12	10	6	3	1
4	3	Химия	143			7	9	6	3	2
5	4	История	144			9	8	3	5	3
6		Итого:								

Рисунок 3 – Сведения об успеваемости студентов

Технология выполнения .

1 Введите формулы для расчета числа студентов группы, сдавших экзамен по дисциплине «Высшая математика». Для этого установите курсор в ячейку E2 и введите формулу SUM(RIGHT), предварительно убрав имеющуюся в ней запись. Для ввода формулы используется меню Таблица\Формула...

2 Произведите те же действия для ячеек E3–E5.

3 Введите формулу для расчетов среднего балла по дисциплине «Высшая математика» для группы 091. Для этого установите курсор в ячейку D2 и введите формулу $= (F2*5+G2*4+H2*3+I2*2) / E2$. Выберите формат числа 0,00.

4 Введите аналогичные формулы в ячейки D3–D5.

5 Введите формулу для расчета общего числа студентов, сдавших экзамен на отлично, хорошо и т. д. Для этого установите курсор в ячейку F6 и введите формулу SUM(ABOVE), затем аналогично в ячейку G6 и т. д.

6 Выполните форматирование таблицы в соответствии с образцом.

Контрольные вопросы

- 1 Как вставить таблицу в документ Word, добавить строку, столбец?
- 2 Как выровнять текст в ячейках таблицы, изменить его направление?
- 3 Как разбить/объединить ячейки в таблице?
- 4 Как построить расчетную формулу в ячейке таблицы?
- 5 Как пересчитать таблицу после изменения в ней числовых значений?
- 6 Как удалить строку или столбец из таблицы Word?
- 7 Как выполнить сортировку данных в таблице?
- 8 Как вычислить сумму значений столбца/строки таблицы?

3 Лабораторная работа № 3. ТП Word. Слияние

Цель работы: создание типовых документов с использованием СЛИЯНИЯ.

С помощью слияния можно создавать документы типа стандартных писем, каждое из которых содержит общие сведения, повторяющиеся в каждом документе, и данные, которые в каждом документе уникальны. Команда Слияние позволяет объединять информацию, которая содержится в двух документах.

Основной документ – это файл, содержащий сведения, которые будут повторяться в каждом документе, письме, бланке, каталоге и т. д.

Источник данных – файл, который содержит изменяющиеся сведения, например, адрес и реквизиты каждого адресата на бланке. Источником данных может быть база данных, а также электронная таблица или таблица Word, оформленная как база данных.

Ключевыми понятиями базы данных, которые используются при организации объединения документов с помощью слияния, являются поля и записи.

Если представить базу данных в виде таблицы, то столбцы таблицы, имена которых указаны в первой строке (заголовок таблицы), называются полями. Расположенные ниже заголовка таблицы строки, в каждой которых содержится однородная информация об одном объекте, называются записями.

Для связи основного документа и источника данных используются имена полей слияния, которые вставляются в качестве меток в основной документ. При создании меток в качестве их имен используются имена полей из источника данных.

После слияния основного документа и источника данных каждая строка источника данных порождает отдельный документ на бланке, конверт, элемент каталога. Все отдельные документы объединяются в составной документ, который сохраняется в виде файла.

Пример документа, отдельные экземпляры которого могут быть получены с использованием слияния (красным цветом выделена переменная информация).

Белорусско-Российский университет

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №1

Дисциплина: Математическое моделирование

Весенняя экзаменационная сессия 2014 / 2015 учебного года



1. *Назначение Matlab. Рабочая среда. Типы файлов.*
2. *Массивы структур.*
3. *Практическое задание.*

Заведующий кафедрой _____

Преподаватель _____

Дата утверждения 12.05.2022 г. Пр. №11

Рисунок 3.1 – Задание

Задание 1

Создайте составной документ путем непосредственного выбора команд на панели инструментов слияния как пример рисунок 3.1 – задание.

Разработайте составной документ для создания извещений – рисунок 3.2 необходимому количеству студентов.

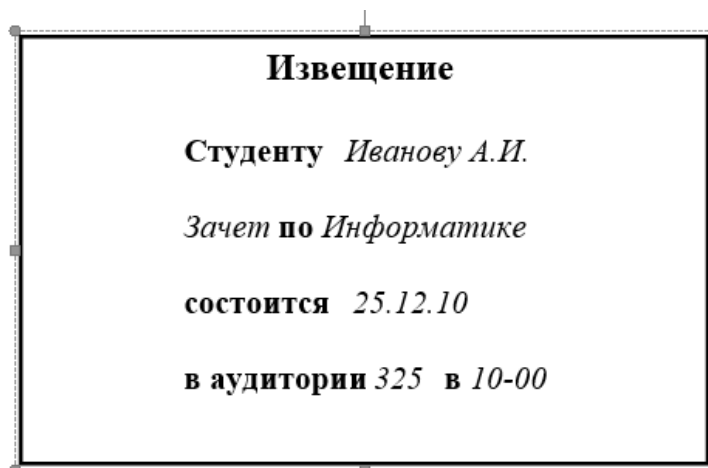


Рисунок 3.2 – Извещение

Жирным шрифтом выделены общие для всех документов данные, при создании составного документа их помещают в файле-основном документе.

Курсивом выделены данные, которые в каждом документе уникальны. Эти данные помещаются в файле источника данных. Источник данных имеет структуру таблицы базы данных (БД). В терминологии БД строки таблицы называются записями, а столбцы – полями.

Существуют различные варианты создания подобных документов. Далее рассмотрим создание составного документа с использованием панели инструментов слияния и с помощью мастера.

В основном документе, который будет использоваться для выполнения слияния, будут помещены метки, т.е. имена соответствующих полей из источника данных.

Далее будет описана последовательность подготовки основного документа и источника данных.

Порядок выполнения практического задания.

1 Создание источника данных как показано на рисунке 3.3 – таблица со структурой.

Создайте новый текстовый файл. Используя команду Таблица→Добавить, создайте таблицу следующей структуры.

№	ФИО	<u>Вид аттест</u>	Предмет	<u>Место пр</u>	Дата	Время
	Иванов А.И.	зачет	Информатика	325	25.12.09	10-00
	Ковалев П.Л.	Экзамен	Химия	412	06.01.10	9-00

Рисунок 3.3 – Таблица со структурой

Заполните в таблице 5–7 строк с информацией по Вашему выбору.

Сохраните в рабочей папке в файле под именем ИСТ_ДАННЫХ.

2 Создание основного документа.

Создайте новый текстовый файл и наберите текст. Пример текста возьмите у преподавателя.

В тексте приведены условные обозначения меток, которые будут вставлены в основной документ при объединении его с источником данных. В тексте на месте меток можно оставить пробелы, а выделенные красным цветом слова не вводить.

Сохраните в рабочей папке в файле под именем ОСН_ДОК.

Далее работу по слиянию документов производим с помощью панели инструментов Слияние. При отсутствии данной панели инструментов для ее вставки выполните Вид→Панели инструментов→Слияние.

3 Выполнение слияния.

1 Активным должен быть основной документ.

2 Используя кнопку Настройка главного документа, установите тип документа →Обычный документ Word.

3 Используя кнопку Открыть источник данных, откройте ранее созданный файл ИСТ_ДААННЫХ.

4 Используя кнопку Вставить поля слияния, расставьте метки в основном документе, выполняя следующие действия:

а) установить курсор в основном документе в место вставки метки;

б) открыть окно Добавление полей слияния, используя кнопку Вставить поля слияния ;

в) выбрать в списке имя нужного поля;

г) нажать кнопку Вставить и закрыть окно.

Для вставки следующей метки повторить шаги а)... г).

Исходные данные возьмите у преподавателя.

Контрольные вопросы

- 1 Что такое слияние?
- 2 Как формируется основной документ?
- 3 Для чего нужно слияние?
- 4 Продемонстрировать слияние по заданию преподавателя.
- 5 По шагам рассказать о формировании процесса слияния.

4 Лабораторная работа № 4. ТП Word. Форма

Цель работы: формирование навыков создания электронных форм, работы с шаблонами и полями. Закрепление навыков по использованию процедуры слияния.

Задание

Создайте электронную форму «Данные социологического опроса» согласно рисунку 4.1. – форма.

Данные социологического опроса студентов БРУ

ФИО

Факультет

Много ли новых фактов, ранее не известных вам, вы получили на практических занятиях?

Сколько раз вы сдавали экзамен?

Как часто вы используете полученные знания по информатике?

Какие побочные эффекты наблюдались у вас при изучении курса?

Головная боль Нервное расстройство Выпадение волос

Эйфория Просветление Провалы памяти

Рисунок 4.1 – Форма

Для создания формы выполните следующую последовательность действий:

- 1 Создайте в рабочей папке папку Эл_Форма.
- 2 Запустите Microsoft Word.
- 3 Создайте шаблон с именем Форма, разместите его в папку Эл_Форма.

Для этого:

- выберите команду Файл → Создать → Шаблон → Обычный → ОК.

Появится окно документа для нового шаблона.

- выберите команду в папку Файл → Сохранить как.

– в диалоговом окне Сохранить как введите имя нового шаблона: Форма, нажмите кнопку Сохранить (сохраните в папке Эл_Форма).

- 4 Далее создайте макет формы:

Введите заголовок формы. Введите неизменяемый текст: заголовок, ФИО, факультет, формулировку четырех вопросов и возможные варианты ответов на четвертый вопрос по образцу, выбрав нужный размер и начертание шрифта.

- 5 Подведите курсор к позиции начала текстового поля в графе «ФИО», щелкните по кнопке текстового поля панели инструментов Формы. Откройте диалоговое окно Параметры поля формы (кнопка на панели инструментов Формы или контекстное меню). В данном случае параметры диалогового окна можно оставить принятыми по умолчанию, убедитесь только, что стоит отметка в окне Разрешить изменения, и щелкните по кнопке ОК. Аналогичным образом создаются остальные тестовые поля.

6 Введите поле со списком в графе «Факультет» щелчком по третьей кнопке. Поле со списком панели инструментов Формы и сразу откройте диалоговое окно Параметры поля со списком. Введите в поле Элемент списка название очередного факультета и щелкните по кнопке Добавить. Повторите эти действия для каждого элемента списка.

Значением поля списка заданным по умолчанию, является первый введенный элемент, однако очередность можно изменить кнопками Порядок.

7 Разместите шесть полей типа Флажок после четвертого вопроса анкеты. Для вставки полей щелкните по второй кнопке панели инструментов Формы. В диалоговом окне Параметры флажка можно лишь увеличить его размеры.

8 Установите защиту шаблона от изменений. Для этого щелкните по кнопке Защита панели инструментов Формы. Сохраните форму в своей папке в папке Эл_Форма:

– выберите команду в папке Файл → Сохранить как;

– в диалоговом окне Сохранить как введите имя нового шаблона: Форма, а в графе Тип Файла выберите Шаблон документа.

9 Введите в поля формы ответы, руководствуясь собственным опытом изучения информатики. документ в своей папке.

Дополнительные сведения по работе получите у преподавателя.

Контрольные вопросы

- 1 Как создать форму?
- 2 Как модернизировать форму?
- 3 Как перемещаться по форме?

5 Лабораторная работа № 5. ТП Word. Макросы

Цель работы: изучение макросов.

Запись макроса.

Выберите команду Запись макроса в меню Разработчик. В поле Имя макроса введите имя нового макроса.

Выберите в поле Макрос доступен для шаблона, или документ, в котором будет храниться макрос.

При желании введите краткое описание макроса в поле Описание.

Если макросу не надо присваивать кнопку на панели инструментов, команду меню или сочетание клавиш, нажмите кнопку ОК, чтобы начать запись макроса.

Действия, сделанные в окне документа с помощью мыши, не записываются. Используйте клавиатуру для записи таких действий, как выделение текста или перемещение курсора. Мышь может быть использована для выбора нужных команд и параметров.

- 1 Для приостановки записи макроса нажмите кнопку Пауза.

- 2 Выполните действия, которые не должны быть записаны.
- 3 Для возобновления записи нажмите кнопку **Возобновить запись**.
- 4 Для окончания записи макроса нажмите кнопку **Остановить запись**.

Использование макросов.

После записи макроса следует проверить его работоспособность. Для этого в окне диалога «Макрос» выберите записанный макрос и нажмите кнопку **Выполнить**.

Кроме рассмотренного, для запуска макроса можно использовать следующие варианты:

- назначьте макросу комбинацию клавиш;
- создайте специальную кнопку на панели инструментов;
- создайте собственную команду меню.

Назначение макросу кнопки запуска.

Если Вы находитесь в окне диалога «Запись макроса», нажмите кнопку **Панели**. В качестве альтернативного способа выберите команду **Сервис / Настройка**. Откроется окно диалога «Настройка», вкладка «Панель инструментов».

В списке элементов Категории выберите **Макросы**, поскольку требуется назначить кнопку панели для созданного макроса. В списке имен выберите имя созданного макроса. При нажатии мышью на выделенную область появляются очертания кнопки, которые следует переместить на панель инструментов.

Задание

Создайте макросы, выполняющие:

- 1 построение заголовка и шапки таблицы;
- 2 добавление в текст таблицы с заданным количеством столбцов и строк;
- 3 вычисление по формуле значений в ячейках таблицы;
- 4 вычисление итоговых значений в таблице;
- 5 вставку в текст заданного текстового фрагмента с указанными параметрами форматирования.

Подробную информацию и задание получите у преподавателя.

Контрольные вопросы

- 1 Что такое макрос?
- 2 Как создать макрос?
- 3 Как вызвать созданный макрос (4 способа)?
- 4 Как создать кнопку на панели инструментов для макроса?
- 5 Как просмотреть макрос (текст макроса)?

6 Лабораторная работа № 6. ЭТ Excel. Элементы окна. Настройка панели инструментов. Ввод и редактирование данных

Цель работы: создание таблиц, форматирование. Ввод и копирование формул.

1 Начало работы.

1.1 Запустите программу Excel.

1.2 Создайте новую рабочую книгу.

1.3 Лист 1 переименуйте, присвойте имя Расходы_Топливо

1.4 Сохраните созданную книгу в рабочей папке, имя файла Lab_1_EX

2. Создание таблицы

2.1 На листе Расходы_Топливо создайте таблицу по образцу, приведенному на рисунке 6.1 – расчет расходов.

Расчет расходов на топливо						
№ п/п	Марка автомобиля	Вид топлива	Цена 1л топлива	Расход топлива на 100 км	Пробег	Стоимость топлива
Итого						

Рисунок 6.1 – Расчет расходов

Для оформления строки заголовков используйте команду **Формат** → **Ячейки**.

2.2 Введите не менее пяти записей.

2.3 В столбце **Стоимость топлива** расходы вычислите, используя формулы. (Стоимость топлива = Цена 1л · Расход на 100 км · Пробег / 100;

В строке **Итого** в столбце **Стоимость топлива** введите формулу суммы значений всех записей).

2.4 Установите денежный формат ячеек соответствующих столбцов. Точность отображения чисел до двух знаков после запятой.

2.5 Оформите таблицу, используя различные цвета.

2.6 Выполните обрамление таблицы.

2.7 Сохраните файл Lab_1_EX.

3 Работа с листами рабочей книги.

Создайте электронные таблицы для расчета оплаты труда рабочих строительной бригады пропорционально отработанному времени, разряду и тарифам. Данные для расчетов разместите на разных листах.

3.1 Переименуйте листы рабочей книги в соответствии с листом Excel. Для добавления листа используйте команду Вставка – Лист, для вызова которой используйте меню приложения или контекстное меню (освойте оба варианта).

3.2. Создайте структуру таблиц для последующего внесения данных и формул.

На листе Начисления создайте таблицу, структура которой представлена на рисунке 6.2 – расчет заработной платы.

Расчет заработной платы							
№	ФИО	Разряд	Отработано часов	Начислено за отработанное время	Премия	Всего начислено	Налог

Рисунок 6.2 – Расчет заработной платы

Вся остальная информация имеется в электронном варианте.

Контрольные вопросы

- 1 Назначение электронных таблиц.
- 2 Как определить какая ячейка текущая?
- 3 Как адресуются ячейки электронной таблицы?

7 Лабораторная работа № 7. ЭТ Excel. Использование встроенных функций

Цель работы: использование встроенных функций.

При выполнении практических заданий необходимо:

- создать файл и сохранить в рабочей папке;
- каждое задание выполнить на отдельном рабочем листе, название которого выбирать самостоятельно.

Для уточнения, в случае необходимости, синтаксиса функций, используемых в выражениях, используйте информацию, приведенную в справочной системе excel.

При выполнении заданий будут использоваться логические функции. Далее приведена краткая информация по синтаксису и особенностях применения некоторых логических функций.

Логические функции ЕСЛИ, И, ИЛИ
ЕСЛИ()

Синтаксис:

ЕСЛИ(логическое_выражение; значение_если_истина; значение_если_ложь).
И()

Синтаксис:

Логич_знач И(логич_знач1; логич_знач2; ... ; логич_знач30)

Предназначение: функция И используется тогда, когда нужно проверить, выполняются ли несколько условий **ОДНОВРЕМЕННО**.

Аргументы: функция И принимает от 1 до 30 аргументов (в Office 2007 -- до 256), каждый из которых является логическим значением ЛОЖЬ или ИСТИНА, либо любым выражением или функцией, которое в результате дает ЛОЖЬ или ИСТИНА.

Возвращаемое значение: функция И возвращает логическое значение. Если **ВСЕ** аргументы функции И равны ИСТИНА, возвращает ИСТИНА. Если хотя бы один аргумент имеет значение ЛОЖЬ, возвращает ЛОЖЬ.

Примечание: функция И почти никогда не используется сама по себе, обычно её используют в качестве аргумента других функций, например, ЕСЛИ.

Пример:

=ЕСЛИ(И((B1>1);(C1<5));12;24) – в ячейке, содержащей формулу, устанавливается значение 12, если (B1>1) и (C1<5), при невыполнении хотя бы одного из двух условий значение в ячейке -24.

ИЛИ()

Синтаксис:

Логич_знач ИЛИ(логич_знач1; логич_знач2; ... ; логич_знач30)

Предназначение: функция ИЛИ используется тогда, когда нужно проверить, выполняется ли **ХОТЯ БЫ ОДНО** из многих условий.

Аргументы: функция ИЛИ принимает от 1 до 30 аргументов (в Office 2007 – до 256), каждый из которых является логическим значением ЛОЖЬ или ИСТИНА, либо любым выражением или функцией, которое в результате дает ЛОЖЬ или ИСТИНА.

Возвращаемое значение: функция ИЛИ возвращает логическое значение. Если **ХОТЯ БЫ ОДИН** аргумент имеет значение ИСТИНА, возвращает ИСТИНА. Если **ВСЕ** аргументы имеют значение ЛОЖЬ, возвращает ЛОЖЬ.

Примечание: функция ИЛИ почти никогда не используется сама по себе, обычно её используют в качестве аргумента других функций, например, ЕСЛИ.

2.1 Вычислите выражения при различных значениях аргументов. Для этого на листе Excel создайте таблицу, в структуре которой предусмотрите диапазоны для ввода значений аргументов и значений вычисленных выражений. Для организации ветвления используйте логические функции ЕСЛИ(), И(), ИЛИ().

Задание получите у преподавателя.

2.2 Одномерные массивы.

Создать в диапазоне ячеек А2-А10 массив № 1, В2-В10 массив № 2 и найдите для этих массивов значения статистических функций (медианы, максимум, среднее арифметическое (СРЗНАЧ), среднее геометрическое (СРГЕОМ) и корреляцию (КОРРЕЛ)).

2.3 Двухмерные массивы.

Создайте в диапазонах ячеек две матрицы 4×4 и найдите для этих массивов значения математических функций:

- обратную матрицу;
- произведение матриц;
- определитель матрицы.
- транспонируйте матрицы с помощью функции ТРАНСП() (из категории ССЫЛКИ И МАССИВЫ).

Задание получите у преподавателя.

Контрольные вопросы

- 1 Особенности ввода формул в ячейки рабочего листа. Компоненты формул.
- 2 Создание формул массива.
- 3 Что такое функция и как её можно задавать?
- 4 Как и когда использовать функцию автозаполнения?

8 Лабораторная работа № 8. ЭТ Excel. Диаграммы. Задача о траектории движения тела

Цель работы: изучение построения диаграмм в электронной таблице Excel.

Для построения диаграмм на основании табличных данных используется Мастер диаграмм.

Используется информация, содержащаяся в столбцах или строках таблицы. По терминологии используемая информация образует ряды данных, которые могут быть смежными и образовывать сплошной диапазон (все материалы взять у преподавателя) или несмежными (все материалы взять у преподавателя).

Задание.

Постройте график функции $y = x^2$.

Последовательность выполнения задания и требования к оформлению.

- 1 Запустите Excel.
- 2 Выберите щелчком рабочий листа и переименуйте его как Графика_1.
- 3 В столбец А, в ячейку А1 введите имя столбца Х. Начиная с ячейки А2 введите 8–10 произвольных возрастающих значений независимой переменной. (Например, прогрессию от 0 с шагом 0,1).
- 4 В ячейку В1 введите имя столбца Y. В ячейку В2 внесите формулу $y=x^2$ ($=A2^2$). Копируйте формулу в остальные ячейки столбца.

5 Выделите все заполненные ячейки столбцов А и В.

6 Щелкните на значке Мастер диаграмм на панели инструментов.

7 В списке Тип выберите пункт Точечная (для отображения графика, заданного парами значений). В палитре Вид выберите средний пункт в первом столбце (маркеры, соединенные гладкими кривыми). Щелкните на кнопке Далее.

8 Так как диапазон ячеек был выделен заранее, мастер диаграмм автоматически определяет расположение рядов данных. Убедитесь, что данные на диаграмме выбраны правильно. На вкладке Ряд в поле Имя укажите: Результаты измерений. Щелкните на кнопке Далее.

9. Выберите вкладку Заголовки. Убедитесь, что заданное название ряда данных автоматически использовано как заголовок диаграммы. Замените его, введя в поле название диаграммы заголовок Экспериментальные данные. Щелкните на кнопке Далее.

10 Установите переключатель Имеющемся. Щелкните на кнопке Готово.

11 Убедитесь, что диаграмма построена и внедрена на рабочий лист. Щелкните на построенной кривой, чтобы выделить ряд данных.

12 Выполните команду Формат > Выделенный ряд. Откройте вкладку Вид.

13 На панели Линия откройте палитру Цвет и выберите красный цвет. В списке Тип линии выберите пунктир.

14 На панели Маркер выберите в списке Тип маркера треугольный маркер. В палитрах Цвет и Фон выберите зеленый цвет.

15 Щелкните на кнопке ОК, снимите выделение с ряда данных и посмотрите, как изменился вид графика (пример диаграммы, построенный в соответствии с вышеизложенными требованиями).

16 Сохраните рабочую книгу в рабочей папке. Имя файла Диаграмма.

Контрольные вопросы

1 Как построить диаграмму?

2 Как задать диапазон данных?

3 Основные компоненты диаграмм.

4 Как редактировать компоненты диаграмм?

9 Лабораторная работа № 9. ЭТ Excel. Решение уравнений и систем уравнений. Задачи оптимизации

Цель работы: изучение команд «Подбор параметра» и «Поиск решения».

Надстройка «Подбор параметра» обеспечивает вычисление значения аргумента (параметра) для заданного значения функции методом последовательных итераций (подстановок).

Процедура «Поиск решения» позволяет найти оптимальное значение формулы, содержащейся в ячейке, которая называется целевой. Эта процедура

работает с группой ячеек, прямо или косвенно связанных с формулой в целевой ячейке (изменяемые ячейки). Чтобы получить по формуле, содержащейся в целевой ячейке, заданный результат (максимальное, минимальное или заданное значения), процедура изменяет значения во влияющих ячейках, учитывая ограничения.

Задача.

Найти решение уравнения $x^3 - 3x^2 + x + 1 = 0$.

При использовании этого метода в одну ячейку вводится формула, вторая ячейка используется для искомого решения.

На рисунке 9.1 – уравнение представлен один из вариантов размещения данных на листе Excel.

x	F(x)
	1,00

Рисунок 9.1 – Уравнение

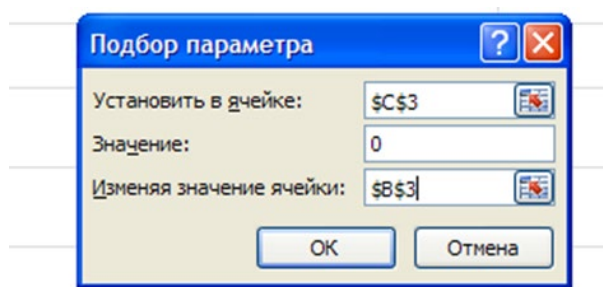


Рисунок 9.2 – Подбор параметров

Подробную информацию возьмите у преподавателя.

Задача. Решить систему уравнений как приведено на рисунке 9.3 – уравнение.

$$\begin{cases} x + y + z = 6 \\ 2x - y + z = 3 \\ x - 4y + 3z = 2. \end{cases}$$

Рисунок 9.3 – уравнение.

При использовании данной команды на листе Excel необходимо предусмотреть диапазоны для размещения формул уравнений системы, правых

частей уравнений, искомым значений неизвестных, а также поясняющих обозначений. В нижеприведенном варианте в ячейках A1–A3 обозначения переменных в соответствующих строках столбца B.

1 В ячейки B1–B3 внесите начальные приближения переменных, например, 0.

2 В столбец D внесите левые части уравнений, ссылаясь на введенные значения.

3 В столбец E внесите правые (известные) части уравнений. В результате получится следующее (рисунок 9.4).

	A	B	C	D	E
1	x	0		=B1+B2+B3	6
2	y	0		=2*B1-B2+B3	3
3	z	0		=3*B3-4*B2+B1	2

Рисунок 9.4 – Таблица

Одно из введенных уравнений принимается за целевую функцию. В рассматриваемой задаче первое. И, соответственно, адрес ячейки с этим уравнением указываем в качестве целевой, равной значению правой части этого уравнения.

Изменяя ячейки, указываем диапазон, отведенный под искомые значения неизвестных. И, наконец, в качестве ограничений указываем диапазон размещения остальных уравнений и диапазон правых частей уравнений.

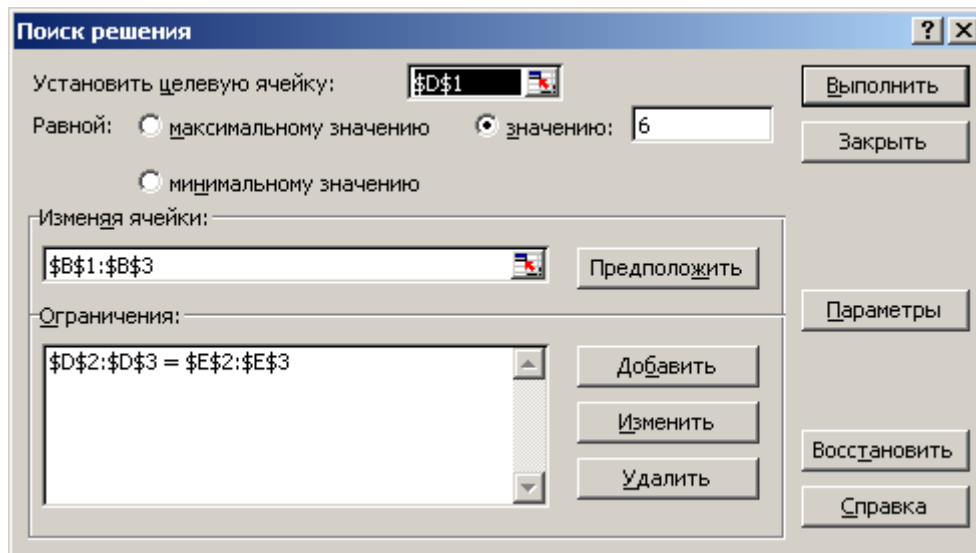


Рисунок 9.5 – Поиск решения

Подробную информацию возьмите у преподавателя.

Контрольные вопросы

- 1 Ввод и редактирование ограничений.
- 2 Для чего служит Кнопка Параметры?
- 3 В поле Ссылка на ячейку вводится адрес или имя ячейки или диапазона?
- 4 Для чего служат поля Ограничения?

10 Лабораторная работа № 10. ЭТ Excel. Методы работы с базами данных в приложении

Цель работы: повышение уровня понимания темы «Базы данных в приложении Microsoft Excel». Овладение специальными техническими умениями конструирования и использования реляционной базы данных на уровне их свободного использования. Развитие навыков самостоятельной работы и способности применить полученные знания на практике при разработке собственной базы данных.

Списки Excel как база данных.

Приложение Microsoft Excel обладает богатыми встроенными средствами для обработки и анализа данных. Аналогом простой базы данных в Excel служить список.

Список – это группа строк таблицы, содержащая связанные данные.

Отличительной особенностью списка является то, что каждый его столбец содержит однотипные данные, например, перечень фамилий, дату рождения и т. д.

Если провести аналогию между списком и табличной базой данных, то столбцы списка являются полями базы данных, а его строки – записями. Считается, что первая строка списка является его заголовком и содержит названия столбцов списка. Заголовок должен иметь на листе электронных таблиц горизонтальную ориентацию. Заголовки применяются Excel при составлении отчетов, а также при поиске и организации данных. Шрифт, размер шрифта, выравнивание и другие параметры форматирования, присвоенные заголовкам столбцов списка, должны отличаться от параметров, назначенных для строк данных. В списке не должно быть пустых строк и столбцов.

Проверка данных при вводе.

Если с файлом работает сразу несколько пользователей, желательно контролировать тип вводимой ими информации и свести к минимуму ошибки ввода. В Excel выполнение подобных условий проверяется при помощи средства, которое называется проверкой ввода. Для этого надо:

- 1) выделить ячейки столбца, для которого устанавливается проверка ввода;
- 2) на ленте Данные в группе Работа с данными выбрать команду Проверка данных;
- 3) на вкладке Параметры в области Условие проверки выбрать Тип данных: Любое значение (используется для отмены проверки ввода), Целое

число, Действительное, Список, Дата, Время, Длина текста и Другой (формат, для которого можно задать собственную формулу, например, "м" or "ж"). При выборе значения внизу окна появляются дополнительные поля для ввода условий или ограничений, например, минимального и максимального допустимых значений;

4) на вкладке Сообщение для ввода можно установить флажок Отображать подсказку, если ячейка является текущей, и ввести сообщение, чтобы оно появлялось на экране при выделении ячеек;

5) на вкладке Сообщение об ошибке можно установить флажок Выводить сообщение об ошибке, чтобы задать тип сообщения об ошибке, которое появится при вводе в ячейку недопустимого значения.

Сортировка данных.

Команда Сортировка позволяет переставить записи в другом порядке на основании значений одного или нескольких столбцов. Записи сортируются по возрастанию/убыванию или по выбранному пользователем порядку (например, по дням недели).

Чтобы отсортировать список надо:

1) установить курсор в ячейку списка;
2) выполнить команду Сортировка на ленте Данные в группе Сортировка и Фильтр;

3) В диалоговом окне Сортировка выбрать поле, по которому будет происходить сортировка; тип сортировки (по значению, цвет ячейки, цвет шрифта, значок ячейки) и порядок (по возрастанию, убыванию, настраиваемый). Пример приведен на рисунке – 10.1).

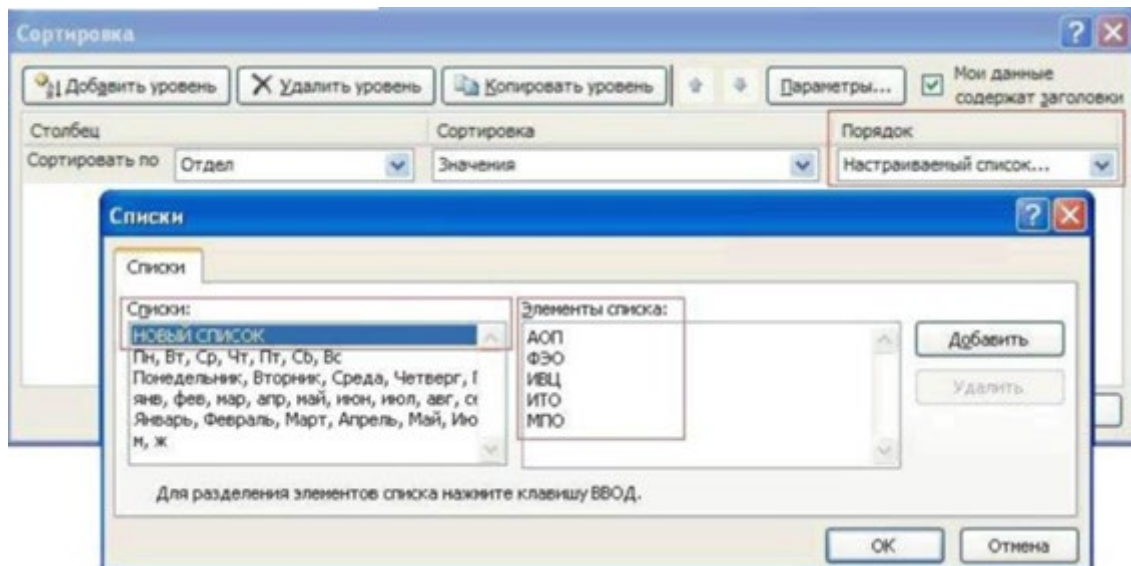


Рисунок 10.1 – Создание пользовательского порядка сортировки

Примечание – Выбор настраиваемого порядка позволяет задать нестандартный порядок сортировки. Для этого надо в диалоговом окне. Списки выбрать Новый список, в поле Элементы списка ввести значения, образующие

пользовательский порядок сортировки (например, АОП, ФЭО, ИВЦ, ИТО, МПО), после чего последовательно выбрать кнопки Добавить и ОК.

Подробную информацию и задания возьмите у преподавателя.

Контрольные вопросы

- 1 Для каких целей применяются электронные таблицы?
- 2 Для чего необходимы базы данных?
- 3 Какие возможности предоставляет программа Microsoft Excel для работы с базами данных?
- 4 Что такое запись?
- 5 Что такое поле?
- 6 Что такое тип данных? Зачем необходимо указывать типы полей?
- 7 Данные каких типов могут быть записаны в ячейку?
- 8 Как установить проверку вводимых в список значений?
- 9 Как записываются абсолютные и относительные адреса ячеек?
- 10 Что такое сортировка? Как отсортировать список по двум и более ключам?
- 11 Что такое фильтр?
- 12 Какие виды фильтров Вы знаете?
- 13 В чем отличие сортировки списка от фильтрации списка?
- 14 Что такое расширенный фильтр?
- 15 Как сформировать критерий для расширенного фильтра?

11 Лабораторная работа № 11. VBA. Элементы управления

Цель работы: ознакомление с интерфейсом VBA, некоторыми его объектами, их свойствами и методами.

VBA относится к языкам объектно-ориентированного программирования (ООП). ООП можно описать как методику анализа, проектирования и написания приложений с помощью объектов. Объект – комбинация кода и данных, рассматривается как единое целое, например, элемент управления, форма и компонент приложения. Каждый объект определяется по принадлежности к классу. Примеры объектов: рабочий лист (Worksheet), диапазон (Range), диаграмма (Chart), форма (UserForm) и т. д.

Доступ к данному языку программирования можно осуществлять практически из любого приложения Windows. Мы будем работать вместе с Microsoft Excel, который будет являться основным приложением для проекта VBA.

Объектами в Excel называется всё то, на что Вы можете указать. Например, это рабочие листы, ячейки, меню, кнопки и диапазоны ячеек. Ссылка на объект часто состоит из нескольких компонент, каждая из которых ограничивает область возможных объектов, пока не будет указан точно

конкретный объект. Каждая ступень ограничения области отделяется точкой (.). Например, запись

Workbooks("Книга2").Worksheets("Лист1")

позволяет получить ссылку на рабочий лист с именем Лист1, который находится в рабочей книге с именем Книга2.

То есть среди множества объектов типа рабочих книг (Workbooks) Excel нас интересует конкретный объект с именем, указанным в виде строковой константы "Книга2", а среди множества объектов типа рабочих листов (Worksheets) этой книги мы хотим получить ссылку (указатель) на конкретный лист с именем, указанным в виде строковой константы "Лист1".

Свойства – это атрибуты, описывающие строго определенные характеристики объекта. Так, ячейка имеет такие свойства, как имя, формат, значение. Изменяя свойства, Вы можете менять те или иные характеристики объекта. Например, чтобы рабочий лист «Лист1» из рабочей книги «Книга2» переименовать на «Мой лист» надо свойству Name этого листа присвоить новое значение:

Workbooks("Книга2").Worksheets("Лист1").Name="Мой лист"

Методы – выполняют определенные действия с объектами. Например, выделяют, удаляют, создают и изменяют объекты. Так, для очистки содержимого ячейки B6 на листе «Лист1» рабочей книги «Книга2» можно использовать метод Clear :

Workbooks("Книга2").Worksheets("Лист1").Range("B6").Clear

При этом свойство Range («имя ячейки») возвращает ссылку на указанную ячейку, к которой и применяется метод Clear.

Если надо очистить ячейку B6 активного листа текущей рабочей книги, то указывать Workbooks(«Книга2»).Worksheets(«Лист1») не надо, и метод по умолчанию будет применен к ячейке активного листа

Range("B6").Clear

Для того, чтобы среди множества листов рабочей книги «Финансы» активным сделать лист с именем «Квартал_3», можно использовать метод Activate:

Workbooks("Финансы").Worksheets("Квартал_3").Activate

Для добавления в Excel новых экземпляров объектов того или иного типа используется метод Add. Например, его можно использовать для добавления нового листа в рабочую книгу. При этом можно указать аргументы, которые определяют тип вставляемого листа (рабочий или модульный), место вставки и количество вставляемых листов. Синтаксис метода Add для объекта Worksheets имеет вид:

Worksheets.Add ([Before], [After], [Count], [Type])

Аргументы Before и After позволяют указать, где должен появиться новый лист. Если Вы указываете аргумент Before, то лист добавляется перед указанным листом (по умолчанию – перед активным). Аргумент Count указывает количество добавляемых листов (по умолчанию 1). Аргумент Type определяет тип добавляемого листа (по умолчанию – рабочий лист). При вызове надо присвоить (знак ":=") значения тем аргументам, которым считаете нужным точно указать, а остальные будут использованы по умолчанию. Например, добавим два новых рабочих листа перед листом "Лист3":

Worksheets.Add Count:=2, Before:=Sheets("Лист3")

Если метод Add используется для ссылки на вновь созданный объект и является частью оператора VBA, то аргументы необходимо заключать в скобки. Например, добавим в рабочую книгу после листа "Квартал_3" новый лист и присвоим ему имя "Квартал_4":

Worksheets.Add(After:=Sheets("Квартал_3")).Name = "Квартал_4"

Редактор VBA. Окно проекта.

Окно проекта в редакторе VBA активизируется выбором команды View►Project Explorer (Ctrl+R) или нажатием кнопки Project Explorer.

В проекте автоматически создается модуль для каждого рабочего листа и для всей книги. Кроме того, модули создаются для каждой пользовательской формы макросов и классов. По своему предназначению модули делятся на два типа – модули объектов и стандартные. К стандартным модулям относятся те, которые содержат макросы. Такие модули добавляются в проект командой Insert►Module (Вставка ►Модуль). К модулям объектов относятся модули, связанные с рабочей книгой, рабочими листами, формами, и модули класса.

Формы создаются командой Insert►UserForm, а модули класса – командой Insert►Class Module (Вставка►Модуль класса). По мере создания добавления и удаления файлов из проекта эти изменения отображаются в окне проекта.

Для создания диалоговых окон разрабатываемых приложений в VBA используются формы. Редактор форм является одним из основных инструментов визуального программирования. После добавления формы в проект на экран выводится незаполненная форма с панелью инструментов Toolbox (Панель элементов).

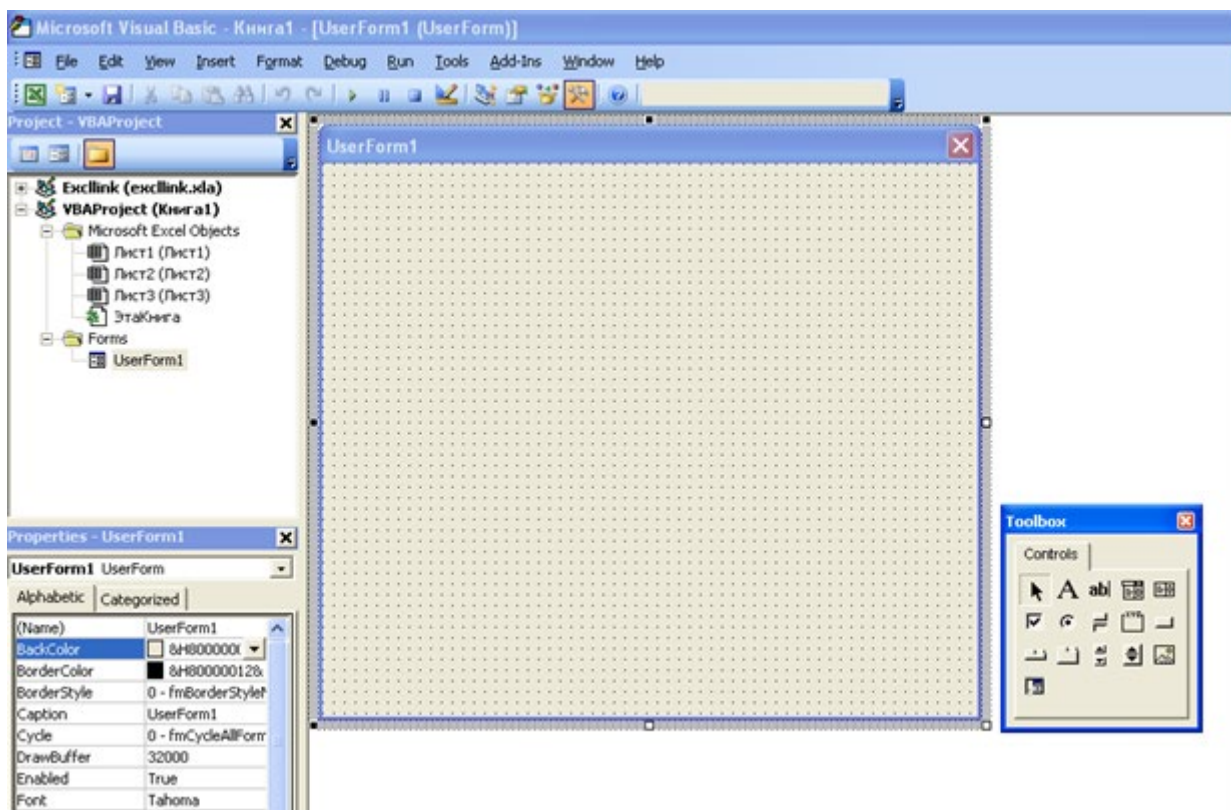


Рисунок 11.1 – Окно проекта в редакторе

Используя панель инструментов ToolBox (Панель элементов), можно сконструировать любое требуемое для приложения диалоговое окно.

Пример создания линейной программы.

Пример выполнить самостоятельно, далее возможно использование для выполнения индивидуального задания.

Для ввода исходных данных, управления ходом решения и вывода результатов используется пользовательская форма (USER FORM).

Задача: Найти сумму $a + b = c$.

Порядок выполнения работы:

- 1) открыть окно редактора VBA (Alt+F11.....)
- 2) выполнить команду ВСТАВКА(Insert)/USER FORM
- 3) поместить на форму элементы, требуемые для решения задачи, с панели элементов и расположить их нужным образом.

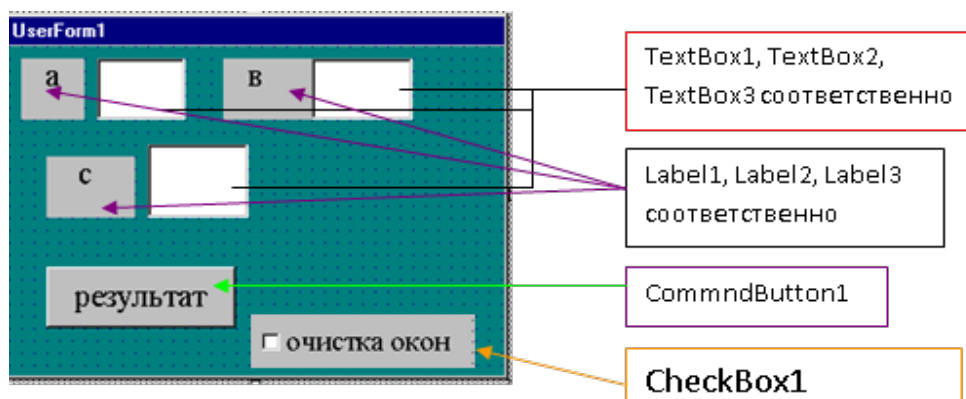


Рисунок 11.2 – UserForm

Вся подробная информация находится у преподавателя.

Контрольные вопросы

- 1 Что такое линейная программа?
- 2 Что происходит, если программа пишется под форму?
- 3 Ка можно изменить свойства объектов на форме?
- 4 Что означает запись «Dim a As Integer»?
- 5 Что означает запись «CInt»?
- 6 Объяснить синтаксис «CheckBox1.Value = False».

12 Лабораторная работа № 12. VBA. Оператор условного перехода. Объекты Excel

Цель работы: разработка программ разветвляющейся структуры с условным оператором IF. Программирование с использованием объектов excel.

Условный оператор позволяет выбирать и выполнять действия в зависимости от истинности некоторого условия. Имеется два варианта синтаксиса. В первом случае он имеет вид:

IF<условие>Then[операторы 1] [Else [операторы 2]]

Во втором случае оператор расположен на нескольких строках:

```
IF<условие>Then
  [операторы]
  [ElseIf<условие – n>Then
    [операторы-n ]...
  [Else]
  [ИначеОператоры]]
EndIf
```

Здесь условие обязательно в обоих вариантах. Оно может быть числовым или строковым выражением со значениями TRUEили FALSE.

Операторы 1 и операторы 2 – это последовательности из одного или нескольких операторов. По крайней мере одна из этих последовательностей должна быть непустой. Если условие истинно (TRUE), выполняется последовательность «операторы 1», если ложно – «операторы 2».

Примеры записи оператора.

Условный оператор в виде одной строки:

```
IF A>10 Then A=A+1: B=B+A: C=C+B ELSE C=A*B: A=C+2
```

Тот же условный оператор в виде блока:

```
IF A>10 Then
A=A+1
B=B+A
C=C+B
ELSE
C=A*B: A=C+2
```

EndIf

Далее рассматриваются примеры использования условного оператора и задания для выполнения.

Задание

Определить положение точки с заданными координатами на плоскости.

Пример 1 (выполнить).

Если точка $M(x_1, y_1)$ лежит:

- выше прямой $y=kx+b$, то выдать сообщение «Точка М лежит выше прямой»;
- ниже прямой, то выдать сообщение «Точка М лежит ниже прямой»;
- на прямой, то выдать сообщение «Точка М лежит на прямой».

Решение

1 этап. Разработать интерфейс приложения.



Рисунок 12.1 – Интерфейс приложения

2 этап. Определить ход решения задачи и составить алгоритм программы. В нашей задаче, чтобы определить положение точки относительно прямой, нужно подставить координату точки X_1 в уравнение прямой и сравнить полученное значение Y_c координатой точки Y_1 . Если значение $Y=Y_1$, точка лежит на прямой; если полученное значение $Y>Y_1$, точка – ниже прямой, и, наконец, если $Y<Y_1$, точка – выше прямой.

Схема алгоритма программы.

3 этап. Загрузить редактор VBA и разработать приложение, используя в качестве интерфейса UserForm.

4 этап. Написать программный код. Для этого рекомендуется выполнить двойной щелчок по кнопке СТАРТ и перейти в окно программы, где набрать текст процедуры обработки события.

Примерная структура процедуры приведена далее. Приведены пронумерованные блоки выражений и операторов. Для вывода сообщения использована функция MsgBox.

Подробная информация и задания находится у преподавателя.

Контрольные вопросы/задания

- 1 Даны две точки $A(x_1, y_1)$ и $B(x_2, y_2)$. Составить алгоритм, определяющий, которая из точек находится ближе к началу координат.
- 2 Подсчитать количество отрицательных чисел среди чисел a, b, c .
- 3 Перераспределить значения переменных x и y так, чтобы в x оказалось большее из этих значений, а в y – меньшее.
- 4 Заданы размеры A, B прямоугольного отверстия и размеры X, Y, Z кирпича. Определить, пройдет ли кирпич через отверстие.

- 5 Составить программу, осуществляющую перевод величин из радианной меры в градусную или наоборот. Программа должна запрашивать, какой перевод нужно осуществить, и выполнять указанное действие.

13 Лабораторная работа № 13. VBA. Оператор выбора With и Select Case

Цель работы: ознакомление с инструкцией With и Select Case. Закрепление полученных знаний на практике.

Select Case выполняет одну из нескольких групп инструкций в зависимости от значения выражения.

```
Синтаксис: Select Case <выражение>
           [Case <списокВыражений-n>
            [инструкции-n]] ...
           [Case Else
            [инструкции_else]]
           End Select
```

Синтаксис инструкции Select Case содержит следующие элементы.

<выражение> – обязательный. Любое числовое выражение или строковое выражение.

<списокВыражений-n> – обязательный при наличии предложения Case. Список с разделителями, состоящий из одной или нескольких форм следующего вида:

- выражение;
- выражение To выражение;
- Is оператор Сравнения выражения.

Ключевое слово To задает диапазон значений. При использовании ключевого слова To перед ним должно находиться меньшее значение. Ключевое слово Is с операторами сравнения (кроме Is и Like) задает диапазон значений. Если ключевое слово Is не указано, оно вставляется по умолчанию.

<инструкции-n> – необязательный. Одна или несколько инструкций, выполняемых в том случае, если выражение совпадает с любым компонентом списка <списокВыражений-n>.

<инструкции_else > – необязательный. Одна или несколько инструкций, выполняемых в том случае, если выражение не совпадает ни с одним из предложений Case.

Если выражение совпадает с любым выражением из списка Выражений в предложении Case, выполняются все инструкции, следующие за данным предложением Case до следующего предложения Case, или, для последнего предложения, до инструкции End Select. Затем управление передается инструкции, следующей за End Select. Если выражение совпадает с выражениями из списка в нескольких предложениях Case, выполняется только первый подходящий набор инструкций.

В каждом предложении Case допускается использование нескольких выражений или диапазонов. Например, допустима следующая строка:

Case 1 To 4, 7 To 9, 11, 13, Is > MaxNumber

Примечание – Следует отличать оператор сравнения Is от ключевого слова Is, используемого в инструкции Select Case.

Подробную информацию и задания возьмите у преподавателя.

Контрольные вопросы

1 Каков принцип работы инструкции Select Case? Особенности применения.

2 Когда и как рекомендуется использовать инструкцию WithEnd With?

14 Лабораторная работа № 14. VBA. Цикл с параметром (For...Next)

Цель работы: ознакомление с теоретическим материалом. Выполнение заданий по обработке одномерных и двумерных массивов.

Цикл For.

Оператор цикла For позволяет повторять группу операторов заданное число раз. Синтаксис:

Forсчётчик_цикла =началоТоконец [step шаг]

Тело цикла

Next[счётчик_цикла]

Здесь счётчик_цикла – это числовая переменная. В начале выполнения цикла она принимает значение, задаваемое числовым выражением начало.

Числовое выражение конец задает заключительное выражение счётчика цикла. Числовое выражение шаг не обязательно и по умолчанию =1.

Тело цикла – это последовательность операторов, которая будет выполнена заданное число раз. Если шаг положителен, цикл завершится, когда впервые выполнится условие

счётчик_цикла>конец

Если шаг цикла отрицателен, условие его завершения

счётчик_цикла<конец

Это условие проверяется перед началом выполнения цикла, а затем – после каждого прибавления шага к счётчику цикла в операторе Next. Если оно выполнено, управление передается на оператор, следующий за Next, нет – выполняются операторы из тела цикла. Завершить цикл For...Next можно и с помощью оператора ExitFor. Такие операторы могут быть расположены в тех местах тела цикла, где требуется из него выйти, не дожидаясь выполнения условия завершения.

Массивы.

1 Описание массивов.

Dim<имя массива>(<начальное значение индекса>To<конечное значение индекса>) As<тип элементов массива>

Пр: DimA(1 To 10) AsInteger – массив состоит из 10 элементов, тип каждого элемента – Integer.

2 Обращение к элементу массива.

Обращение к элементу массива осуществляется следующим образом: указывается имя массива, а затем в круглых скобках – номер элемента в массиве.

Пр: A(1)=5 – первому элементу массива A присваивается значение 5.

A(17)=A(1) – 17-му элементу массива A присваивается значение первого элемента массива A.

3 Ввод массивов.

Массивы можно вводить как с листа Excel, так и используя встроенную функцию InputBox.

Пр: вводятся массивы A,B,C из 15 элементов:

Fori=1 To 15

' Ввод массива A с листа Excel, используя свойство Range (элементы вводятся из столбца A, строки изменяются с 1 по 15).

A(i)=Worksheets(1).Range("A" & i).Value

' Ввод массива B с листа Excel, используя свойство Cells (строки изменяются с 1 по 15, столбец 2 (столбец B))

B(i)=Worksheets(1).Cells(i,2)

' Ввод массива C через функциюInputBox

C(i)=InputBox("Введите " &i& "ый элемент массива")

Nexti

4 Решение задачи из примера через массивы:

1) заполнить диапазон ячеек A1:A5 значениями элементов массива;

2) текст программы

Sub mm ()

Dim A(1 to 5) As Integer

N=5

Y=0

For i=1 to n

A(i)=Worksheets(1).Cells(i,1)

Next i

For i=1 To n

Y=Y+log(A(i))/2^i

Next i

Worksheets(1).Range ("A6").Value = "результат"

Worksheets(1).Range ("A7").Value = Y

EndSub

Дополнительная информация находится у преподавателя.

Контрольные вопросы/задания

- 1 Найти количество положительных чисел.
- 2 Найти количество отрицательных чисел.
- 3 Найти сумму положительных чисел.
- 4 Найти сумму отрицательных чисел.
- 5 Найти количество элементов массива, больших или равных заданному значению.
- 6 Найти произведение положительных элементов массива.
- 7 Найти сумму элементов, стоящих на четных местах.
- 8 Найти сумму элементов, стоящих на нечетных местах.

15 Лабораторная работа № 15. VBA. Циклы с предусловием и постусловием

Цель работы: изучение циклов с предусловием и постусловием.

Циклы с предусловием;

а) Do While P
S
[Exit Do]
Loop

б) Do While P
S
Wend

Тело цикла S будет выполняться в том случае, когда условие P имеет значение

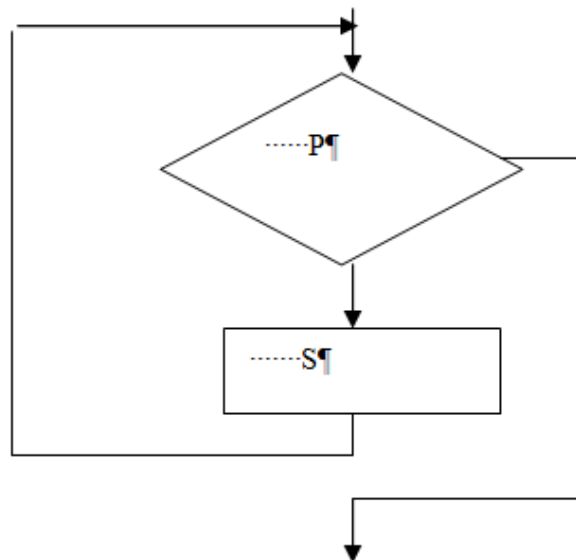


Рисунок 15.1 – Блок-схема

истина (цикл продолжается при истинном значении условия P). Если условие P ложно, то выполняются операторы, стоящие после цикла. В первом случае есть возможность досрочного выхода из цикла (это реализовано через Exit Do).

Пример – Необходимо организовать ввод чисел, пока их сумма не превысит значение заданного числа m .

Составим алгоритм задачи, используя различные циклы с предусловием и постусловием: вводите числа,

Процедура для решения задачи, с использованием цикла Do While ...Loop представлена на рисунке 15.2

```
Public Sub Sum_10
    Dim x As Integer
    Dim m As Integer
    Dim s As Integer
    Dim i As Integer
    m = InputBox("Введите число m")
    MsgBox ("Вводите суммируемые числа ")
    i=0
    s=0
    Do While s <= m
        i = i + 1
        x = InputBox("Введите " & i & "число")
        s = s + x
    Loop
    MsgBox ("Количество введенных чисел " & i)
End Sub
```

Рисунок 15.2 – Код программы

Кроме уже вышеописанных выше циклов с предусловием, существует еще один – это цикл Do Until ... Loop. Тело S этого цикла выполняется до тех пор, пока условие P не примет значение истина (цикл продолжается при ложном значении условия P). Есть возможность досрочного выхода из цикла. Его синтаксис следующий:

```
Do Until P
    S
    [Exit Do]
Loop
```

Далее смотрите код программы той же задачи, но с использованием цикла Do Until ... Loop.

Подробную информацию и задания к выполнению возьмите у преподавателя.

Контрольные вопросы

- 1 Что можно реализовать через Exit Do?
- 2 Для чего можно использовать цикл Do While ...Loop?
- 3 Для чего можно использовать цикл Do Until ... Loop?

16 Лабораторная работа № 16. VBA. Пользовательские функции

Цель работы: изучение пользовательских функций.

Приведем некоторые общие сведения об организации программ в VBA. Программа VBA состоит из одного или нескольких модулей.

Модуль – это лист с текстом программы, вставленный в рабочую книгу. Обычно текст программы VBA начинается с опций, которые управляют описанием переменных, способом сравнения строк и т. д. Затем следует объявление глобальных переменных или констант для данного модуля, т. е. таких переменных, которые используются во всех процедурах модуля. Далее располагается непосредственно текст функций пользователя и процедур, составляющих саму программу.

При написании программ удобно внутрь текста помещать комментарии. Комментарии – это пояснительный текст, который можно записать в любом месте программ. Каждая строка комментариев начинается со знака апострофа. Комментарии игнорируются компилятором и поэтому никакого влияния на программу не оказывают. Комментарии удобно использовать также при отладке операторов для их временного отключения.

Правильно организованной программой является такая программа, которая достигает результата наименьшим количеством итераций и при этом использует оптимальный вариант решения поставленной задачи. Это достигается многими компонентами, одним из которых и является функция.

Возьмем к примеру ситуацию, когда при проектировании программы Вы замечаете, что одну и ту же многошаговую операцию необходимо будет произвести с различными переменными. Разумным шагом в сложившейся ситуации является следующий – один раз написать код многошаговой операции и использовать его во всех необходимых местах данной программы, а в последствии и при написании других программ.

Процедура Function представляет собой набор команд, с помощью которого можно решить определенную задачу. Отличие процедуры Function от процедуры Sub состоит в том, что процедуры данного типа обязательно возвращают значение. При создании процедуры Function можно описать тип данных, который возвращает функция.

Синтаксис описания функции выглядит следующим образом:

Function имя ([аргументы]) [As тип]

 Инструкции

 имя=выражение

End Function

1 Ключевые слова Function и End Function используются в описании любой функции.

2 Параметр имя определяет имя функции.

3 Необязательные параметры аргументы позволяют передать в функцию требуемые значения.

4 Параметр As тип позволяет явно задать тип данных, который возвращает функция. Если он опущен, то по умолчанию возвращается значение типа Variant.

5 Внутри функции обычно имеется инструкция имя=выражение, которая используется для задания возвращаемого значения. Обратите внимание, что имя функции стоит слева от знака равенства, а параметр выражение определяет значение, которое должно быть совместимым с объявленным в параметре As тип типом данных для функции.

Подробную информацию возьмите у преподавателя.

Контрольные вопросы

- 1 Что делает функция Length_C?
- 2 Что представляет собой процедура Function?
- 3 Function и End Function используются в описании?
- 4 Для чего можно использовать окно Immediate?

17 Лабораторная работа № 17. VBA. Пользовательские процедуры

Цель работы: изучение пользовательских процедур.

Процедура – это набор описаний и инструкций, сгруппированных для вычисления. Существует три типа процедур: Sub, Function, Property. Имя процедуры всегда определяется на уровне модуля. В процедурах должны содержаться все исполняемые программы. Вложенность процедуры в другие процедуры не допускается. Для того, чтобы вставить процедуру необходимо вызвать редактор VBA и в меню редактора Вставка выбрать подменю Модуль. Затем снова выбрать Вставка и, в открывшемся подменю выбрать Процедура. Появляется диалоговое окно Вставка процедуры следующего вида (рисунок 17.1).

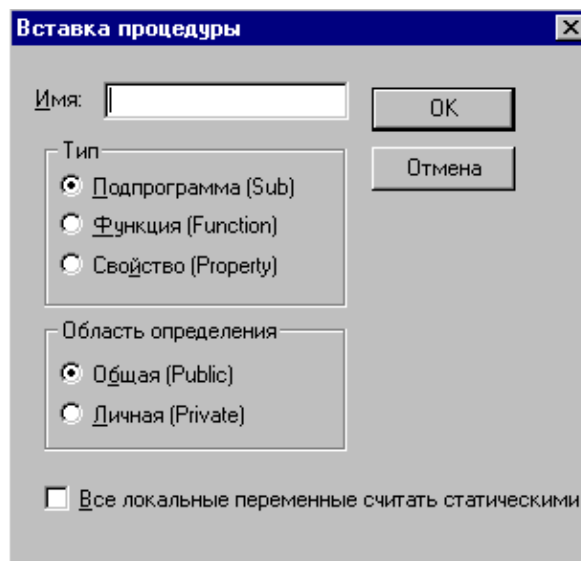


Рисунок 17.1 – Вставка процедуры

В окне Имя пишется название процедуры. В рамке Тип выбирается необходимая процедура, а в рамке Область определения – необходимая область определения для Вашей процедуры.

Инструкция Sub описывает имя, аргументы и текст программы, составляющий тело процедуры Sub.

Синтаксис:

```
[Private | Public] [Static] Sub имя [(списокАргументов)]
    [инструкции]
[Exit Sub]
    [инструкции]
End Sub
```

Синтаксис инструкции Sub содержит следующие элементы.

Элемент Описание.

Public Необязательный. Указывает, что процедура Sub доступна для всех других процедур во всех модулях. При использовании в личном модуле (модуле, который содержит инструкцию Option Private) такая процедура является недоступной вне проекта.

Private Необязательный. Указывает, что процедура Sub доступна для других процедур только того модуля, в котором она описана.

Static Необязательный. Указывает, что локальные переменные процедуры Sub сохраняются в промежутках времени между вызовами этой процедуры. Атрибут Static не действует на переменные, описанные вне процедуры Sub, даже если они используются в процедуре.

Имя Обязательный. Имя процедуры Sub, удовлетворяющее стандартным правилам именованию переменных.

Список Аргументов Необязательный. Список переменных, представляющий аргументы, которые передаются в процедуру Sub при ее вызове. Имена переменных разделяются запятой.

Инструкции Необязательный. Любая группа инструкций, выполняемых в процедуре Sub.

Дополнительную информацию и задания возьмите у преподавателя.

Контрольные вопросы

1. На что указывает элемент Public?
2. Что такое инструкция Option Private? Для чего она служит?
3. Для чего служит элемент Static?
4. Допускаются ли рекурсивные процедуры Sub?
5. Ключевое слово Static используется ли с рекурсивными процедурами Sub?

18 Лабораторная работа № 18. Графический интерфейс пользователя. Простейшие вычисления Matlab

Цель работы: изучение основных положений по рассматриваемой тематике, используя приведенные в работе сведения и справочную систему Matlab. Выполнение практических заданий.

Задание.

Выполнение:

- запустите Matlab;
- используя команду меню, View установите режим отображения только командного окна;
- используя средства ОС опробуйте способы изменения размеров окна;
- установите текущей рабочей папкой;
- введите в командном окне рисунок – 18.1:

→ $x=2$; и далее ↵ (символ нажатия клавиши ввод) ¶

→ $y=3$; ↵ ↵ ¶

→ $z=(x+y+5)/2-x^2$ ↵ ¶

Рисунок – 18.1 Уравнения

Проанализируйте результаты, отображаемые в командном окне после ввода.

В командном окне показываются вводимые с клавиатуры выражения и результаты вычисления. После многократного повторения вычислений, когда строки сдвигаются вверх так, что верхняя строка покидает область видимости, а в самом низу окна появляется свободная строка для ввода новых данных. Эта строка содержит знак приглашения >>.

Информация, покинувшая видимую часть окна, не исчезает. Ее можно увидеть снова, если осуществить вертикальную протяжку содержимого командного окна, используя полосу протяжки, или с помощью следующих клавиш клавиатуры: PageUp, PageDown, Ctrl+Home и Ctrl+End.

Клавиши «Стрелка вверх» и «Стрелка вниз» в любом текстовом редакторе осуществляют перемещение курсора вверх-вниз и вертикальную протяжку содержимого окна, в Matlab работают иначе. Эти клавиши позволяют вернуть в строку ввода, ранее введенную в командное окно входную информацию, как правило, эта возможность используется для редактирования.

В итоге можно сказать, что видимая информация в окне системы Matlab располагается в двух разных зонах: зоне просмотра и зоне редактирования, как на рисунке 18.2.

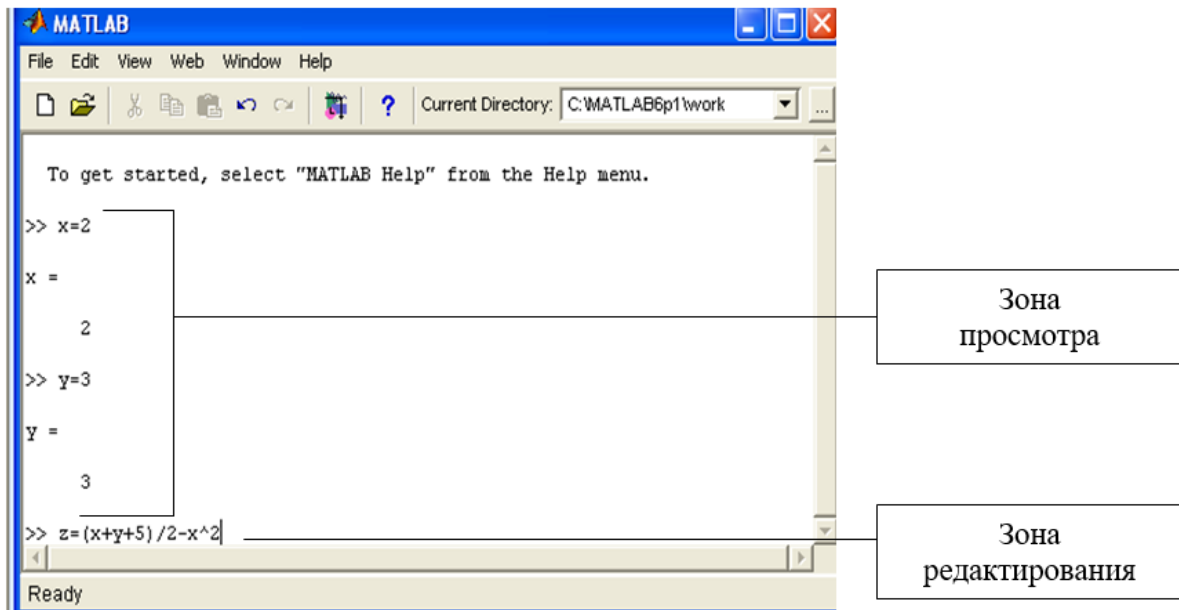


Рисунок 18.2 – Окно Matlab

В зоне просмотра ничего нельзя исправить, хотя в нее можно поместить курсор, однако реакцией на ввод с клавиатуры будет перемещение курсора в строку ввода, расположенную в зоне редактирования. В зоне просмотра можно выделять с помощью мыши любую информацию и помещать ее в буфер обмена, чтобы потом ее вставить в документ текстового редактора, либо в строку ввода.

Дополнительные сведения находятся у преподавателя вместе с заданиями.

Контрольные вопросы

- 1 Как настроить режим отображения окна Matlab?
- 2 В каких зонах располагается в окне введенная информация? Можно ли разбивать выражение при вводе на несколько строк?
- 3 Как отредактировать ранее введенную в командном окне информацию?
- 4 Где сохраняются переменные текущего сеанса работы? Типы файлов, в которых сохраняются эти переменные. Способы создания файлов.
- 5 Способы загрузки сохраненных переменных в память компьютера.
- 6 Типы числовых данных, используемых в Matlab. Форматы их отображения в командном окне. Удаление переменных из памяти компьютера.
- 7 Варианты использования точки с запятой.
- 8 Назначение команд `who` и `whos`.
- 9 Способы формирования векторов и матриц.
- 10 Функции определения размеров массивов и их структуры.
- 11 Какой вид имеет приглашение, выводимое при загрузке Matlab?
- 12 Для чего служит команда `HELP`?
- 13 Для чего служит команда `DEMO`?
- 14 Для чего служит команда `INFO`?

15. Перечислите основные команды Matlab для работы в режиме прямых вычислений.
16. С помощью какой команды устанавливается формат чисел?
17. Для чего служит команда `format hex`?
18. Перечислите основные системные переменные Matlab.
19. Приведите примеры математических функций системы Matlab.

19 Лабораторная работа № 19. Операции с векторами и матрицами в системе Matlab

Цель работы: изучение реализации средствами системы Matlab основных операций с векторами и матрицами. После выполнения практических заданий переменные рабочей области сохраните в `mat`-файле в рабочей папке.

Возможно задание векторов и матриц с комплексными элементами.

Матрицы можно расширять, используя матрицы малых размеров как элементы матриц больших размеров.

Генерацию некоторых наиболее распространенных видов матриц обеспечивают следующие матричные функции:

- `zeros(m,n)` – генерация матрицы с нулевыми элементами;
 - `ones(m,n)` – генерация матрицы с единичными элементами;
 - `rand(m,n)` – генерация матрицы с элементами, имеющими случайные значения;
 - `eye(m,n)` – генерация матрицы с единичными диагональными элементами (m – количество строк; n – количество столбцов матрицы).
- `a(i,j)` – выделение элемента i -й строки j -го столбца;
`a(i,:)` – выделение i -й строки;
`a(:,j)` – выделение j -го столбца.

В Matlab возможны следующие операции с векторами и матрицами:

- + – сложение;
- – вычитание;
- * – умножение;
- \ / – деление;
- ' – транспонирование;
- ^ – возведение в степень;
- `inv(m)` – обращение матрицы;
- `pinv(m)` – псевдообращение матрицы;
- `sqrtm(m)` – матричный квадратный корень;
- `poly(m)` – вектор с коэффициентами характеристического многочлена матрицы;
- `det(m)` – значение определителя матрицы;
- `trace(m)` – след матрицы;
- `rank(m)` – ранг матрицы.

Система Matlab имеет ряд функций, предназначенных для обработки данных, заданных в матричной или векторной форме.

Функция `size(m)` служит для определения числа строк и столбцов матрицы `m`. Она возвращает вектор `[n, p]`, содержащий эти данные.

Функция `max(v)` возвращает значение максимального по значению элемента вектора `v`. Если ее аргументом является матрица, например `max(m)`, то функция возвращает вектор-строку, содержащий значения максимальных элементов каждого из столбцов.

Аналогично действует функция `min(m)`, выделяющая элементы с минимальными значениями. Функция `mean(v)` возвращает среднее значение элементов вектора `v`, а функция `mean(m)` с матричным аргументом возвращает вектор-строку средних значений каждого из столбцов данных. Функция `std(v)` возвращает статистический параметр – стандартное (квадратичное) отклонение для одномерного массива данных, представленного вектором `v`. Для матричного аргумента эта функция возвращает вектор-строку стандартных отклонений для каждого из столбцов.

Функция сортировки `sort(v)` возвращает вектор, элементы которого расположены в порядке роста их значений. Для матричного аргумента эта функция возвращает матрицу, у которой отсортированы элементы каждого столбца.

Функция `sum(v)` возвращает сумму элементов вектора `v`, а для матричного аргумента функция `sum(m)` возвращает вектор-строку сумм элементов по каждому из столбцов.

Аналогично функция `prod(m)` возвращает вектор произведений элементов каждого из столбцов.

Порядок выполнения лабораторной работы.

1 Ввод с клавиатуры векторов и матриц.

Ввести:

- вектор-строку (`v`);
- вектор-столбец (`w`);
- матрицу (`m`).

2 Генерация матриц специального вида.

Создать:

- матрицу с нулевыми элементами (`m0`);
- матрицу с единичными элементами (`m1`);
- матрицу с элементами, имеющими случайные значения (`mr`);
- матрицу с единичными диагональными элементами (`me`).

3 Вычисление матрицы `mm` по формуле, представленной в табл. 1 (см. Приложение).

4 Изучение функций обработки данных.

Для матрицы `mm` определить и сохранить значения в переменных (имена выбрать самостоятельно):

- число строк и столбцов матрицы;
- значение максимального элемента матрицы;
- значение минимального элемента матрицы;

- значение суммы элементов матрицы;
- значение произведения элементов матрицы.

5. Изучение функции сортировки.

Расположить с помощью функции сортировки `sort` элементы вектора v в порядке роста их значений. Сохранить вектор после сортировки v_s .

Подробную и дополнительную информацию получите у преподавателя.

Контрольные вопросы

- 1 Как осуществляется ввод вектора-строки?
- 2 Как осуществляется ввод вектора-столбца?
- 3 Как осуществляется ввод матрицы?
- 4 Для чего служит команда `zeros`?
- 5 Для чего служит команда `ones`?
- 6 Для чего служит команда `rand`?
- 7 Для чего служит команда `eye`?
- 8 Как определяется число строк и столбцов матрицы?
- 9 Какие операции служат для определения минимального и максимального элемента матрицы?
- 10 С помощью каких операций осуществляется суммирование и умножение элементов матриц?

20 Лабораторная работа № 20. Построение графиков в Matlab

Цель работы: приобретение навыков построения графиков.

Задание 1

Задана функция $y(x)=e^{-x} \sin 10x$ на отрезке $[0, 1]$ с шагом 0.1.

Постройте график функции, Для этого введите в командной строке `>> plot(x,y)`.

После выполнения команды на экране появится окно Figure No. 1 с графиком функции. Команда `plot` соединяет точки с координатами $(x(i), y(i))$ прямыми линиями, автоматически масштабируя оси для оптимального расположения графика в окне. Построенный график не должен иметь изломов, так как сама функция гладкая. Для более точного представления функции нужно задать большее количество точек на отрезке $[0,1]$.

Задание 2

При помощи клавиш \uparrow \downarrow найдите ранее введенную команду задания вектора x , задайте более мелкий шаг (0.01), найдите и выполните команду вычисления функции и заново запустите построение графика. У Вас должен получиться гладкий график функций.

Таким образом, подбирая шаг изменения аргумента, можно добиться точного представления функции на графике.

Графическая функция `fplot` строит график функции без задания пользователем фиксированного шага изменения аргумента.

Функция задается в символьном виде.

Синтаксис функции:

`fplot ('f(x)', [xmin xmax])`

Строит график функции $f(x)$ в интервале изменения аргумента x от x_{\min} до x_{\max} .

Выполните:

- постройте функции $y(x)=e^{-x} \sin 10x$, используя `fplot`;
- дополните график сеткой;
- включите и выключите сетку.

В математической, физической и иной литературе при построении графиков в дополнение к разметке осей часто используют масштабную сетку. Команды `grid` позволяют задавать построение сетки или отменять это построение:

- `grid on` – добавляет сетку к текущему графику;
- `grid off` – отключает сетку;
- `grid` – последовательно производит включение и отключение сетки.

Дополнительную информацию и задания возьмите у преподавателя.

Контрольные вопросы

1. Графическая функция `fplot`. Для чего она нужна?
2. Расшифруйте запись `plot(x,y)`.
3. Как осуществляется построение сферы по окружностям?

Список литературы

1. **Гуриков, С. Р.** Информатика: учебник / С. Р. Гуриков. – 2-е изд., перераб. и доп. – Москва: ИНФРА-М; ФОРУМ, 2020. – 630 с.
2. **Гвоздева, В. А.** Информатика, автоматизированные информационные технологии и системы: учебник / В. А. Гвоздева. – Москва: ФОРУМ; ИНФРА-М, 2021. – 542 с.
3. **Рудикова, Л. В.** Microsoft Office Excel 2019: практическое пособие / Л. В. Рудикова. – Санкт-Петербург: БХВ-Петербург, 2020. – 624 с.