



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ДГТУ)**

Колледж экономики, управления и права

**Методические указания по организации  
практической работы студентов  
по учебной дисциплине  
Информатика**

09.02.05 Прикладная информатика (по отраслям).

Ростов-на-Дону

2018

Методические указания по учебной дисциплине «Информатика» разработанные учетом ФГОС среднего профессионального образования специальности социально-экономического профиля: Прикладная информатика (по отраслям) предназначены для обучающихся и преподавателей колледжа.


Методические указания определяют этапы выполнения работы на практическом занятии, содержат рекомендации разработаны на основе требований ФГОС среднего общего образования, предъявляемых к структуре, содержанию и результатам освоения учебной дисциплины «Информатика».

Составитель (автор): А.С Пегливанова преподаватель колледжа ЭУП

С.В. Шигаева преподаватель колледжа ЭУП

Рассмотрены на заседании предметной (цикловой) комиссии специальности  
общеобразовательных дисциплин

Протокол № 1 от «31» августа 2018 г

Председатель предметной (цикловой) комиссии  Н.В. Река  
личная подпись

и одобрены решением учебно-методического совета колледжа.

Протокол № 1 от «31» августа 2018 г

Председатель учебно-методического совета колледжа

комиссии  С.В.Шинаикова  
личная подпись

Рекомендованы к практическому применению в образовательном процессе

## Содержание

Практическая работа № 1-2 .....	5
Тема: «Решение задач на определение количества информации».....	5
Практическая работа №3 .....	9
Тема «Запись и перевод чисел в различные системы счисления».....	9
Практическая работа №4 .....	14
Тема: «Самостоятельное решение задач по разделу» .....	14
Практическая работа №5 .....	15
Тема: «Составление таблиц истинности логических функций и выражений».....	15
Практическая работа №6 .....	19
Тема: «Самостоятельное решение задач» .....	19
Практическое занятие № 7-8.....	20
Тема: «Программное обеспечение внешних устройств. Подключение внешних устройств к компьютеру и их настройка». ....	20
Практическая работа № 9 .....	22
Тема: «Хранение информационных объектов различных видов на различных цифровых носителях. Определение объемов различных носителей информации». ....	22
Практическая работа № 10 .....	25
Тема: «Создание архива данных. Извлечение данных из архива.» .....	25
Практическая работа № 11 .....	<b>Error! Bookmark not defined.</b>
Тема: «Запись информации на внешние носители различных видов» ...	<b>Error! Bookmark not defined.</b>
Практическая работа № 12-13.....	28
Тема: «Операционная система. Графический интерфейс пользователя».....	28
Практическая работа №14 .....	35
Тема: «Файл как единица хранения информации на компьютере. Атрибуты файла и его объем. Учет объемов файлов при их хранении, передаче .....	35
Практическое занятие № 15 .....	38
Тема: «Современные антивирусные программы». ....	38
Практическая работа №16. ....	43
Тема: «Защита информации, антивирусная защита» Эксплуатационные требования к рабочему месту» .....	43
Практическая работа № 17 .....	44
Тема: Информационная деятельность человека. ....	44
Практическая работа № 18 .....	47
Тема: «Построение блок-схемы алгоритма». ....	91
Практическая работа № 19 .....	<b>Error! Bookmark not defined.</b>

Тема: «Составление и отладка линейных и разветвляющихся программ.»	<b>Error! Bookmark not defined.</b>
Практическая работа № 20	<b>Error! Bookmark not defined.</b>
Тема: «Составление и отладка циклических программ»	<b>Error! Bookmark not defined.</b>
Практическая работа № 21	97
Тема: «Составление программ с одномерными и двумерными массивами»	97
Практическая работа № 22	102
Тема:« Самостоятельное составление программ по заданным критериям»	102
Практическая работа № 23	108
Тема:« Форматирование электронного документа»	108
Практическая работа № 24	108
Тема:« Внедрение объектов в электронный документ»	<b>Error! Bookmark not defined.</b>
Практическая работа № 25	<b>Error! Bookmark not defined.</b>
Тема:« Создание и форматирование таблиц в Excel»	<b>Error! Bookmark not defined.</b>
Практическая работа № 26	120
Тема: «Построение диаграмм»	129
Лабораторная работа № 27	137
Тема: «Создание файла баз данных»	120
Лабораторная работа № 28	<b>Error! Bookmark not defined.</b>
Тема: «Создание запросов к базе данных	<b>Error! Bookmark not defined.</b>
Практическая работа № 29	123
Тема: «Создание растровых и векторных изображений и чертежей в графических редакторах	123
Практическая работа №30	<b>Error! Bookmark not defined.</b>
Тема: «Создание мультимедийных презентаций (по профилю специальности).»	<b>Error! Bookmark not defined.</b>

## Практическая работа № 1-2

**Тема: «Решение задач на определение количества информации, на кодирование текстовой, графической и звуковой информации»**

**Цель:** изучить основные подходы к понятиям информация и измерение информации.

### Теоретический материал

#### 1. Содержательный подход

Для равновероятных событий используется формула Хартли  $I = \log_2 N$ , где  $N$  – количество возможных равновероятных событий,  $I$  – количество информации.

**Пример 1.** В барабане для розыгрыша лотереи находится 32 шара. Сколько информации содержит сообщение о первом выпавшем номере (например, выпал номер 15)?

**Решение:**

т.к. вытаскивание любого из шаров равновероятно, то количество информации вычисляется по формуле  $2^I = N$ , где  $I$  – количество информации, а  $N$  – количество шаров. Тогда  $2^I = 32$ , отсюда  $I = 5$  бит.

**Ответ:** 5 бит.

**Пример 2.** Какой объем информации содержит сообщение, уменьшающее неопределенность знаний в 4 раза?

**Решение:**

1 бит – это количество информации, уменьшающее неопределенность знаний в 2 раза.

1 бит – это количество информации, уменьшающее неопределенность знаний в  $2^I$  раз, отсюда,  $4 = 2^I$ ,  $I = 2$  бит.

**Ответ:** 2 бит.

**Пример 3.** При угадывании целого числа в некотором диапазоне было получено 8 бит информации. Сколько чисел содержал этот диапазон?

**Решение:**

$i = 8$  бит

$N = 28$

$N = 256$

**Ответ:** 256 чисел в диапазоне.

**Для реализации Примера 4 – Примера 5 необходимо использовать Калькулятор Windows.**

**Пример 4.**

Какое количество информации несет сообщение: «Встреча назначена на сентябрь».

**Решение:**

Поскольку появление в сообщении месяца сентябрь равновероятно из 12 месяцев, то количество информации определяется по формуле:  $2^I = N$ , где  $I$  – количество информации,  $N$  – количество месяцев. Отсюда:  $2^I = 12$ ,  $I = \log_2 12 \approx 3.584962501$  бит.

**Пример 5.**

Какое количество информации несет сообщение о том, что встреча назначена на 23 октября в 15.00?

**Решение:**

Поскольку появление в сообщении определенного числа месяца, определенного месяца и определенного часа равновероятно из общего числа дней в месяце, общего числа месяцев, общего числа часов, то количество информации определяется по формуле:  $2^I = N$ , где  $I$  – количество информации,  $N = 31 * 12 * 24$  – (количество дней в месяце) \* (количество месяцев) \* (количество часов в сутках). Отсюда:  $2^I = 31 * 12 * 24$ ,  $I = \log_2 (31 * 12 * 24) \approx 13.12412131$  бит.

## 2. Вероятностный подход

### Формула Шеннона

Так как наступление каждого из  $N$  событий имеет одинаковую вероятность  $P$ , то  $P=1/N$ . Если событий 6, то вероятность появления одного события равно  $1/6$ , если событий 100, то вероятность равна  $0,01 \Rightarrow N=1/P$ .

Американский ученый Клод Шеннон, что формулу Хартли можно записать иначе:  $I = \log_2(1/P) = \log_2 P^{-1} = -\log_2 P$ , так как  $p < 1$ , то  $\log_2 P < 0$ , а  $I > 0$  (минус на минус дает плюс)

В 1948 год он предложил другую формулу определения количества информации, учитывая возможную неодинаковую вероятность событий в наборе.

Для событий с различными вероятностями количество информации определяется по формуле Шеннона:  $I = -\sum_{i=1}^N P_i \log_2 P_i$ , где  $I$  – количество информации,  $N$  – количество возможных не равновероятных событий,  $P_i$  – вероятности отдельных событий.

При решении задач необходимо подчеркнуть, что при равновероятных событиях мы получаем большее количество информации, чем при неравно вероятных событиях.

#### Пример 6.

В корзине лежат 8 черных шаров и 24 белых. Сколько информации несет сообщение о том, что достали черный шар?

#### Решение:

Вероятность достать черный шар равна  $p_ч = 8/(24+8) = 0.25$

Количество информации в сообщении о том, что достали черный шар, вычисляется по формуле:  $I_ч = \log_2 1/p_ч$ . Отсюда  $I = \log_2(1/0.25) = 2$  бита.

## 3. Алфавитный подход к измерению информации

$N = 2^i$ $i = \log_2 N$ $I = K * i$	$N$ - полное количество символов в алфавите $i$ - количество информации, которое несет каждый символ $K$ - размер текста $I$ - размер информации, содержащейся в тексте
--	--

**Пример 7.** Алфавит племени Мульти состоит из 8 букв. Какое количество информации несет одна буква этого алфавита?

#### Решение:

Количество информации, которое несет каждый символ алфавита мощностью  $N$  равно:  $I = \log_2 N$ ,  $N = 2^i$ ,  $8 = 2^3$ ,  $I = 3$  (бит)

**Пример 8.** Информационное сообщение объемом 1,5 Кбайта содержит 3072 символа. Сколько символов содержит алфавит, при помощи которого было записано это сообщение?

#### Решение:

Объем сообщения в битах:  $1,5 * 1024 * 8 = 12288$  (бит)

Количество информации которое несет каждый символ в алфавите:  $12288/3072 = 4$  (бит)

Мощность алфавита:  $N = 2^i$ ,  $N = 2^4 = 16$  (символов)

### Задание для практической работы

#### Вариант 1

1. В рулетке общее количество лунок равно 128. Какое количество информации мы получаем в зрительном сообщении об остановке шарика в одной из лунок?

2. При угадывании целого числа в некотором диапазоне было получено 6 бит информации. Сколько чисел содержит этот диапазон?

3. В коробке лежат 7 разноцветных карандашей. Какое количество информации содержит сообщение, что из коробки достали красный карандаш?

4. Какое количество вопросов достаточно задать вашему собеседнику, чтобы точно определить день и месяц его рождения?

5. В коробке лежат 64 цветных карандаша. Сообщение о том, что достали белый карандаш, несет 4 бита информации. Сколько белых карандашей было в корзине?

6. В непрозрачном мешочке хранятся 10 белых, 20 красных, 30 синих и 40 зеленых шариков. Какое количество информации будет содержать зрительное сообщение о цвете вынутого шарика?

7. Для записи текста использовался 256-символьный алфавит. Каждая страница содержит 30 строк по 70 символов в строке. Какой объем информации содержат 5 страниц текста?

### **Вариант 2**

1. Группа школьников пришла в бассейн, в котором 4 дорожки для плавания. Тренер сообщил, что группа будет плавать на дорожке номер 3. Сколько информации получили школьники из этого сообщения?

2. При угадывании целого числа в диапазоне от 1 до  $N$  было получено 7 бит информации. Чему равно  $N$ ?

3. При игре в кости используется кубик с шестью гранями. Сколько бит информации получает игрок при каждом бросании кубика?

4. Какое количество вопросов достаточно задать вашему собеседнику, чтобы точно определить день и месяц его рождения?

5. В коробке имеется 50 шаров. Из них 40 белых и 10 черных. Сколько информации в сообщении о том, что вытащили белый (черный) шар?

6. В классе 30 человек. За контрольную работу по математике получено 6 пятерок, 15 четверок, 8 троек и 1 двойка. Какое количество информации в сообщении о том, что Иванов получил четверку?

7. Сообщение занимает 3 страницы по 25 строк. В каждой строке записано по 60 символов. Сколько символов в использованном алфавите, если все сообщение содержит 1125 байтов?

### **Вариант 3**

1. В корзине лежат 8 шаров. Все шары разного цвета. Сколько информации несет сообщение о том, что из корзины достали красный шар?

2. Сообщение о том, что ваш друг живет на 10 этаже, несет 4 бита информации. Сколько этажей в доме?

3. В коробке лежат 10 разноцветных карандашей. Какое количество информации содержит сообщение, что из коробки достали красный карандаш?

4. Какое количество вопросов достаточно задать вашему собеседнику, чтобы точно определить день и месяц его рождения?

5. Сережа — лучший ученик в классе. Сколько информации в сообщении о том, что за контрольную работу по математике Сережа получит "5", если за несколько лет учебы он получил по математике 100 оценок, из них: 60 пятерок, 30 четверок, 8 троек и 2 двойки.

6. Какое количество информации будет содержать зрительное сообщение о цвете вынутого шарика, если в непрозрачном мешочке хранятся: 30 белых, 30 красных, 30 синих и 10 зеленых шариков.

7. Для записи сообщения использовался 64-х символьный алфавит. Каждая страница содержит 30 строк. Все сообщение содержит 8775 байтов информации и занимает 6 страниц. Сколько символов в строке?

### **Вариант 4**

1. Была получена телеграмма: «Встречайте, вагон 7». Известно, что в составе поезда 16 вагонов. Какое количество информации было получено?

2. Сообщение о том, что Петя живет во втором подъезде, несет 3 бита информации. Сколько подъездов в доме?

3. В коробке лежат 15 разноцветных карандашей. Какое количество информации содержит сообщение, что из коробки достали красный карандаш?

4. Какое количество вопросов достаточно задать вашему собеседнику, чтобы точно определить день и месяц его рождения?

5. В пруду живут 8000 карасей, 2000 щук и 40 000 пескарей. Сколько информации в сообщении о том, что рыбак поймал в этом пруду пескаря (карася, щуку)?

6. Какое количество информации будет содержать зрительное сообщение о цвете вынутого шарика, если в непрозрачном мешочке хранятся: 25 белых, 25 красных, 25 синих и 25 зеленых шариков.

7. Для записи текста использовался 256-символьный алфавит. Каждая страница содержит 30 строк по 70 символов в строке. Какой объем информации содержат 5 страниц текста?

#### **Вопросы для самоконтроля**

1. Какие бывают походы к измерению количества информации?



## Практическая работа №3

### Тема «Запись и перевод чисел в различные системы счисления».

**Цель:** изучить основные алгоритмы перехода из одной системы счисления в другую, сформировать умения переводить числа из одной системы счисления в другую, работать с числами разных систем счисления – позиционных и непозиционных

#### Обеспечение практической работы:

- задания для выполнения работы;
- материал лекций

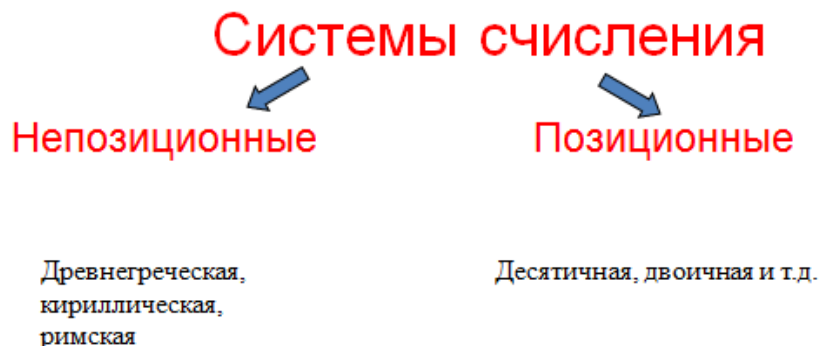
#### В результате выполнения данной работы студент должен уметь:

- выполнять простой перевод системы счисления из одной в другую;

#### Теоретический материал

**Система счисления** — совокупность правил наименования и изображения чисел с помощью набора символов, называемых цифрами.

Количество цифр (знаков), используемых для представления чисел называют *основанием системы счисления*.



### Римская система записи чисел

I	V	X	L	C	D	M
1	5	10	50	100	500	1000

**Например,** число **CCXXXII** складывается из двух сотен, трех десятков и двух единиц и равно **232**.

В римских числах цифры записываются слева направо в порядке убывания. В таком случае их значения складываются. Если слева записана меньшая цифра, а справа – большая, то их значения вычитаются.

#### Например:

$$VI = 5 + 1 = 6, IV = 5 - 1 = 4.$$

$$MCMXCVII = 1000 + (-100 + 1000) + (-10 + 100) + 5 + 1 + 1 = 1997$$

#### Позиционная система счисления

##### Развернутая форма записи числа

В числе 555 первая пятерка обозначает пять сотен, вторая – пять десятков, третья – пять единиц. Одна и та же цифра в зависимости от позиции в записи числа обозначает разные величины.

$$555 = 5 * 100 + 5 * 10 + 5$$

Свернутая форма записи  
числа

Развернутая форма записи  
числа

Всякое десятичное число можно представить как сумму произведений составляющих его цифр на соответствующие степени десятки. То же самое относится и к десятичным дробям.

$$10^0 = 1$$

$$10^1 = 10$$

$$10^2 = 100$$

$$10^3 = 1000$$

$$10^{-1} = 0,1$$

$$10^{-2} = 0,01$$

$$10^{-3} = 0,001 \text{ и т.д.}$$

*Например:*

$$26,387 = 2 * 10^1 + 6 * 10^0 + 3 * 10^{-1} + 8 * 10^{-2} + 7 * 10^{-3}$$

**Алгоритмы перевода чисел из одной системы счисления в другую:**

**1. Алгоритм перевода целых чисел из десятичной системы счисления в любую другую.**

1. Последовательно выполнять деление данного числа и получаемых целых частных на основание новой системы счисления до тех пор, пока не получится частное, меньше делителя.

2. Полученные остатки, являющиеся цифрами числа в новой системе счисления, привести в соответствие с алфавитом новой системы счисления.

3. Составить число в новой системе счисления, записывая его, начиная с последнего остатка.

**Пример 1.**

Перевести число  $97_{10}$  в двоичную систему счисления.

97	2					
1	48	2				
	0	24	2			
		0	12	2		
			0	6	2	
				0	3	2
					1	1

Получаем  $97_{10} = 1100001_2$

**Пример 2.**

Перевести число  $126_{10}$  в восьмеричную систему счисления.

126	8	
6	15	8
	7	1

Получаем  $126_{10} = 176_8$

**Пример 3.**

Перевести число  $180_{10}$  в шестнадцатеричную систему счисления.

180	16
4	11(B)

## 2. Алгоритм перевода правильных дробей из десятичной системы счисления в любую другую

1. Последовательно умножаем данное число и получаемые дробные части произведения на основание новой системы счисления до тех пор, пока

Дробная часть произведения не станет равной нулю или будет достигнута требуемая точность представления числа.

2. Полученные целые части произведений, являющиеся цифрами числа в новой системе счисления, привести в соответствие с алфавитом новой системы счисления.

3. Составить дробную часть числа в новой системе счисления, начиная с целой части первого произведения.

### Пример 4.

Перевести число  $0,65625_{10}$  в восьмеричную систему счисления.

0,	65625
	• 8
5	25000
	• 8
2	00000

Получаем:  $0,65625_{10} = 0,52_8$

### Пример 5.

Перевести число  $0,65625_{10}$  в шестнадцатеричную систему счисления.

0,	65625
	• 16
10	50000
(A)	• 16
8	00000

Получаем:  $0,65625_{10} = 0,A8_{16}$ .

### Пример 6.

Перевести число  $0,625_{10}$  в двоичную систему счисления.

0,	625
	• 2
0	1250
	• 2
0	2500
	• 2
0	5000
	• 2
1	0000

Получаем:  $0,625_{10} = 0,0001_2$

### Пример 7.

Перевести число  $0,9_{10}$  в двоичную систему счисления.

0,	9
	• 2
1	8
	• 2
1	6
	• 2
1	2
	• 2
0	4
	• 2
0	8
	• 2
1	6

Этот процесс можно продолжать бесконечно. В этом случае деление продолжаем до тех пор, пока не получим нужную точность представления.

Получаем:  $0,9_{10} = 0,111001_2$  с точностью до семи значащих цифр после запятой.

### 3. Перевод произвольных чисел

Перевод произвольных чисел, т.е. содержащих целую и дробную часть, осуществляется в два этапа. Отдельно переводится целая часть, отдельно  $\sim\sim$  дробная. В итоговой записи полученного числа целая часть отделяется от дробной запятой.

#### Пример 8.

Перевести число 18,34 из десятичной в двоичную систему счисления.

Переводим целую часть:

18	2		
9	2		
1	4	2	
	0		1

Переведем дробную часть.

0,	34
	* 2
0	68
	* 2
1	36
	* 2
0	72
	* 2
1	44

Получаем:  $18,34_{10} = 1010,0101_2$

#### Пример 9.

Перевести число 124,26 в шестнадцатеричную систему счисления.

Переводим целую часть:

124	16
12	7
(C)	

Переведем дробную часть

0,	26
	* 16
4	16
	* 16
2	56
	* 16
8	96
	* 16
15	36
(F)	

Получаем:  $124,26_{10} = 7C,428F_{16}$ .

### Задание для практической работы

#### Задание 1.

В некоторой системе счисления цифры имеют форму различных геометрических фигур. Приведены примеры некоторых чисел, записанных в этой системы счисления:

○ △	- 4	□ □ □	- 190
△ ○	- 6	▽ □ ▽	- 1900
□ ○ □	- 19		

Какому числу соответствует следующая запись:

▽ □ ▽ □ □ △ ○ ○ ?

**Задание 2.**

Выполните действия и запишите результат римскими цифрами:

**XXII – V; CV – LIII + XIX;**

**MCM + VIII; XX : V;**

**X \* IV; LXVI : XI;**

**XXIV \* VII.**

**Задание 3.**

Записать числа в развернутой форме:

**285                      3864                      893,735                      34,07**

**Задание 4.** Переведите число  $2004_{10}$  в:

- А) двоичную систему счисления;
- Б) восьмеричную систему счисления;
- В) шестнадцатеричную;

**Задание 5.** Переведите числа из десятичной системы счисления:

- А) 0, 141 в пятеричную систему счисления с точностью до пяти знаков.
- Б) 0, 675 в троичную систему счисления с точностью до пяти знаков.
- В) 0, 2004 в восьмеричную систему счисления с точностью до пяти знаков.
- Г) 0,7982 в двоичную систему счисления с точностью до шести знаков.

**Задание 6.** Перевести числа в десятичную систему счисления.

- a)  $1101_2$
- b)  $0,123_5$
- c)  $16,4_8$
- d)  $7688_9$
- e)  $432,1_5$
- f)  $120_3$
- g)  $102,31_4$

**Задание 7.** Представьте в десятичной системе счисления число  $101,1$ , считая записанным в системах счисления от двоичной до девятеричной.

**Задание 8.** В коробке лежит  $31_8$  шар. Среди них  $12_8$  красных и  $17_8$  желтых. Докажите, что здесь нет ошибки.

**Задание 9.**

В классе  $1111_2$  девочек и  $1010_2$  мальчиков. Сколько учеников в классе?

**Задание 10.**

Используя приложение Калькулятор операционной системы Windows, заполните таблицу.

Dec	Oct	Bin	Hex
1011			
	1011		
		1011	
			1011

**Вопросы для самоконтроля**

1. Что такое система счисления?
2. Чем отличаются позиционные системы счисления от непозиционных, в чем их преимущества?
3. Привести примеры позиционных и непозиционных систем счисления.
4. А. С. Пушкин родился в MDCCXCIX году?
5. Что такое основание системы счисления?
6. В какой стране впервые стали использоваться специальные обозначения для 5,10,100,1000,1000000?

## Практическая работа №4

Тема: «Самостоятельное решение задач по разделу»

Цель: закрепить теоретические знания и приобрести практические навыки:

**Обеспечение практической работы:**

- задания для выполнения работы;
- материал лекций

**Задания да самостоятельной работы:**

### Вариант 1.

1. Выпишите алфавит 5-ричной системы счисления.
2. Какое минимальное основание должна иметь система счисления, если в ней могут быть записаны числа:  
10, 21, 201, 1201?
3. Переведите:
  - a)  $3218 \rightarrow A_{10}$
  - b)  $101,112 \rightarrow A_{10}$
  - c)  $34510 \rightarrow A_5$
  - d)  $9810 \rightarrow A_2$
4. В какой системе счисления справедливо равенство:  $2 \times 2 = 10$ .
5. Какое число предшествует числу 108 в 8-ричной системе счисления.

### Вариант 2.

1. Выпишите алфавит 9-ричной системы счисления.
2. Какое минимальное основание должна иметь система счисления, если в ней могут быть записаны числа:  
403, 561, 666, 125?  
Переведите:
  - a)  $1314 \rightarrow A_{10}$
  - b)  $125,346 \rightarrow A_{10}$
  - c)  $23410 \rightarrow A_2$
  - d)  $14210 \rightarrow A_4$
3. В какой системе счисления справедливо равенство:  $2 \times 3 = 11$ .
4. Какое число предшествует числу 109 в 9-ричной системе счисления

## Практическая работа №5

**Тема:** «Составление таблиц истинности логических функций и выражений»

**Цель:** освоить алгоритм построения таблиц истинности для логических функций; научиться определять и анализировать функции проводимости переключательных схем.

### **Обеспечение практической работы:**

- задания для выполнения работы;
- материал лекций

### **В результате выполнения данной работы студент должен уметь:**

- строить схемы из элементарных логических элементов по заданному аналитическому выражению функции;
- производить синтез и анализ аналитических выражений логических функций.

### **Методические указания**

#### **Часть 1. Построение таблиц истинности для логических функций**

**Логическая функция** - это функция, в которой переменные принимают только два значения: *логическая единица* или *логический ноль*. Истинность или ложность сложных суждений представляет собой функцию истинности или ложности простых. Эту функцию называют *булевой функцией суждений*  $f(a, b)$ .

Любая логическая функция может быть задана с помощью таблицы истинности, в левой части которой записывается набор аргументов, а в правой части - соответствующие значения логической функции. При построении таблицы истинности необходимо учитывать порядок выполнения логических операций.

#### **Порядок выполнения логических операций в сложном логическом выражении:**

1. инверсия;
2. конъюнкция;
3. дизъюнкция;
4. импликация;
5. эквивалентность.

Для изменения указанного порядка выполнения операций используются скобки.

На основе анализа логической связи между высказываниями делается логический вывод. Для получения логического вывода составляется *таблица истинности*, в которой записывают все возможные комбинации каждого простого высказывания.

#### **Алгоритм построения таблиц истинности для сложных выражений:**

1. Определить количество строк:  
 $\text{количество строк} = 2^n + \text{строка для заголовка}$ ,  
 $n$  - количество простых высказываний.
2. Определить количество столбцов:  
 $\text{количество столбцов} = \text{количество переменных} + \text{количество логических операций}$ ;
- определить количество переменных (простых выражений);
- определить количество логических операций и последовательность их выполнения.
3. Заполнить столбцы результатами выполнения логических операций в обозначенной последовательности с учетом таблиц истинности основных логических операций.

## Логические операции

### Инверсия (логическое отрицание)

Операция, выражаемая словом "не", называется *логическим отрицанием* (*инверсией*) делает истинное выражение ложным и, наоборот, ложное – истинным. Обозначается « $\bar{\phantom{A}}$ ».

Обозначение: НЕ,  $\overline{OA}$ ,  $\bar{A}$ , NOT A

Таблица истинности для логического выражения  $A$  имеет вид

A	$\bar{A}$
1	0
0	1

### Конъюнкция (логическое умножение)

Операция, выражаемая связкой "и", называется *логическим умножением* (конъюнкцией) и обозначается "U" (может также обозначаться знаками «?» (точка) или &). Высказывание  $A \cup B$  истинно тогда и только тогда, когда оба высказывания  $A$  и  $B$  истинны.

Обозначение:  $A$  и  $B$ ,  $A \cup B$ ,  $A \& B$ ,  $A$  AND  $B$

Таблица истинности для логических переменных  $A$  и  $B$

A	B	$A \cup B$
1	1	1
1	0	0
0	1	0
0	0	0

### Дизъюнкция (логическое сложение)

Операция, выражаемая связкой "или" (в неисключающем смысле этого слова), называется *логическим сложением* (дизъюнкцией) и обозначается знаком  $\cup$  (или +). Высказывание  $A \cup B$  ложно тогда и только тогда, когда оба высказывания  $A$  и  $B$  ложны.

Обозначение:  $A$  ИЛИ  $B$ ,  $A \cup B$ ,  $A + B$ ,  $A$  OR  $B$

Таблица истинности для логических переменных  $A$  и  $B$

A	B	$A \cup B$
1	1	1
1	0	1
0	1	1
0	0	0

В алгебре логики любую логическую функцию можно выразить через основные логические операции, записать ее в виде логического выражения и упростить ее, применяя законы логики и свойства логических операций. По формуле логической функции легко рассчитать ее таблицу истинности. Необходимо только учитывать порядок выполнения логических операций (приоритет) и скобки. Операции в логическом выражении выполняются слева направо с учетом скобок.

*Приоритет выполнения логических операций:*

- инверсия,
- конъюнкция,
- дизъюнкция

### Задание 1

Составить таблицу истинности для заданного логического выражения. Вид выражения выбрать из табл. 1.11 в соответствии с номером варианта.

### Варианты для задания 1

#### Таблица 1.11

Варианты для задания 1

Вариант 1	Вариант 6	Вариант 11
$(x + \bar{y}) \mid (x \sim yz)$	$((x \vee \bar{y})z) \rightarrow ((x \sim z) + y)$	$((x \sim z) + y) \cdot (x \mid yz)$
Вариант 2	Вариант 7	Вариант 12
$\bar{x} \rightarrow (z \sim (y + x\bar{z}))$	$(x \vee \bar{y})z \rightarrow ((x \downarrow y) \mid z)$	$\bar{x} \mid \rightarrow (\bar{z} \sim (y + xz))$
Вариант 3	Вариант 8	Вариант 13
$(xz \rightarrow y) \mid (xy + xz)$	$(x + yz) \mid (xy)$	$(x \sim (\bar{y} + z)) \mid (xy)$
Вариант 4	Вариант 9	Вариант 14
$(\bar{x} \vee y) \downarrow z \rightarrow ((x + y) \mid z)$	$(x + yz) \rightarrow (z \sim (y \downarrow (x \vee z)))$	$(x\bar{y}) \mid (x \sim yz)$
Вариант 5	Вариант 10	Вариант 15
$(x \sim \bar{y}) \mid (x \downarrow (\bar{y}z + \bar{x}))$	$((x \vee \bar{y})z) \rightarrow (x \sim y)$	$((x \downarrow \bar{y})z) \rightarrow ((x \sim z) + y)$



### Пример выполнения задания

Составим таблицу истинности для выражения  $(x \sim z) \mid ((x \blacksquare y) \sim (y \blacksquare z))$ . Заданное выражение содержит три логические переменные. Столбцы 1, 2, 3 таблицы истинности (табл. 1.12) содержат возможные комбинации значений переменных  $x, y, z$ . Таблица истинности будет содержать 8 строк по количеству возможных различных комбинаций значений переменных. Таблица истинности с промежуточными результатами вычислений

1	2	3	4	5	6	7	8
$x$	$y$	$z$	$\sim z$	$\bullet y$	$\bullet z$	$(x \bullet y) \sim (y \bullet z)$	$(x \sim z) \mid ((x y) \sim (yz))$
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	0	1	0	0	1	0
0	1	1	0	0	1	0	1
1	0	0	0	0	0	1	1
1	0	1	1	0	0	1	0
1	1	0	0	1	0	0	1
1	1	1	1	1	1	1	0

Таблица 1.12

Таблица истинности с промежуточными результатами вычислений

Столбцы 4, 5, 6, 7 содержат результаты промежуточных вычислений, в столбце 8 находятся значения логического выражения для каждой комбинации значений переменных.

### Задание 2

Построить функциональную схему для заданной логической функции  $F$ . Вид выражения выбрать из табл. 1.13 в соответствии с номером варианта.

Варианты для задания 2

Таблица 1.13

Вариант 1	Вариант 6	Вариант 11
$F = A + B + A \blacksquare B$	$F = A \vee (B + C) + AB$	$F = C + AB + (AB + A)$
Вариант 2	Вариант 7	Вариант 12
$F = (A \vee C) A + (B \vee C)$	$F = (A \vee B)(A + B)$	$F = (B \vee C) + A \vee AB$
Вариант 3	Вариант 8	Вариант 13
$f = AB + c + A$	$F = AB \vee AC + C$	$F = A \vee B + CA$
Вариант 4	Вариант 9	Вариант 14
$F = A \vee B + A \blacksquare B$	$F = (AB + C) \vee BC$	$F = A + B(A \vee C)$
Вариант 5	Вариант 10	Вариант 15
$F = CA \vee (A + B)$	$F = AC + CvB$	$F = A(B + C) + A \vee B$

### Пример выполнения задания

Допустим, задана логическая функция:  $F = BA + BA + CB$ . Построение схемы удобнее выполнять, начиная с ее выхода.

**Первый этап.** Функция  $F$  рассматривается как результат операции «ИЛИ»

(логическое сложение), примененной к функциям  $BA, BA$  и  $CB$ . Используем логический элемент «ИЛИ» и получаем первый фрагмент схемы (рис. 1.8).

**Второй этап.** К входам элемента «ИЛИ» подключаются логические элементы «И», входными переменными которых являются  $A$ ,  $B$ ,  $C$  и их инверсии (рис.1.9).

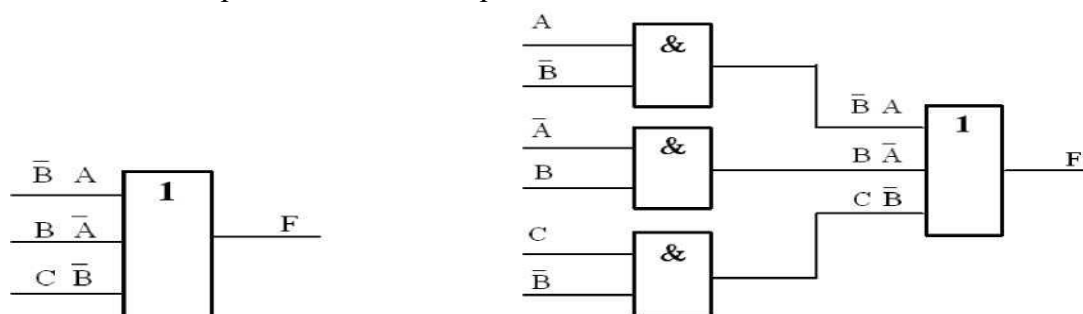


Рис. 1.8. Первый этап построения

**Третий этап.** Для получения инверсий для  $A$  и  $B$  на соответствующих входах ставим инверторы и соединяем одноименные входные переменные в одну линию (рис. 1.10). Построение схемы закончено.

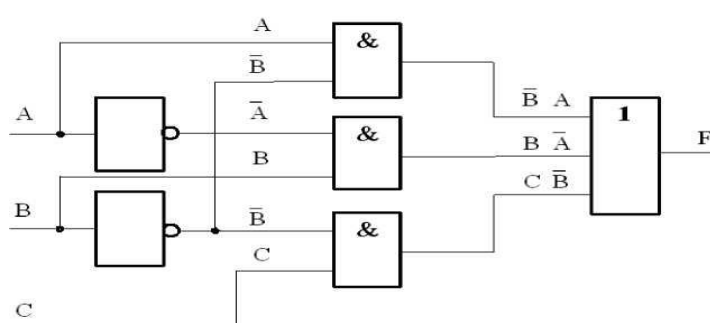


Рис. 1.10. Результат построения схемы

**Вопросы для самоконтроля:**

1. Каков порядок выполнения логических операций?
2. По какому правилу выполняется операция отрицание?
3. По какому правилу выполняется операция конъюнкция?
4. По какому правилу выполняется операция дизъюнкция?
5. По какому правилу выполняется операция импликация?
6. По какому правилу выполняется операция эквиваленция?
7. Что называют таблицей истинности?
8. Каков алгоритм построения таблицы истинности?
9. Какие высказывания называют: тождественно истинными, тождественно ложными, равносильными?

## Практическая работа №6

Тема: «Самостоятельное решение задач»

Цель: освоить алгоритм построения таблиц истинности для логических функций.  
научиться определять и анализировать функции проводимости переключательных схем.

### Обеспечение практической работы:

- задания для выполнения работы;
- материал лекций

### В результате выполнения данной работы студент должен уметь:

- строить схемы из элементарных логических элементов по заданному аналитическому выражению функции;
- производить синтез и анализ аналитических выражений логических функций

### Задания для самостоятельной работы:

#### Задание 1

- |    |   |    |  |
|----|---|----|--|
| 1  | $F = A \vee \bar{B} \vee (\bar{A} \vee C)$          | 16 | $F = A \leftrightarrow C \vee B \rightarrow A$                 |
| 2  | $F = A \rightarrow \bar{B} \vee C$                  | 17 | $F = A \leftrightarrow \bar{C} \vee B \rightarrow \bar{A}$     |
| 3  | $F = B \vee (\bar{A} \leftrightarrow C)$            | 18 | $F = (A \leftrightarrow C) \vee (B \rightarrow A)$             |
| 4  | $F = \bar{B} \vee (A \leftrightarrow C)$            | 19 | $F = A \leftrightarrow C \vee (B \rightarrow \bar{A})$         |
| 5  | $F = A \wedge B \rightarrow \bar{B} \wedge C$       | 20 | $F = A \leftrightarrow (C \vee B \rightarrow A)$               |
| 6  | $F = A \wedge B \leftrightarrow \bar{B} \vee C$     | 21 | $F = (\bar{A} \leftrightarrow C) \vee B \rightarrow A$         |
| 7  | $F = (A \vee \bar{B}) \vee (A \rightarrow C)$       | 22 | $F = \bar{A} \leftrightarrow (C \vee \bar{B} \rightarrow A)$   |
| 8  | $F = (A \rightarrow \bar{B}) \vee C$                | 23 | $F = A \wedge (B \rightarrow \bar{C}) \wedge C$                |
| 9  | $F = B \vee C \leftrightarrow \bar{A} \vee \bar{C}$ | 24 | $F = A \wedge (B \leftrightarrow \bar{A}) \vee C$              |
| 10 | $F = \bar{B} \vee (A \wedge C \rightarrow B)$       | 25 | $F = (C \vee \bar{B}) \vee (\bar{A} \vee C)$                   |
| 11 | $F = A \vee B \rightarrow \bar{B} \vee C$           | 26 | $F = A \rightarrow \bar{B} \vee (C \rightarrow B)$             |
| 12 | $F = A \wedge B \leftrightarrow \bar{B} \vee C$     | 27 | $F = (A \wedge B \rightarrow \bar{B}) \wedge (C \vee \bar{A})$ |
| 13 | $F = A \rightarrow \bar{B} \vee (\bar{A} \vee C)$   | 28 | $F = \bar{B} \vee (A \leftrightarrow C) \wedge C$              |
| 14 | $F = \bar{A} \wedge B \rightarrow \bar{B} \vee C$   | 29 | $F = A \wedge B \rightarrow \bar{B} \wedge C$                  |
| 15 | $F = B \vee (\bar{A} \leftrightarrow C) \wedge A$   | 30 | $F = A \wedge B \leftrightarrow \bar{B} \vee C$                |

#### Задание 2

- |    |   |    |   |
|----|---|----|---|
| 1  | $(A \leftrightarrow B) \vee A \bar{B} \vee C$                         | 16 | $B \vee (A \leftrightarrow CB) \vee A \bar{C}$                |
| 2  | $(A \rightarrow B) \vee A \bar{C} \vee BC$                            | 17 | $(AC \rightarrow B) \vee A \bar{B} \bar{C}$                   |
| 3  | $(AC \rightarrow B) \vee A \bar{C}$                                   | 18 | $(\bar{A} \leftrightarrow C) \vee (B \bar{C} \rightarrow AB)$ |
| 4  | $A \bar{B} \vee (A \leftrightarrow C) B$                              | 19 | $(B \rightarrow C) \vee (B \rightarrow AC)$                   |
| 5  | $(\bar{A} \rightarrow B) \vee (A \bar{C} \vee BC)$                    | 20 | $(AB \rightarrow C) \vee A \vee \bar{A} \bar{C}$              |
| 6  | $(A \leftrightarrow C) \vee A \bar{B} \vee AC$                        | 21 | $(A \leftrightarrow C) \vee (A \bar{B} \rightarrow C)$        |
| 7  | $(A \leftrightarrow C) \vee A \bar{B} \vee BC$                        | 22 | $(\bar{A} \bar{B} \rightarrow \bar{C}) \vee ABC$              |
| 8  | $(C \leftrightarrow B) \vee A \bar{C} \vee BC$                        | 23 | $(AB \rightarrow C) \vee A \bar{C}$                           |
| 9  | $(BC \rightarrow A) \vee A \bar{C}$                                   | 24 | $(\bar{A} \rightarrow BC) \vee (A \leftrightarrow C)$         |
| 10 | $(AB \rightarrow C) \vee A \bar{C}$                                   | 25 | $(\bar{A} \leftrightarrow B) \vee (A \rightarrow BC)$         |
| 11 | $(\bar{A} \rightarrow C) \vee (B \bar{C} \vee AB)$                    | 26 | $(A \rightarrow \bar{B}) \vee (C \bar{A} \rightarrow B)$      |
| 12 | $(\bar{A} \leftrightarrow B) \vee (A \rightarrow BC)$                 | 27 | $(A \rightarrow \bar{B} \bar{C}) \vee A \bar{B} \vee BC$      |
| 13 | $(B \rightarrow C) \vee A \bar{B} \vee \bar{A} \bar{C}$               | 28 | $(A \rightarrow C) \vee A \bar{B} \vee BC$                    |
| 14 | $(A \rightarrow \bar{B} \bar{C}) \vee A \bar{B} \vee \bar{B} \bar{C}$ | 29 | $(A \rightarrow \bar{B}) \vee (B \bar{A} \rightarrow C)$      |
| 15 | $(AC \rightarrow \bar{B}) \vee B \bar{C}$                             | 30 | $(AB \rightarrow \bar{C}) \vee A \bar{B} \bar{C}$             |

## Практическое занятие № 7-8.

**Тема:** «Программное обеспечение внешних устройств. Подключение внешних устройств к компьютеру и их настройка».

**Цель работы:** выработать практические навыки обмена файлами между пользователями локальной компьютерной сети.

### Методические указания

**Абоненты сети**- объекты, генерирующие или потребляющие информацию.

Абонентами сети могут быть отдельные ЭВМ, промышленные роботы, станки с ЧПУ (станки с числовым программным управлением) и т.д. Любой абонент сети подключён к станции.

**Станция**- аппаратура, которая выполняет функции, связанные с передачей и приёмом информации.

Для организации взаимодействия абонентов и станции необходима физическая передающая среда.

**Физическая передающая среда**- линии связи или пространство, в котором распространяются электрические сигналы, и аппаратура передачи данных.

Одной из основных характеристик линий или каналов связи является скорость передачи данных (пропускная способность).

**Скорость передачи данных**- количество бит информации, передаваемой за единицу времени.

Обычно скорость передачи данных измеряется в битах в секунду (бит/с) и кратных единицах Кбит/с и Мбит/с.

Соотношения между единицами измерения: 1 Кбит/с =1024 бит/с; 1 Мбит/с =1024 Кбит/с; 1 Гбит/с =1024 Мбит/с.

**Топология локальных сетей.** Под топологией компьютерной сети обычно понимают физическое расположение компьютеров сети относительно друг друга и способ соединения их линиями.

Топология определяет требования к оборудованию, тип используемого кабеля, методы управления обменом, надежность работы, возможность расширения сети. Существует три основных вида топологии сети: шина, звезда и кольцо.

### Задания для самостоятельной работы студентов

**Задание №1.** Определение общих ресурсов компьютера. Для этого:

- В операционной системе Windows найти на рабочем столе значок Сеть.
- Открыть папку, где будут видны все компьютеры, которые подключены в одну сеть.
- В данном окне появятся все компьютеры, которые подключены к сети. Открыть один из них. Посмотреть ресурсы компьютера, которыми можно воспользоваться. Такие ресурсы называются общими.

**Задание № 2.**

Предоставить доступ для пользователей локальной сети к папке на своем компьютере, подключенном к локальной сети. Для этого:

- В операционной системе Windows открыть окно папки Компьютер и на одном из дисков C: или D: создать свою папку. Назвать ее номером своей группы.
- Щелкнуть правой кнопкой мыши по значку папки и в контекстном меню папки выбрать команду Общий доступ.
- В появившемся диалоговом окне Дополнительный общий доступ установить флажок Открыть общий доступ к этой папке.
- Если все правильно сделано, то на диске (у вашей папки) появится значок, который показывает, что папка является общей.

**Задание №3.** Осуществить проверку возможности доступа к ресурсам компьютеров, подключенных к локальной сети. Для этого:

- Щелкнуть по значку Сеть, в окне появится список компьютеров, подключенных к локальной сети (смотри задание 1.)
- Открыть свой компьютер и внимательно посмотреть: какие из ресурсов доступны пользователям. Если название Вашей папки есть в перечне, то все сделано правильно.

**Задание №4.** Максимальная скорость передачи данных в локальной сети 100 Мбит/с. Сколько страниц текста можно передать за 1 сек, если 1 страница текста содержит 50 строк и на каждой строке - 70 символов?

**Задание №5.** Ответьте на вопросы:

1. Указать основное назначение компьютерной сети.
2. Указать объект, который является абонентом сети.
3. Указать основную характеристику каналов связи.
4. Что такое локальная сеть, глобальная сеть?
5. Что понимается под топологией локальной сети?
6. Какие существуют виды топологии локальной сети?
7. Охарактеризуйте кратко топологию «шина», «звезда», «кольцо».
8. Что такое протокол обмена?

## Практическая работа № 9

Тема: «Хранение информационных объектов различных видов на различных цифровых носителях. Определение объемов различных носителей информации».

Цели занятия:

- закрепить знания о программном обеспечении;
- изучить методы работы с программным обеспечением.
- научиться устанавливать и деинсталлировать программы.
- научиться пользоваться образовательными информационными ресурсами, искать нужную информацию с их помощью;

Оборудование, программное обеспечение: ПК, ОС Windows.

### Методические рекомендации

**Информационный объект** — это совокупность логически связанной информации.

Для хранения и передачи электронных информационных объектов используют съемные цифровые носители. К ним относятся:

- **съемный жесткий диск** — устройство хранения информации, основанное на принципе магнитной записи, информация записывается на жёсткие (алюминиевые или стеклянные) пластины, покрытые слоем ферромагнитного материала,
- **дискета** — портативный носитель информации, используемый для многократной записи и хранения данных, представляющий собой помещённый в защитный пластиковый корпус гибкий магнитный диск, покрытый ферромагнитным слоем,
- **компакт-диск** — оптический носитель информации в виде пластикового диска с отверстием в центре, процесс записи и считывания информации которого осуществляется при помощи лазера (CD-ROM и DVD-диск - предназначенный только для чтения; CD-RW и DVD-RW информация может записываться многократно),
- **карта памяти или флеш-карта** — компактное электронное запоминающее устройство, используемое для хранения цифровой информации (они широко используются в электронных устройствах, включая цифровые фотоаппараты, сотовые телефоны, ноутбуки, MP3-плееры и игровые консоли),
- **USB-флеш-накопитель (сленг. флэшка)** — запоминающее устройство, использующее в качестве носителя флеш-память и подключаемое к компьютеру или иному считывающему устройству по интерфейсу USB.

Все программы и данные хранятся в долговременной (внешней) памяти компьютера в виде файлов.

**Файл** — это определенное количество информации (программа или данные), имеющее имя и хранящееся в долговременной (внешней) памяти.

Имя файла состоит из двух частей, разделенных точкой: собственно имя файла и расширение, определяющее его тип (программа, данные и т. д.). Собственно имя файлу дает пользователь, а тип файла обычно задается программой автоматически при его создании.

Тип файла	Расширение
Исполняемые программы	exe, com
Текстовые файлы	txt, rtf, doc
Графические файлы	bmp, gif, jpg, png, pds и др.
Web-страницы	htm, html
Звуковые файлы	wav, mp3, midi, kar, ogg
Видеофайлы	avi, mpeg

В операционной системе Windows имя файла может иметь до 255 символов, причем допускается использование русского алфавита, разрешается использовать пробелы и другие ранее запрещенные символы, за исключением следующих девяти: \/:\*?"<>|. В имени файла можно использовать несколько точек. Расширением имени считаются все символы, стоящие за последней точкой.

Роль расширения имени файла чисто информационная, а не командная. Если файлу с рисунком присвоить расширение имени TXT, то содержимое файла от этого не превратится в текст. Его можно просмотреть в программе, предназначенной для работы с текстами, но ничего вразумительного такой просмотр не даст.

Атрибуты файла устанавливаются для каждого файла и указывают системе, какие операции можно производить с файлами. Существует четыре атрибута:

- только чтение (R);
- архивный (A);
- скрытый (H);
- системный (S).

#### **Атрибут файла «Только чтение».**

Данный атрибут указывает, что файл нельзя изменять. Все попытки изменить файл с атрибутом «только чтение», удалить его или переименовать завершатся неудачно.

#### **Атрибут файла «Скрытый».**

Файл с таким атрибутом не отображается в папке. Атрибут можно применять также и к целым папкам. Надо помнить, что в системе предусмотрена возможность отображения скрытых файлов, для этого достаточно в меню Проводника Сервис – Свойства папки – вкладка Вид – Показывать скрытые файлы и папки.

#### **Атрибут файла «Архивный».**

Такой атрибут имеют практически все файлы, его включение/отключение практически не имеет никакого смысла. Использовался атрибут программами резервного копирования для определения изменений в файле.

#### **Атрибут файла «Системный».**

Этот атрибут устанавливается для файлов, необходимых операционной системе для стабильной работы. Фактически он делает файл скрытым и только для чтения. Самостоятельно выставить системный атрибут для файла невозможно.

Для изменения атрибутов файла необходимо открыть окно его свойств и включить соответствующие опции.

Существуют также дополнительные атрибуты, к ним относятся атрибуты индексирования и архивации, а также атрибуты сжатия и шифрования.

При передаче и хранении различных файлов необходимо учитывать объем этих файлов. Если объем слишком велик, можно создать архив файлов с помощью программ архиваторов (7-zip, WinRAR, WinZip).

**Архивация** – это сжатие файлов, то есть уменьшение их размера.

При создании архивов исполняемые программы, текстовые файлы, графические файлы, Web-страницы, звуковые файлы, видео файлы сжимаются по-разному.

#### **Запись информации.**

**Запись информации** - это способ фиксирования информации на материальном носителе.

Способы записи информации на компакт-диски:

1) **с помощью специальных программ записи** (Nero, CDBurnerXP, Burn4Free, CD DVD Burning и др.);

2) **через задачи для записи CD** (помещаем нужные объекты на диск с помощью перетаскивания или копирования, выбираем в задачах записи CD «записать файлы на компакт-диск»).

**Способы записи информации на остальные съемные цифровые носители:**

1) **копирование** (выделяем нужные объекты, нажимаем правой кнопкой мыши, в появившемся контекстном меню выбираем «копировать»; через контекстное меню правой кнопки мыши, выбирая «вставить», вставляем объекты на нужный цифровой носитель);

2) **перетаскивание** (выделяем нужные объекты, нажимаем левую кнопку мыши, удерживая её, перетаскиваем документы на нужный цифровой носитель).

**Содержание работы:**

**Задание 1**

**Письменно ответьте на вопросы:**

- 1 Носители информации - это
- 2 Пакет магнитных дисков, надетых на общую ось это –
- 3 Что является оптическим носителем информации?
- 4 В целях сохранения информации гибкие магнитные диски необходимо оберегать от ...
- 5 Что было самым первым носителем магнитной записи?
- 6 Материальная среда, используемая для записи и хранения информации
- 7 Какой компьютерный носитель информации вышел из массового употребления?
- 8 На ЭВМ первого поколения сменным носителем информации для устройств внешней памяти была ...
- 9 Что участвует в записи и чтении CD и DVD- дисков?
- 10 В какой системе счисления хранится информация в компьютере?

**Задание 2**

**Сделайте вывод о проделанной работе**



## Практическая работа № 10-11

Тема: «Создание архива данных. Извлечение данных из архива».

Цели занятия:

- изучение принципов архивации файлов, функций и режимов работы наиболее распространенных архиваторов,
- приобретение практических навыков работы по созданию архивных файлов и извлечению файлов из архивов.

**Оборудование, программное обеспечение:** ПК, ОС Windows, программа для архивации данных WinZip, WinRar.

### Методические указания к практической работе

**Архивация** (упаковка) — помещение (загрузка) исходных файлов в архивный файл в сжатом или несжатом виде.

Для архивации используются специальные программы, архиваторы, осуществляющие упаковку и позволяющие уменьшать размер архива, по сравнению с оригиналом, примерно в два и более раз.

Архиваторы позволяют защищать созданные ими архивы паролем, сохранять и восстанавливать структуру подкаталогов, записывать большой архивный файл на несколько дисков (многотомный архив).

Сжиматься могут как один, так и несколько файлов, которые в сжатом виде помещаются в так называемый архивный файл или архив. Программы большого объема, распространяемые на дискетах, также находятся на них в виде архивов.

**Архивный файл** — это специальным образом организованный файл, содержащий в себе один или несколько файлов в сжатом или несжатом виде и служебную информацию об именах файлов, дате и времени их создания или модификации.

**Разархивация** (распаковка) — процесс восстановления файлов из архива в первоначальном виде. При распаковке файлы извлекаются из архива и помещаются на диск или в оперативную память.

**Самораспаковывающийся архивный файл** — это загрузочный, исполняемый модуль, который способен к самостоятельной разархивации находящихся в нем файлов без использования программы-архиватора.

Самораспаковывающийся архив получил название SFX-архив (Self-eXtracting). Архивы такого типа в обычно создаются в форме .EXE-файла.

Архиваторы, служащие для сжатия и хранения информации, обеспечивают представление в едином архивном файле одного или нескольких файлов, каждый из которых может быть при необходимости извлечен в первоначальном виде. В *оглавлении архивного файла* для каждого содержащегося в нем файла хранится следующая информация:

- имя файла;
- сведения о каталоге, в котором содержится файл;
- дата и время последней модификации файла;
- размер файла на диске и в архиве;
- код циклического контроля для каждого файла, используемый для проверки целостности архива.

**Архиваторы имеют следующие функциональные возможности:**

1. Уменьшение требуемого объема памяти для хранения файлов от 20% до 90% первоначального объема.
2. Обновление в архиве только тех файлов, которые изменялись со времени их последнего занесения в архив, т.е. программа-упаковщик сама следит за изменениями, внесенными пользователем в архивируемые файлы, и помещает в архив только новые и измененные файлы.
3. Объединение группы файлов с сохранением в архиве имен директорий с именами файлов, что позволяет при разархивации восстанавливать полную структуру директорий и файлов.
4. Написания комментариев к архиву и файлам в архиве.

5. Создание саморазархивируемых архивов, которые для извлечения файлов не требуют наличия самого архиватора.
6. Создание многотомных архивов – последовательности архивных файлов. Многотомные архивы предназначены для архивации больших комплексов файлов на дискеты.

### Содержание работы:

#### Задание №1.

1. В операционной системе Windows создайте папку **Archives** по адресу **C:\TEMP**. Создайте папки **Pictures** и **Documents** по адресу **C:\TEMP\Archives**.
2. Найдите и скопируйте в папку **Pictures** по два рисунка с расширением **\*.jpg** и **\*.bmp**.
3. Сравните размеры файлов **\*.bmp** и **\*.jpg**. и запишите данные в таблицу\_1.
4. В папку **Documents** поместите файлы **\*.doc** (не менее 3) и запишите их исходные размеры в таблицу\_1.

#### Задание №2. Архивация файлов WinZip

1. Запустите **WinZip 7**. (Пуск → **Все программы** → **7-Zip** → **7 ZipFileManager**).
2. В появившемся диалоговом окне выберите папку, в которой будет создан архив: **C:\TEMP\Archives\Pictures**. Установите курсор на имя графического файла **Зима.jpg**. Выполните команду **Добавить (+)**.
3. Введите имя архива в поле **Архив – Зима.zip** и убедитесь, что в поле **Формат архива** установлен тип **Zip**.
4. Установите в поле **Режим изменения:** *добавить и заменить*.
5. В раскрывающемся списке **Уровень сжатия:** выберите пункт **Нормальный**. Запустите процесс архивации кнопкой **ОК**.
6. Сравните размер исходного файла с размером архивного файла. Данные запишите в таблицу\_1.
7. Создайте архив **Зима1.zip**, защищенный паролем. Для ввода пароля в диалоговом окне **Добавит к архиву** в поле **Введите пароль:** ведите пароль, в поле **Повторите пароль:** подтвердите пароль. Обратите внимание на флажок **Показать пароль**. Если он не установлен, пароль при вводе не будет отображаться на экране, а его символы будут заменены подстановочным символом **"\*"**. Это мера защиты пароля от посторонних. Однако в данном случае пользователь не может быть уверен в том, что он набрал пароль правильно. Поэтому при не установленном флажке система запрашивает повторный (контрольный) ввод пароля. Щелкните на кнопке **ОК** - начнется процесс создания защищенного архива.
8. Выделите архив **Зима1.zip**, выполните команду **Извлечь**. В появившемся диалоговом окне **Извлечь** в поле **Распаковать в:** выберите папку-приемник - **C:\TEMP\Archives\Pictures\Зима1\**.
9. Щелкните на кнопке **ОК**. Процесс извлечения данных из архива не запустится, а вместо него откроется диалоговое окно для ввода пароля.
10. Убедитесь в том, что ввод неправильного пароля не позволяет извлечь файлы из архива.
11. Убедитесь в том, что ввод правильного пароля действительно запускает процесс.
12. Удалите созданный вами защищенный архив и извлеченные файлы.
13. Создайте самораспаковывающийся ZIP-архив. Для этого установите курсор на имя архива **Зима.zip**, выполните команду **Добавить (+)**.
14. Введите имя архива в поле **Архив – Зима.7z** и убедитесь, что в поле **Формат архива** установлен тип **7z**.
15. Установите в поле **Режим изменения:** *добавить и заменить*.
16. Установите флажок **Создать SFX-архив**.
17. В раскрывающемся списке **Уровень сжатия:** выберите пункт **Нормальный**. Запустите процесс архивации кнопкой **ОК**.
18. Аналогичным образом создайте архивы для файлов **Рябина.bmp**, **Документ1.doc**, **Документ2.doc**, **Документ3.doc**. Сравнительные характеристики исходных файлов и их архивов занести в таблицу\_1.

#### Задание №3. Архивация файлов WinRar

1. Запустите **WinRar**(Пуск → Все программы → WinRar).
2. В появившемся диалоговом окне выберите папку, в которой будет создан архив: C:\TEMP\Archives\Pictures.
3. Установите курсор на имя графического файла **Зима.jpg**.
4. Выполните команду **Добавить**. В появившемся диалоговом окне введите имя архива **Зима.rar**. Выберите формат нового архива - **RAR**, метод сжатия - **Обычный**. Убедитесь, что в группе **Параметры архивации** ни в одном из окошечек нет флажков. Щелкните на кнопке **ОК** для создания архива. Во время архивации отображается окно со статистикой. По окончании архивации окно статистики исчезнет, а созданный архив станет текущим выделенным файлом.
5. Аналогичным образом создайте архивы для файлов Рябина.bmp, Документ1.doc, Документ2.doc, Документ3.doc. Сравнительные характеристики исходных файлов и их архивов занести в таблицу\_1.
6. Создайте самораспаковывающийся RAR – архив, включающий в себя текстовые и графические файлы.
7. Определите процент сжатия файлов и заполните таблицу\_1. Процент сжатия определяется по формуле  $S = \frac{S_0 - S}{S_0} \cdot 100\%$ , где S – размер архивных файлов, S<sub>0</sub>– размер исходных файлов.  
Таблица\_1

**Текстовые файлы:**

1. Документ1.doc
2. Документ2.doc
3. Документ3.doc

**Графические файлы:**

1. Зима.jpg
- 2.Рябина.bmp

Процент сжатия **текстовой** информации (для всех файлов)

Процент сжатия **графической** информации (для всех файлов)

**Задание №4. Ответить на вопросы:**

1. Что называется архивацией?
2. Для чего предназначена архивация?
3. Какой файл называется архивным?
4. Что называется разархивацией?
5. Какая информация хранится в оглавлении архивного файла?
6. Какие функциональные возможности имеют архиваторы?

**Сделать вывод о проделанной практической работе:**

## **Практическая работа № 12-13**

**Тема: «Операционная система. Графический интерфейс пользователя»**

**Цель работы.** Изучение основных понятий операционной системы и файловой системы. Приобретение умения выполнять создание, копирование, перемещение и удаление файлов и папок

### **План**

- 1) Изучить назначение и основные функции операционной системы
- 2) Изучить основные принципы хранения информации во внешней памяти
- 3) Выполнить создание, копирование, перемещение и удаление файлов и папок
- 4) Ответить на контрольные вопросы

### **Методические указания**

Операционная система (ОС) предназначена для управления выполнением пользовательских программ, планирования и управления вычислительными ресурсами ЭВМ.

Основными функциями ОС являются:

1. загрузка программ в оперативную память (ОП) и управление ходом их выполнения;
2. обеспечение операций по обмену данными между выполняющейся программой и внешними устройствами;
3. обслуживание нестандартных ситуаций в ходе выполнения программы (например, если пользователь по каким-то причинам желает прекратить выполнение программы);
4. удаление выполненной программы из ОП и освобождение места для загрузки новой программы;
5. организация хранения программ и данных на внешних носителях;
6. организация взаимодействия пользователя и операционных систем – прием и выполнение команд пользователя;
7. выполнение различных вспомогательных функций, таких как форматирование дисковых устройств, копирование информации с одного дискового устройства на другое, проверка качества рабочих поверхностей дисковых носителей, служба времени (системные часы);
8. обеспечение защиты данных; и некоторые другие функции.

ОС – комплекс программных средств, обеспечивающих интерфейс нескольких видов:

⇒ интерфейс пользователя (взаимодействие между пользователем и программно-аппаратными средствами);

⇒ аппаратно-программный интерфейс (взаимодействие между аппаратными и программными средствами);

⇒ программный интерфейс (взаимодействие между разными видами ПО).

Операционная система может находиться на одном из внешних запоминающих устройств (гибком, жестком или лазерном диске). А образующие базовую систему ввода/вывода части ОС

записываются в постоянную память (ПЗУ). Диск, на котором размещена ОС, называется системным диском. Загрузка ОС в ОП осуществляется с системного диска при включении ПК.

Организация файловой системы - одна из самых важных функций любой операционной системы. Вся информация на дисках организуется в виде файлов. Файл (от англ. file – картотека, архив) – это именованная область диска для постоянного хранения информации (программ, данных для их работы, текстов, рисунков и т. д.). Каждый файл имеет имя и тип (расширение), которые записываются через точку: имя.тип.

Для каждого файла, кроме имени и расширения, ОС хранит информацию о размере файла, дате и времени его создания или последней модификации, и несколько величин, называемых атрибутами. Атрибуты – дополнительные параметры, определяющие свойства файлов: Read Only (Только для чтения); Hidden (Скрытый); System (Системный); Archive (Архивный).

Файлы по любому общему признаку, выбранному пользователем, объединяются в каталоги (папки). Каталог (папка, folder) – место на диске, в котором хранятся сведения о файлах: их имена и их атрибуты. На логическом уровне каталоги – это элементы иерархической структуры, необходимые для обеспечения удобного доступа к файлам, особенно, если файлов на диске слишком много. Каждый каталог имеет свое имя, задаваемое пользователем при его создании. Каталог может быть вложенным, т. е. находиться в каталоге более высокого уровня. Корневой каталог является самым верхним уровнем вложенности иерархической структуры и организуется на диске ОС при форматировании диска. Корневой каталог обозначается обратной косой чертой (backslash) \ и указывается после имени физических ВЗУ: C:\, D:\ - имена корневых каталогов разделов жесткого диска; E:\ - корневой каталог лазерного диска; A:\, B:\ - корневые каталоги гибких магнитных дисков.

Основным недостатком файлов MSDOS является их небольшой размер. Это недостаток устранен в операционных системах Windows 9x за счет введения нового понятия – «длинного» имени файла. **Длинные имена файлов** обладают следующими свойствами:

- они могут содержать до 255 символов включая расширение;
- они могут содержать пробелы;
- они могут содержать несколько точек;
- в имени файла нельзя ставить следующие символы: / - косую черту (слэш); \ - обратную косую черту (бекслэш); ? - знак вопроса; \* - звездочку; > - знак больше; < - знак меньше; : - двоеточие; " - кавычку; | - вертикальную черту

В отличие от имен файлов, в именах **папок (директорий, каталогов)** расширение обычно не ставится, так как в этом нет особой необходимости.

В пределах одной папки могут находиться сколько угодно файлов, но имена файлов вместе с расширениями должны быть уникальными, то есть не должны повторяться. Зато не запрещено иметь в одной папке несколько файлов с одинаковыми именами, но разными типами: письмо.txt, письмо.doc

В процессе работы с файлами возникает необходимость создавать новые файлы, заменять одни файлы другими, перемещать их с одного места на другое, переименовывать, удалять.

Windows является наиболее популярной операционной системой с графическим интерфейсом и обеспечивает возможность многозадачности - одновременной работы нескольких приложений.

Простой и удобный интерфейс операционных систем семейства Windows обеспечивает естественность общения пользователя с компьютером.

**Интерфейс** пользователя - способ представления информации на экране, связующее звено между пользователем и компьютером.

После загрузки операционной системы Windows вся поверхность монитора - это **Рабочий стол**. На Рабочем столе как на поверхности обычного письменного стола - всё самое нужное для работы: значки, папки, Мой компьютер, ярлыки, корзина. В нижней части Рабочего стола располагается **панель задач**, на ней находится кнопка Start (**Пуск**), с которой начинается и заканчивается работа, а также ярлыки всех загруженных в оперативную память программ, между которыми можно переключаться. **Главное системное меню** "всплывает" после щелчка мыши кнопки Пуск. С его помощью можно выполнить различные операции: запустить программу, открыть документ, вызвать панель управления для настройки компьютера, провести поиск файла или папки, получить справку и т.д. Одно из основных понятий графического интерфейса ОС Windows - это **окна**.

Проводник – служебная программа Windows, предназначенная для навигации по файловой структуре компьютера и ее обслуживания. Цель навигации состоит в обеспечении доступа к нужной папке и ее содержимому. Запуск проводника может осуществляться одним из способов:

- a) Пуск - Программы - Проводник
- b) Контекстное меню кнопки Пуск - Проводник
- c) Контекстное меню стандартной папки Мой компьютер Проводник

Окно Проводника разделено на две области: левую, называемую панелью папок, и правую, называемую панелью содержимого (рис. 1.) В левой панели показана структура папок.

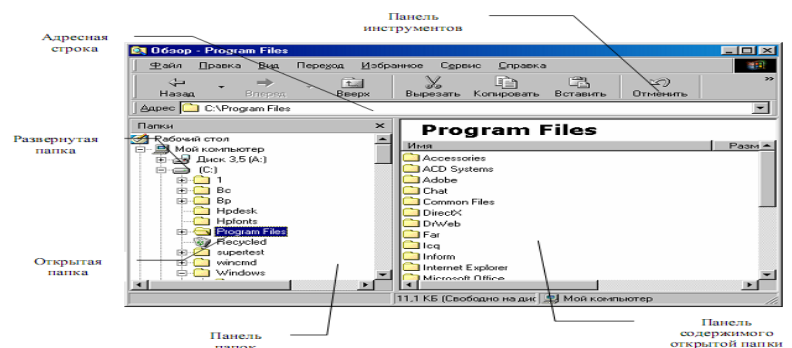


Рис. 1. Окно программы «Проводник»

Каждая папка может быть раскрыта щелчком левой кнопки мыши на ее значке, при этом ее содержимое отображается на правой панели, закрывается папка автоматически при раскрытии любой другой папки. Одна из папок в левой панели раскрыта всегда. Если папка имеет вложенные папки, то на левой панели рядом с папкой отображается узел, отмеченный знаком

«+», с помощью щелчка на котором папку можно развернуть, при этом значок узла меняется на «-». Сворачиваются папки также при помощи щелчка, выполненном на узле. Изменение вида отображаемых в правой части окна объектов (в виде крупных или мелких значков, в виде списка или таблицы) выполняется с помощью команд пункта меню Вид.

### Задания к практической работе

**Задание 1.** Создайте на диске D: в папке Студенты, папку Ваша группа, а в ней папку Ваша фамилия и в ней три папки База данных, Информатика, Презентации; в папке Информатика создайте три папки Тексты, Рисунки, таблицы (рис. 2) с использованием программы Мой компьютер.

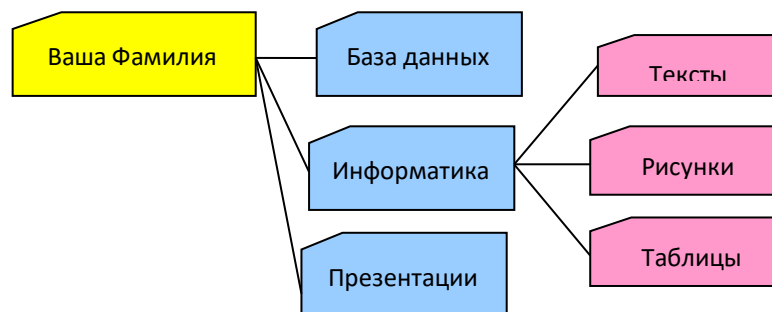


Рис. 2

### Порядок работы

1. Откройте окно Мой компьютер. Установите стиль просмотра Крупные значки (Вид - Крупные значки).
2. Перейдите на диск D: в папку Студенты, создайте папку Ваша группа (Файл – Создать – Папка).
3. Откройте папку Ваша группа и создайте в ней папку Ваша фамилия (Файл – Создать – Папка)
4. Затем откройте папку Ваша фамилия создайте папки База данных, Информатика и Презентации.
5. Откройте папку Информатика и нажмите на панели содержимого папки правую клавишу мыши, затем в контекстном меню выберите Создать – Папку и присвойте имя папки Тексты и нажмите клавишу Enter. Затем создайте папки Рисунки и Таблицы.
6. Закройте окно Мой компьютер (Файл - Закрывать)

**Задание 2.** Создайте рисунок в стандартном приложении WindowsPaint и сохраните его на диск D: в папке Студенты, Ваша группа, Ваша фамилия, Информатика, Рисунки под именем Рисунок\_Фамилия.jpg

### Порядок работы

1. Откройте программу Paint (Пуск - Программы - Стандартные).
2. Задайте ширину рисунка, равную 300 точек и высоту – 200 точек (Рисунок→Атрибуты).
3. Используя различные инструменты и используя различные цвета создайте рисунок, где должны присутствовать объекты, нарисованные с использованием инструментов эллипс, линия, распылитель, карандаш, прямоугольник. Для заливки используйте инструмент Заливка (выбор основного цвета – щелчок левой клавиши, фоновый – правой). Для коррекции использовать инструмент Ластик.

4. Сохраните изображение под именем Рисунок\_Фамилия.jpg (Файл – Сохранить как - D:\Студенты\Ваша группа\Ваша фамилия\Информатика имя файла Рисунок\_Фамилия, тип файла jpg)

5. Закройте окно Paint(Файл - Выход)

**Задание 3.** Создайте текст в стандартном приложении Windows Блокнот и сохраните его на диск D: в папке Студенты, Ваша группа, Ваша фамилия, Информатика, Тексты под именем Блокнот\_Фамилия.txt

#### **Порядок работы**

1. Запустите текстовый редактор Блокнот (Пуск - Программы - Стандартные).

2. Вставьте текущую дату (Правка→Время и дата). Введите с клавиатуры свою фамилию имя и отчество и группу. Нажмите клавишу Enter. Напишите фразу: Практическая работа . Тема: «Операционная система. Графический интерфейс пользователя. Нажмите клавишу Enter.

3. Скопируйте с помощью Буфера обмена (Правка→Копировать и Правка→Вставить) данную фразу 5 раз.

4. Сохраните созданный документ под именем Блокнот\_Фамилия.txt (Файл – Сохранить как - D:\Студенты\Ваша группа\Ваша фамилия\Информатика имя файла Блокнот\_Фамилия, тип файла txt)

5. Закройте окно Блокнот (Файл - Выход)

**Задание 4.** В программе Проводник выполнить копирование файла Рисунок\_Фамилия.jpg из папки Рисунки в папку Презентации; перемещение папки Тексты в папку Ваша фамилия; удалите (удаление папок выполнять только в присутствии преподавателя) папку Информатика.

#### **Порядок работы**

1. Запустите программу Проводник (Пуск – Программы – (Стандартные) - Проводник).

2. На панели папок (в левой части Проводника) откройте папку **Рисунки**, при этом содержимое папки Рисунки отобразится в правой части Проводника. В правой части выделить файл Рисунок\_Фамилия.jpg для копирования и перетащить влево левой клавишей мыши в папку **Презентации** при нажатой клавише Ctrl.

3. На панели папок (в левой части Проводника) откройте папку **Информатика**, при этом содержимое папки **Информатика** отобразится в правой части Проводника. В правой части выделить Папку **Тексты** для перемещения и перетащить влево левой клавишей мыши в папку **Ваша фамилия**.

4. Для удаления папки **Информатика** необходимо в правой части Проводника выделить папку **Информатика** (предварительно в левой части Проводника открыть папку **Ваша фамилия**) для удаления, нажать правую клавишу мыши и выбрать удаление папки.

#### **Вопросы для самоконтроля:**

1. Для чего предназначена операционная система?



2. Каковы функции операционной системы?
3. Какие операционные системы различают по числу обрабатываемых задач? Что такое задача?
4. Какие операционные системы различают по типу интерфейса?
5. Приведите пример операционной системы с интерфейсом командной строки.
6. Приведите пример операционной системы с графическим интерфейсом.
7. Как операционная система управляет работой периферийных устройств компьютера?
8. Что такое драйвер?
9. Что такое файл?
10. Какова структура имени файла?
11. Какие расширения могут иметь программные приложения?
12. Какие расширения могут иметь текстовые файлы?
13. Какие расширения могут иметь графические файлы?
14. Какие расширения могут иметь видеофайлы?
15. Что такое каталог (папка)?
16. Как обозначается корневой каталог?
17. Как открыть и закрыть папку?
18. Что представляет собой программное приложение?
19. Что такое документ?
20. Что составляет основу работы пользователя с операционной системой Windows?
21. Перечислите элементы графического интерфейса Windows?
22. Что представляет собой Рабочий стол в Windows?
23. Что такое меню? Какие виды меню вы знаете?
24. Опишите правила работы с буфером обмена.
25. Для чего предназначен Проводник?

#### **Дополнительная часть**

26. Где располагается операционная система чаще всего?
27. Какая часть операционной системы хранится в постоянном запоминающем устройстве?
28. Что представляет собой WIMP-интерфейс операционной системы?
29. Что представляет собой SILK-интерфейс операционной системы?
30. Чем отличаются стандартные драйверы от загружаемых?
31. Что представляет собой шаблон? Какие символы используют в шаблонах? Приведите примеры.
32. Что такое FAT?
33. Какая информация содержится в FAT?
34. Опишите иерархическую структуру организации каталогов.
35. Какой каталог называется корневым?

36. Чем отличается корневой каталог от подкаталога?
37. Чем отличается имя файла от имени папки?
38. Что такое буфер обмена?
39. Что такое значок (пиктограмма)?
40. Чем отличается ярлык от пиктограммы?
41. Чем отличается счетчик от поля ввода?
42. Что представляет собой список в Windows?
43. Чем отличается кнопка выбора от флажка?
44. Как выполнить создание новой папки в папке Мой компьютер?
45. Как выполнить копирование файлов (папок) в программе Проводник?
46. Как выполнить перемещение файлов (папок) в программе Проводник?

## **Практическая работа №14**

**Тема: «Файл как единица хранения информации на компьютере. Атрибуты файла и его объем. Учет объемов файлов при их хранении, передаче»**

### **Цели урока:**

1) - помочь учащимся закрепить представление о файлах и файловых системах, именах файлов, пути к файлам, типах файлов и их связи с программами и расширениями имен файлов, дать основные понятия, необходимые для работы на компьютере;

2) - воспитание информационной культуры учащихся, внимательности, аккуратности, дисциплинированности, усидчивости;

3) - развитие познавательных интересов, навыков работы с мышью и клавиатурой, самоконтроля, умения конспектировать

### **Структура урока**

1. Организационный момент
2. Актуализация опорных знаний
3. Практическая работа
4. Закрепление нового материала
5. Домашнее задание
6. Подведение итогов урока.

### **Ход урока:**

**I. Организационный момент.**

**II. Актуализация опорных знаний.**

#### **Задание 1**

Предложите варианты имен и типов для перечисленных ниже файлов.

#### **Содержание**

**Имя**

**Тип**

**Полное имя файла**

Фото моей семьи

Рецепт яблочного пирога

Буклет Моя школа

Открытое письмо Биллу Гейтсу

BillG

doc

BillG.doc

Семейный альбом Моя родословная

Репродукция картины Малевича

Чёрный квадрат

Петиция директору школы об увеличении количества уроков информатики

Реферат по истории

Реклама концерта рок-группы

Статья в журнал

Информатика и образование

## **Задание 2**

Предложите варианты программ, открывающих файлы с тем или иным расширением.

Расширение имени файла

Программа

ТХТ

DOC

RTF

BMP

ARJ

HTML

## **Задание 3**

Выпиши в один столбик правильные имена файлов, а во второй правильные имена каталогов:

Письмо.18, letter.txt, WinWord, письмо.doc, Школа?234, Мои документы, роза.bmp, stop12.exe, 1С, red.com

#### **Задание 4**

Назовите виды окон Windows. Назовите основные элементы стандартного окна

#### **Задание 5**

Постройте дерево каталогов

C:\Рисунки\Природа\Небо.bmp

C:\Рисунки\Природа\Снег.bmp

C:\Рисунки\Компьютер\Монитор.bmp

C:\Мои документы\Доклад.doc

Ответ

## Практическое занятие № 15

**Тема:** «Современные антивирусные программы».

**Цель занятия:** изучить классификацию вирусов, способы их распространения, способы борьбы с ними; изучить классификацию и назначение антивирусных программ.

### Теоритические сведения


#### *Понятие вируса.*

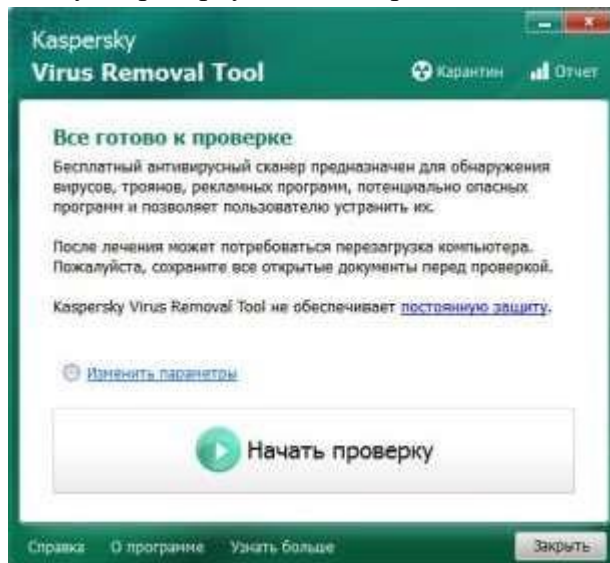
Официальное появление *первого компьютерного вируса* датируется 1981 годом, задолго до выхода первой версии Microsoft Windows. Этот вирус, замаскированный под компьютерную игру, атаковал наиболее популярный компьютер того времени — Apple II. Распространялся он с черепашной скоростью (с помощью дискет).

Согласно подсчетам экспертов, объем *malware* (общепринятое название всех видов вредоносных программ) возрастает более чем на 15 % в год. Согласно данным компании Sophos, разработчика антивирусных программ, каждый день появляются примерно 30 новых вирусов, а перечень активных вирусов пополняется 10 тыс. новых наименований в год.

*Вирус* — это часть программного кода, которая тиражируется путем добавления в другой объект, обычно незаметно и без разрешения пользователя.

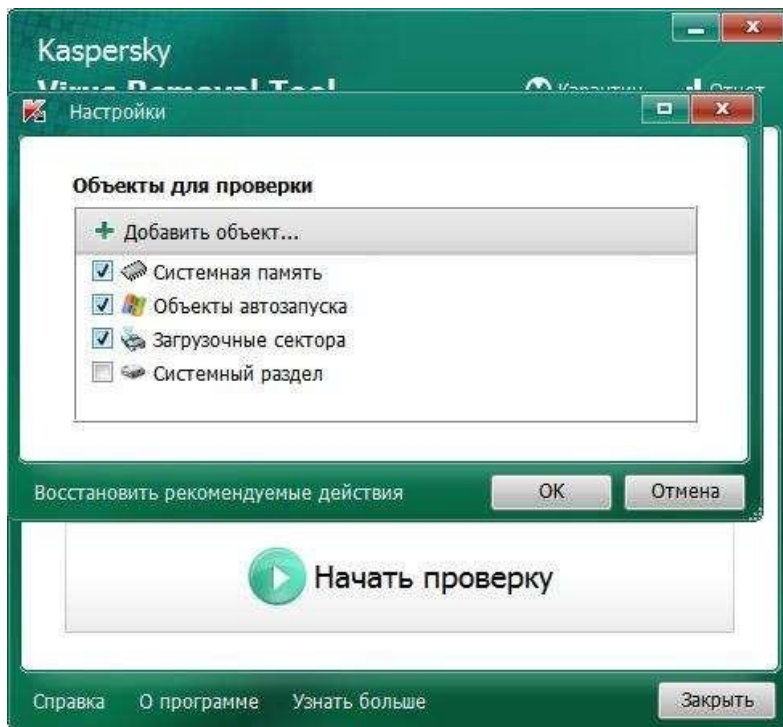
### Задание № 1

**1.1.** Запустите утилиту Антивируса Касперского **Kaspersky Virus Removal Tool** (программа находится в папке материалов PZ-8 и обозначена иконкой ) и выполнить минимальную проверку компьютера.



Сделайте скриншот и вставьте его в протокол выполнения работы.

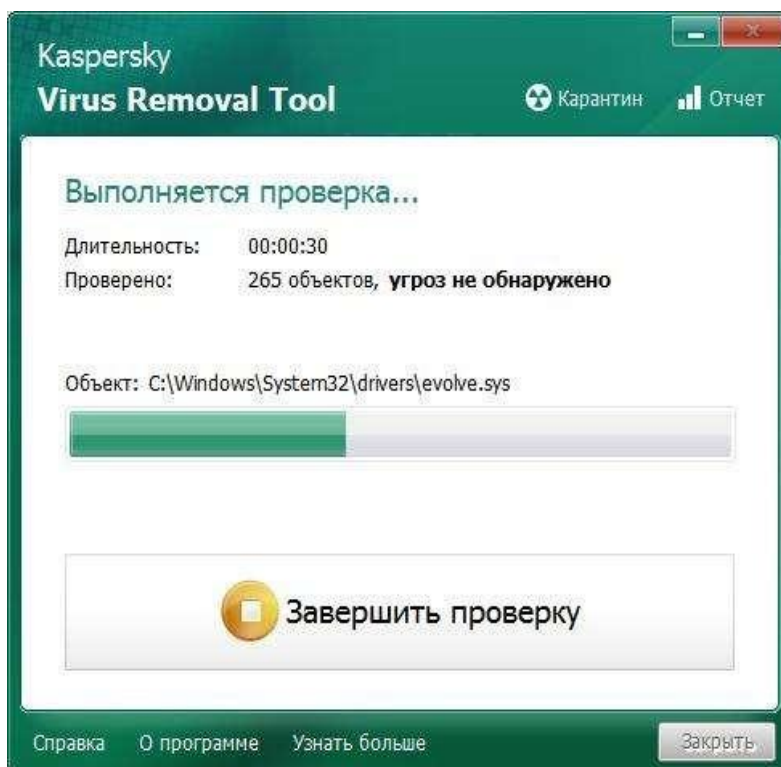
**1.2** Проверьте параметры проверки с помощью ссылки *Изменить параметры*



При указанных параметрах указаны начальные объекты проверки.

*Сделайте скриншот стандартных параметров проверки и вставьте его в протокол выполнения работы.*

1.3 Запустите проверку и дождитесь результатов.



*Сделайте скриншот результатов проверки и вставьте его в протокол выполнения работы.*

2 Измените параметры проверки – назначьте проверку папки с распакованными заданиями практического задания №8 (PZ-8)



*Сделайте скриншот настройки параметров проверки и вставьте его в протокол выполнения работы.*

2.1 Выполните проверку указанной папки и сделайте скриншот результатов проверки и вставьте его в протокол выполнения работы.

2.2 При окончании поиска изучите отчет поиска.

*Сделайте скриншот отчета и карантина проверки и вставьте его в протокол выполнения работы.*

### **Задание: 3-5**

Используя таблицу вариантов вопросов найти ответы и записать их в отчет.

ФИО	Задание 3	Задание 4	Задание 5
Буханова Ольга	1.	30	<b>AVG AntiVirus Free 2017</b>
Вишневская Наталья	2.	29	<b>AppRemover 3.1.24.1</b>
Войтов Алексей	3.	28	<b>Avira Free Antivirus</b>
Габлеева Татьяна	4.	27	<b>Comodo Cleaning Essentials 2.5</b>
Гущина Екатерина	5.	26	<b>McAfee Cloud AV 0.5.157.3 Beta</b>
Ефремова Людмила	6.	25	<b>Panda Free Antivirus 18.01.00</b>
Женевская Анастасия	7.	24	<b>Microsoft Security Essentials RU 4.10.209.0 Final</b>
Жорова Алёна	8.	23	<b>BitDefender Antivirus Free Edition 2017</b>
Журавлёва Вероника	9.	22	<b>Kaspersky Free Anti-Virus 2017</b>
Илюкевич Иван	10.	21	<b>Avast! Free Antivirus 2017</b>
Карасёв Владимир	11.	20	<b>360 Total Security</b>
Кильдюшева Валерия	12.	19	<b>Ad-Aware Free Antivirus+</b>
Коржуев Илья	13.	18	<b>Avetix Antivirus Free 5.0.159</b>



## Перечень вопросов

1. Охарактеризуйте информацию и ее свойства.
2. Что является предметом и объектом защиты информации?
3. Приведите классификацию конфиденциальной информации.
4. Охарактеризуйте свойства достоверности и своевременности информации.
5. Что понимается под угрозой информации?
6. Назовите разновидности угроз информации.
7. Приведите классификацию угроз информации.
8. Какие основные направления и методы реализации угроз Вам известны?
9. Поясните классификацию злоумышленников.
10. Охарактеризуйте причины и виды утечки информации.
11. Назовите и приведите примеры каналов утечки информации.
12. Классификация автоматизированных систем по уровню защищенности.
13. Назначение межсетевых экранов.
14. Классы защищенности межсетевых экранов.
15. Перечислите уровни информационной безопасности.
16. Что понимается под политикой безопасности?
17. Перечислите способы кодирования информации.
18. Методы шифрования информации.
19. Принцип работы электронной цифровой подписи.
20. Назовите существующие системы защиты от несанкционированного доступа к информации.
21. Перечислите каналы утечки информации.
22. Что такое не санкционированный доступ к компьютерным системам?
23. Принцип работы Закладки типа «Программного червя».
24. Принципиальное отличие компьютерного вируса от троянской программы.
25. Способы проникновения вируса на компьютер.
26. Характер воздействия компьютерных вирусов
27. Перечислите основные классы компьютерных вирусов.
28. Когда произошла первая эпидемия «компьютерного вируса»,
29. Перечислите названия безопасных программных утилит антивирусных программ.
30. Перечислите группы антивирусных программ

## Задание № 6

При выполнении задания опишите особенности данной программы, технологии ее работы. Сайт загрузки (разработчика) данной программы. Название антивирусной программы выбрать согласно своему варианту .

## *Содержание отчета*

В начале файла отчета должны быть указаны:

### **Протокол выполнение работы**

**Задание 1** Запустить утилиту Антивируса Касперского (Kaspersky Virus Removal Tool) и выпонить минимальную проверку компьютера.скриншот начала работы программы

- скриншот стандартных параметров проверки.
- скриншот результатов проверки

**Задание 2**Выполнить проверку папки PZ-8

- скриншот настройки параметров проверки папки PZ-8 (на рабочем столе)
- скриншот отчета проверки
- скриншот карантина проверки

## **Практическая работа №16.**

**Тема: «Защита информации, антивирусная защита» Эксплуатационные требования к рабочему месту»**

**Цель занятия:** выработать практические навыки работы с антивирусными программами, навыки правильной работы с компьютером.

### **Методические указания**

Вирусы. Антивирусное программное обеспечение

Компьютерный вирус - программа способная самопроизвольно внедряться и внедрять свои копии в другие программы, файлы, системные области компьютера и в вычислительные сети, с целью создания всевозможных помех работе на компьютере.

Признаки заражения:

- о* прекращение работы или неправильная работа ранее функционировавших программ
- о* медленная работа компьютера *о* невозможность загрузки ОС
- о* исчезновение файлов и каталогов или искажение их содержимого *о* изменение размеров файлов и их времени модификации *о* уменьшение размера оперативной памяти
- о* непредусмотренные сообщения, изображения и звуковые сигналы *о* частые сбои и зависания компьютера и др.

### **2.Задания для самостоятельной работы.**

Задание. Обновить через Интернет антивирусную программу, установленную на Вашем компьютере. Выполнить проверку папки «Мои документы» на вирусы. Дать характеристику этой программы.

## Практическая работа № 17

**Тема: Информационная деятельность человека.**

**Цели занятия:** научиться пользоваться образовательными информационными ресурсами, искать нужную информацию с их помощью;

**Оборудование, программное обеспечение:** ПК, ОС Windows.

### Методические рекомендации

Понятие «информационного ресурса общества» (ИРО) является одним из ключевых понятий социальной информатики. Широкое использование этого понятия началось после выхода в 1984 году книги Громова Г.Р. «Национальные информационные ресурсы: проблемы промышленной эксплуатации».

«Информационный ресурс – это знания, представленные в проектной форме», – такое краткое и недостаточно строгое определение было предложено профессором Ю.М. Каныгиным.

Таким образом, информационные ресурсы – это знания, подготовленные для целесообразного социального использования.

Понятие ИРО, накопленных в обществе знаний, может быть рассмотрено в узком и широком смысле слова.

ИРО в узком смысле слова – это знания, уже готовые для целесообразного социального использования, то есть отчужденные от носителей и материализованные знания.

ИРО в широком смысле слова включают в себя все отчужденные от носителей и включенные в информационный обмен знания, существующие как в устной, так и в материализованной форме.

Понятие *ресурс* определяется в Словаре русского языка С.И. Ожегова как запас, источник чего-нибудь.

Что же касается *информационных ресурсов*, то это понятие является сравнительно новым. Оно еще только начинает входить в жизнь современного общества, хотя в последние годы становится все более употребительным не только в научной литературе, но и в общественно-политической деятельности. Причиной этого, безусловно, является глобальная информатизация общества, в котором все больше начинает осознаваться особо важная роль информации и научных знаний.

Для *классификации информационных ресурсов* могут быть использованы следующие их наиболее важные параметры:

- тематика хранящейся в них информации;
- форма собственности – государственная (федеральная, субъекта федерации, муниципальная), общественных организаций, акционерная, частная;
- доступность информации – открытая, закрытая, конфиденциальная;
- принадлежность к определенной информационной системе – библиотечной, – архивной, научно-технической;
- источник информации – официальная информация, публикации в СМИ, статистическая отчетность, результаты социологических исследований;
- назначение и характер использования информации – массовое региональное, ведомственное;
- форма представления информации – текстовая, цифровая, графическая, мультимедийная;
- вид носителя информации – бумажный, электронный.

Под **образовательными информационными** ресурсами мы будем понимать текстовую, графическую и мультимедийную информацию, а также исполняемые программы (дистрибутивы), то есть электронные ресурсы, созданные специально для использования в процессе обучения на определенной ступени образования и для определенной предметной области.

При работе с образовательными ресурсами появляются такие понятия, как *субъект* и *объект* этих ресурсов. Классификацию субъектов информационной деятельности произведем следующим образом:

- субъект, создающий объекты (все пользователи образовательной системы-преподаватель, студент);
- субъект, использующий объекты (все пользователи образовательной системы);
- субъект, администрирующий объекты, то есть обеспечивающий среду работы с объектами других субъектов (администраторы сети);
- субъект, контролирующий использование объектов субъектами (инженеры).

К образовательным электронным ресурсам можно отнести:

- учебные материалы (электронные учебники, учебные пособия, рефераты, дипломы),
- учебно-методические материалы (электронные методики, учебные программы),
- научно-методические (диссертации, кандидатские работы),
- дополнительные текстовые и иллюстративные материалы (лабораторные работы, лекции),
- системы тестирования (тесты– электронная проверка знаний),
- электронные полнотекстовые библиотеки;
- электронные периодические издания сферы образования;
- электронные оглавления и аннотации статей периодических изданий сферы образования,
- электронные архивы выпусков.

### Содержание работы:

#### ЗАДАНИЕ 1

1. Загрузите Интернет.
2. В строке поиска введите фразу «каталог образовательных ресурсов».
3. Перечислите, какие разделы включают в себя образовательные ресурсы сети Интернет.
4. Охарактеризуйте любые три

#### Название Характеристика

#### ЗАДАНИЕ 2

С помощью Универсального справочника-энциклопедии найдите ответы на следующие вопросы:

##### Вопрос

##### Ответ

- 1) *укажите время утверждения григорианского календаря*
- 2) *каков диаметр пылинки*
- 3) *укажите смертельный уровень звука*
- 4) *какова температура кипения железа*
- 5) *какова температура плавления йода*
- 6) *укажите скорость обращения Земли вокруг Солнца*
- 7) *какова масса Земли*

- 8) *какая гора в Австралии является самой высокой*
- 9) *дайте характеристику народа кампа*
- 10) *укажите годы правления Ивана III*
- 11) *укажите годы правления Екатерины II*
- 12) *укажите годы правления Ивана IV*
- 13) *укажите годы правления Хрущева Н.С.*
- 14) *в каком году был изобретен первый деревянный велосипед*

**ЗАДАНИЕ 3. Ответьте на вопросы**

1. Что Вы понимаете под информационными ресурсами?
2. Перечислите параметры для классификации информационных ресурсов.
3. Что понимают под образовательными информационными ресурсами?
4. Что можно отнести к образовательным электронным ресурсам?

**ЗАДАНИЕ 4. Сделайте вывод о проделанной работе**

## Практическая работа № 18

### Тема: «Построение блок-схемы алгоритма».

**Цель работы:** научиться составлять алгоритмы графическим способом (блок-схем)

#### *Методические указания*

Этапы решения задачи на ЭВМ. Работа по решению любой задачи с использованием компьютера включает в себя шесть этапов

1. постановка задачи
2. формализация задачи
3. построение алгоритма
4. составление программы на языке программирования
5. отладка и тестирование программы
6. проведение расчетов и анализ полученных результатов

Часто эту последовательность называют технологической цепочкой решения задачи на ЭВМ.

На этапе постановки задачи следует четко определить, что дано и что требуется найти. Важно описать полный набор исходных данных, необходимых для решения задачи. На этапе формализации чаще всего задача переводится на язык математических формул, уравнений и отношений. Если решение задачи требует математического описания какого-то реального объекта, явления или процесса, то ее формализация равносильна получению соответствующей математической модели.

Третий этап — это построение алгоритма. Опытные программисты часто сразу пишут программы на определенном языке, не прибегая к каким-либо специальным средствам описания алгоритмов (блок-схемам, псевдокодам), однако в учебных целях полезно сначала использовать эти средства, а затем переводить полученный алгоритм на язык программирования.

Алгоритм — это последовательность команд управления каким-либо исполнителем. В школьном курсе информатики с понятием алгоритма и

методами построения алгоритмов ученики знакомятся на примерах учебных исполнителей: Робота, Черепахи, Чертежника и др. Эти исполнители ничего не вычисляют. Они создают рисунки на экране, перемещаются в лабиринтах, перетаскивают предметы с места на место.

Данные и величины. Совокупность величин, с которыми работает компьютер, принято называть данными. По отношению к программе различают исходные, окончательные (результаты) и промежуточные данные, которые получают в процессе вычислений.

В каждом языке программирования существует своя концепция и своя система типов данных. Однако в любой язык входит минимально необходимый набор основных типов данных: целые вещественные, логические и символьные. С типом величины связаны три ее свойства: множество допустимых значений, множество допустимых операций, форма внутреннего представления.

Блок-схема — графическое представление алгоритма. Она состоит из функциональных блоков, которые выполняют различные назначения (ввод/вывод, начало/конец, вызов функции и т.д.).

Существует несколько основных видов блоков, которые нетрудно запомнить:



**Пример №1:** Рассчитать площадь и периметр прямоугольника по двум известным сторонам.

Данная задача не должна представлять особой трудности, так как построена она на хорошо известных всем нам формулах расчета площади и периметра прямоугольника, поэтому заикливаться на выведении этих формул мы не будем.

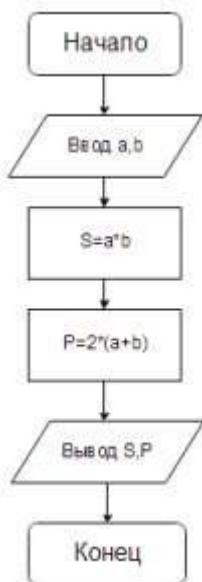
*Составим алгоритм решения подобных задач:*

1. Прочитать задачу.
2. Выписать известные и неизвестные нам переменные в «дано». (В задаче №1 к известным переменным относятся стороны:  $a, b$ ; к неизвестным — площадь  $S$  и периметр  $P$ )
3. Вспомнить либо составить необходимые формулы. (У нас:  $S=a*b$ ;  $P=2*(a+b)$ )
4. Составить блок-схему.
5. Записать решение на языке программирования Pascal. Запишем условие в более кратком виде.

Дано:  $a, b$

Найти:  $S, P$

**Блок-схема:**



Словесное описание алгоритма:

Структура программы, решающей данную задачу, тоже проста:

1. Описание переменных;
2. Ввод значений сторон прямоугольника;
3. Расчет площади прямоугольника;



4. Расчет периметра прямоугольника;
5. Вывод значений площади и периметра;
6. Конец.

### ЗАДАНИЕ

Составить словесно-формульный алгоритм и блок-схему для следующих задач:

1. Вычислить периметр и площадь прямоугольного треугольника по длинам двух катетов:  $a$  и  $b$ ;
2. Вычислить длину окружности и площадь круга с заданным радиусом  $R$ ;
3. Вычислить расстояние между двумя точками с заданными координатами  $(x_1, y_1)$  и  $(x_2, y_2)$

## Практическая работа №19

### Тема «Составление и отладка линейных и разветвляющихся программ».

#### Цель:

- развить навыки составления структурных схем линейных и разветвляющихся алгоритмов;
- развить навыки составления программного кода по структурной схеме;
- развить навыки тестирования программных кодов.

#### Обеспечение практической работы:

- тетрадь для выполнения практических работ;
- канцелярские принадлежности;
- компилятор Паскаль ABC;
- калькулятор;
- задания для выполнения работы;
- материал лекций.

#### В результате выполнения данной работы студент должен уметь:

- составлять структурные схемы линейных и разветвляющихся алгоритмов;
- составлять программы по структурной схеме;
- тестировать программные коды.

#### Теоретический материал

#### Пример 1. Составить программу для решения задачи.

Заданы длины трех сторон треугольника  $a$ ,  $b$ ,  $c$ . Вычислить периметр и площадь треугольника. Значения  $a$ ,  $b$ ,  $c$  ввести с клавиатуры.

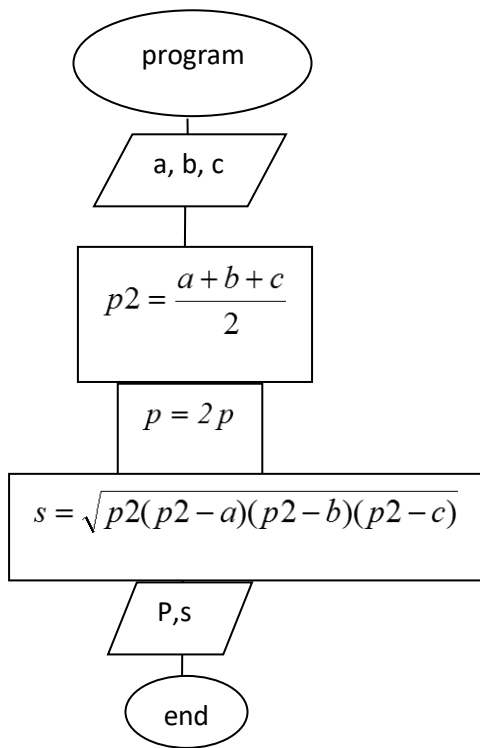
*Решение.*

##### 1. Построение математической модели.

Решение задачи можно разделить на несколько этапов:

1. Ввод значений  $a$ ,  $b$  и  $c$ .
2. Вычисление полупериметра по формуле  $p = \frac{a + b + c}{2}$ .
3. Вычисление площади треугольника по формуле Герона  $s = \sqrt{p(p-a)(p-b)(p-c)}$ .
4. Вывод площади треугольника  $s$  и периметра  $P$ .

##### 2. Составление блок – схемы



### 3. Программа с комментариями.

```

program four;
{Описание переменных вещественного типа: a, b, c –
стороны треугольника, s – площадь, p – полупериметр}
var
  a, b, c, s, p2, p: real;
begin
  {Ввод исходных данных}
  write ('a='); readln (a);
  write ('b='); readln (b);
  write ('c='); readln (c);
  {Вычисление полупериметра треугольника}
  p2:=(a+b+c)/2;
  {Вычисление периметра треугольника}
  p=2*p2
  {Вычисление площади треугольника}
  s:=sqrt(p2*(p2-a)*(p2-b)*(p2-c));
  {Вывод периметра и площади в экспоненциальной
  форме}
  writeln ('периметр треугольника', p);

```

```

writeln ('площадь треугольника', s);
end.

```

### 4. Тестирование.

Пусть a=5, b=3, c=4.

Тогда  $p2=(5+3+4)/2=6$ ;  $p=2p=12$ ;  $s = \sqrt{6(6-5)(6-3)(6-4)} = \sqrt{6 \cdot 1 \cdot 3 \cdot 2} = 6$

**Пример 2.** Разработать программу, которая вычисляет значение функции, заданной следующим образом:

$$y = \begin{cases} |x| & \text{при } |x| \leq 1; \\ x^2 & \text{при } 1 < |x| \leq 2; \\ 4 & \text{иначе.} \end{cases}$$

Решение.

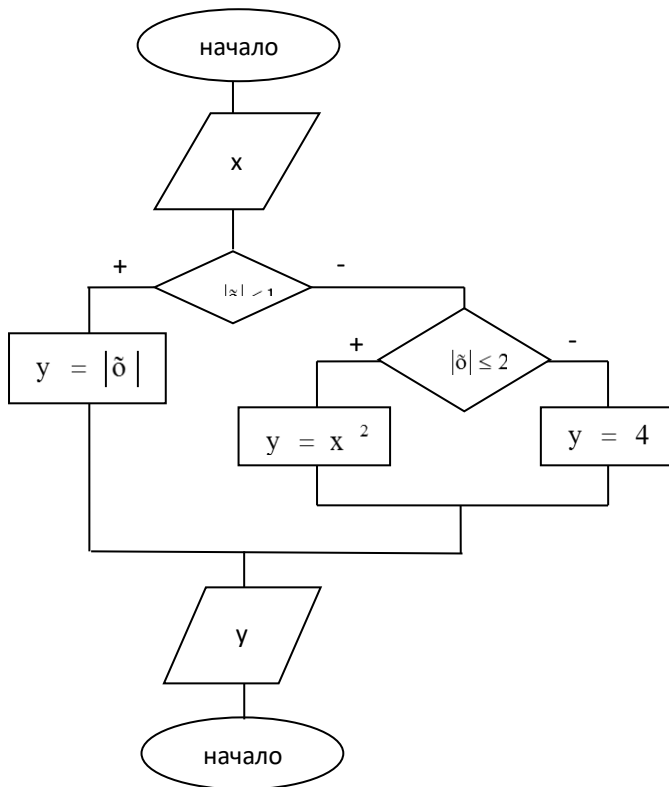
1. Построение математической модели.

Исходные данные: x.

Результат: y.

Вычисления: Если  $|\tilde{0}| \leq 1$ , то  $y = |\tilde{0}|$ , иначе если  $|\tilde{0}| \leq 2$ , то  $y = x^2$ , иначе  $y=4$ .

2. Составление блок – схемы.



### 3. Программа с комментариями

```

Program ex2;
Var x,y: real;
Begin
  WriteLn('введите значение аргумента. ');
  ReadLn(x);
  if abs(x) <= 1 then y := abs(x)
  { первый отрезок }
  else
    if (abs(x) > 1) and
    (abs(x) <= 2) then y := sqr(x)
    { второй отрезок }
  else
    y := 4;
  end;
  WriteLn('результат: ', y);
End.
  
```

Пример 8. По заданному номеру месяца  $m$  вывести на печать название времени года.  
Решение.

Для решения данной задачи необходимо проверить выполнение четырех условий. Если заданное число  $m$  равно 12, 1 или 2, то это зима, если  $m$  попадает в диапазон от 3 до 5, то – весна. Лето определяется принадлежностью числа  $m$  диапазону от 6 до 8, и соответственно, при равенстве переменной  $m$  9, 10 или 11 – это осень. Понятно, что область возможных значений переменной  $m$  находится в диапазоне от 1 до 12.

```

Var m: byte;
begin
  write ('введите номер месяца m=');
  readln (m);
  if (m >= 1) and (m <= 12) then      { проверка области допустимых значений переменной m. }
    case m of
      { в зависимости от значения m на печать }
      { выводится название времени года. }
      12, 1, 2: writeln ('ЗИМА');
      3 .. 5: writeln ('ВЕСНА');
      6 .. 8: writeln ('ЛЕТО');
      9 .. 11: writeln ('ОСЕНЬ');
    end;
end
  
```

*else*

*writeln* ('ОШИБКА ПРИ ВВОДЕ!!!'); { если значение переменной *m* выходит за пределы области допустимых значений, то выдается сообщение об ошибке. }

*end.*

### **Задание для практической работы**

#### **Задача №1.**

Согласно варианту выполнить следующее задание:

Напишите программу для расчета по двум формулам.

**Вариант 1**

$$z_1 = \frac{2tg^2(5\pi + 2\alpha)}{|6x^3 + 4|}$$

$$z_2 = \frac{1}{4} - \frac{1}{4} e^{\sin(b-8\alpha)}$$

**Вариант 2**

$$z_1 = \cos \alpha + \ln |\sqrt{3x}| + \cos 3\alpha + \sin 3\alpha$$

$$z_2 = 2\sqrt{2} \cos \alpha \cdot \sin\left(\frac{\pi}{4} + 2\alpha\right)$$

**Вариант 3**

$$z_1 = \frac{\log_5(\sin 2\alpha + \sin 5\alpha) - \sin 3\alpha}{tg\alpha + 1 - 2\sin^2 2\alpha}$$

$$z_2 = 7^{2\sin^4 \alpha}$$

**Вариант 4**

$$z_1 = \frac{\sin^2 2\alpha + |\sin 5\alpha - \sin 3\alpha|}{\cos \alpha - 5}$$

$$z_2 = tg(3 \ln \alpha) + e^{6x}$$

**Вариант 5**

$$z_1 = 1\sqrt{5\alpha^2 - \sin(\alpha - 6)} - \frac{1}{4} \sin^2 2\alpha + \cos 2\alpha$$

$$z_2 = \cos^2(\ln a) + \cos^2(a^2 + 2b + 7)$$

**Вариант 6**

$$z_1 = \log_3 \alpha + tg(\alpha^{-3}) + \cos 7\alpha$$

$$z_2 = 23ctg(\alpha^4 - 3\alpha) \cdot \cos \frac{5}{2} \alpha \cdot \sqrt{4\alpha + 1}$$

**Вариант 7**

$$z_1 = \cos^2\left(\frac{3}{8}\pi - \frac{\alpha}{4}\right) - \log_5(\alpha^2 - 8)$$

$$z_2 = \frac{\sqrt{2}}{2} \sin \frac{\alpha}{2} + \sqrt{|\alpha^{-4} + 4\alpha^2|}$$

**Вариант 8**

$$z_1 = \cos^4 x + \lg^2 y + \frac{1}{4} \cos^2(a^2 + 5) - 1$$

$$z_2 = \frac{\sin^4(y-x)}{x+5} - \sin(\alpha^{-2} + 3\alpha^2 - 7\alpha + 8)$$

**Вариант 9**

$$z_1 = \frac{(\cos \alpha - \cos \beta)^2}{e^{3x}} - \operatorname{tg}(|\beta^3 + \beta - 2\beta^2|)$$

$$z_2 = -4 \sin^2 \frac{\alpha - \beta}{2} \cdot \cos(\alpha + \beta) + \log_2(\beta^2 - \alpha^2)$$

### Вариант 10

$$z_1 = \frac{\sin(\frac{\pi}{2} + 3\alpha)}{1 + \operatorname{ctg}(\beta + \pi) - \sin(3\alpha - \pi)}$$

$$z_2 = \operatorname{ctg}(\frac{5}{4}\pi + \frac{3}{2}\alpha) - \log_3(\alpha + 8\pi)$$

### Вариант 11

$$z_1 = \frac{1 - 2 \sin^2 \alpha}{1 - \sqrt{\alpha^3 + \beta^3} + \sin 2\alpha}$$

$$z_2 = \frac{1 - \operatorname{tg} \alpha}{1 + \operatorname{tg} \alpha \sin \sqrt{\beta^3}}$$

### Вариант 12

$$z_1 = \frac{\sin 4\alpha}{1 + \cos 4\alpha} \cdot \frac{e^{-3} + \beta^2}{1 + \cos 2\alpha}$$

$$z_2 = \operatorname{ctg}(\frac{3}{2}\pi - \alpha) - \log_7 \sqrt{|\alpha^3 + \alpha^2 - \alpha^{-1}|}$$

### Вариант 13

$$z_1 = \frac{\operatorname{tg} \alpha + \cos(2\beta - \alpha)}{e^{\sqrt{|\beta^3 - \alpha|}} - \sin(2\beta - \alpha)}$$

$$z_2 = \frac{1 + \sin 2\beta}{\cos 2\beta} + \log_8 \left| \sqrt{6\beta^{-2} + 8\alpha^2 - \beta} \right|$$

### Вариант 14

$$z_1 = \frac{e^{x^2+5}}{\cos \alpha - \sin \alpha} + \ln(|8\pi - 16\alpha^2 - 9\beta^{-3}|)$$

$$z_2 = \operatorname{tg} 2\alpha + \sin(|2\alpha^2 - 8\beta^3|)$$

### Вариант 15

$$z_1 = \frac{\sqrt{2b + 2\sqrt{\ln(b^2 - 4|\alpha|)}}}{\sqrt{b^2 - 4} + b + 2}$$

$$z_2 = \frac{\operatorname{tg}(e^{7x} + 6)}{\sqrt{b+2}} - \sin \beta^2$$

### Вариант 16

$$z_1 = \frac{\log_4(x^2 + 2x) - 3\cos(\pi + \alpha) + (x+1)\sqrt{x^2 - 9}}{x^2 + (x-1)\sqrt{x^2 - 9}}$$

$$z_2 = \sqrt{\frac{|x+3|}{x - 3\operatorname{arctg}(\beta^2)}}$$

### Вариант 17

$$z_1 = \frac{\sqrt{(3m+2)^2 - 24\operatorname{tg}(e^x + 4)}}{3\log_5\left|\sqrt{m} - \frac{2}{\sqrt{m}}\pi\right|}$$

$$z_2 = -\cos(e^{\alpha^2-3} + 2)$$

### Вариант 18

$$z_1 = \left( \frac{a+2}{\sqrt{2a + \ln(|\beta + x^3|)}} + \frac{\operatorname{tg}(e^{3x} + 5)}{a - \sqrt{2a}} \right) \cdot \frac{\sqrt{a} - \sqrt{2}}{a+2}$$

$$z_2 = \frac{1}{7\sqrt{a} + \sqrt{2}} - \log_5^2(b^{-2})$$

### Вариант 19

$$z_1 = \left( \frac{1 + e^{a+a^2}}{2a + a^2 - \sqrt{\alpha^2 + \beta^4 - \pi}} + \ln 2 - \frac{1 - a + a^2}{2a - a^2} \right)^{-1} (5 - 2a^2)$$

$$z_2 = \frac{4 - a^2}{2} \sqrt[4]{\alpha^3 + \operatorname{tg}\beta^2 + \beta^{-2}}$$

### Вариант 20

$$z_1 = \frac{(m-1)\sqrt{m} - \sqrt{n}}{\sqrt{m^3 n} + \operatorname{tg}(nm + m^2) - \log_5 m}$$

$$z_2 = \frac{(\sqrt{m} - \sqrt{n})^2}{m} \left( \cos^2 \alpha + \sin^2(\alpha^2 + \beta^2) \right)$$

### Вариант 21

$$z_1 = \frac{\log_5 \alpha + \operatorname{tg}(\alpha^{-1}) + \cos 7\alpha}{5x + 9}$$

$$z_2 = 2\operatorname{tg}(6 - 3\alpha) \cdot \cos|\alpha \cdot \sqrt{4\alpha + 1}|$$

### Вариант 22

$$z_1 = \cos^2\left(6 - \frac{\alpha}{4}\right) - \log_8(\alpha^2 - 8x^{-5})$$

$$z_2 = 7\cos\frac{\alpha}{2} + \sqrt{|\alpha^{-4} + 4\operatorname{tg}\alpha^2|}$$

### Вариант 23



$$z_1 = \cos^6 x + \ln^2 y + 5 \cos^2 (a^2 + 5)^{-1}$$

$$z_2 = \frac{\sin^4 (y - x)}{e^x + 5} - \sin(3a^2 - 7a + 8)$$

$$z_1 = \frac{(\cos^2 x - \sqrt{\cos y} + 5)^{-3}}{e^{3x}} - \operatorname{tg}(e^3 - 6x|x + x))$$

$$z_2 = -4 \sin^2 \frac{x^3 - y}{2 + 5x} + \log_{\frac{2}{3}}(x^2 - y^2)$$

### Задача №2.

Согласно варианту, составить программу решения поставленной задачи.

Вариант	№ задач	Вариант	№ задач	Вариант	№ задач
1	1, 31, 61, 88	11	11, 39, 77, 92	21	21, 42, 79, 94
2	2, 33, 60, 94	12	12, 41, 82, 93	22	22, 53, 63, 89
3	3, 34, 68, 91	13	13, 48, 85, 96	23	23, 46, 84, 98
4	4, 32, 71, 96	14	14, 50, 62, 89	24	24, 49, 87, 93
5	5, 36, 69, 92	15	15, 43, 73, 90	25	25, 56, 70, 90
6	6, 38, 76, 97	16	16, 52, 83, 91	26	26, 59, 64, 95
7	7, 35, 81, 90	17	17, 58, 65, 94	27	27, 44, 63, 98
8	8, 37, 66, 97	18	18, 54, 80, 88	28	28, 51, 66, 95
9	9, 40, 86, 93	19	19, 45, 67, 89	29	29, 47, 75, 92
10	10, 50, 78, 88	20	20, 57, 74, 91	30	30, 55, 71, 95

1. Даны действительные числа  $x$  и  $y$ . Получить  $\frac{|x| - |y|}{1 + |xy|}$ .

2. Дана длина ребра куба. Найти объем куба и площадь его боковой поверхности.

3. Даны два действительных положительных числа. Найти среднее арифметическое этих чисел и среднее геометрическое этих чисел.

4. Даны два действительных числа. Найти среднее арифметическое этих чисел и среднее геометрическое их модулей.

5. Даны катеты прямоугольного треугольника. Найти его гипотенузу и площадь.

6. Смешано  $V_1$  литров воды температуры  $t_1$  с  $V_2$  литрами воды температуры  $t_2$ . Найти объем и температуру образовавшейся смеси.

7. Определить периметр правильного  $n$ -угольника, описанного около окружности радиуса  $r$ .

8. Три сопротивления  $R_1$ ,  $R_2$ ,  $R_3$  соединены параллельно. Найти сопротивление соединения.

9. Определить время падения камня на поверхность земли с высоты  $h$ .

10. Даны  $x$ ,  $y$ ,  $z$ . Вычислить  $a$ ,  $b$ , если

$$a = \frac{\sqrt{|x-1|} - \sqrt[3]{|y|}}{1 + \frac{x^2}{2} - \frac{y^2}{4}}, \quad b = x(\operatorname{arctg} z + e^{-(x+3)});$$

11. Даны  $x$ ,  $y$ ,  $z$ . Вычислить  $a$ ,  $b$ , если

$$a = \frac{3 + e^{-x}}{1 + x^2|y - \operatorname{tg}z|}, \quad b = 1 + |y - x| + \frac{(y - x)^2}{2} + \frac{|y - x|^3}{3};$$

12. Даны  $x, y, z$ . Вычислить  $a, b$ , если

$$a = (1 + y) \frac{x + y / (x^2 + 4)}{e^{-x-2} + |1(x^2 + 4)|}, \quad b = \frac{1 + \cos(y - 2)}{x^4 / 2 + \sin^2 z};$$

13. Даны  $x, y, z$ . Вычислить  $a, b$ , если

$$a = y + \frac{x}{y^2 + \left| \frac{x^2}{y + x^3 / 3} \right|}, \quad b = \left( 1 + \operatorname{tg}^2 \frac{z}{2} \right);$$

14. Даны  $x, y, z$ . Вычислить  $a, b$ , если

$$a = \frac{1 + \sin^2(x + y)}{2 + |x - 2x / (1 + x^2 y^2)|} + x, \quad b = \cos^2 \left( \operatorname{arctg} \frac{1}{z} \right);$$

15. Дана сторона равностороннего треугольника. Найти площадь этого треугольника.

16. Даны гипотенуза и катет прямоугольного треугольника. Найти второй катет и радиус вписанной окружности.

17. Известна длина окружности. Найти площадь круга, ограниченного этой окружностью.

18. Найти сумму членов арифметической прогрессии  $a, a + d, \dots, a + (n - 1)d$ .

19. Найти площадь равнобокой трапеции с основаниями  $a$  и  $b$  и угол  $\alpha$  при большем основании  $a$ .

20. Вычислить расстояние между двумя точками с координатами  $x_1, y_1, x_2, y_2$ .

21. Треугольник задан координатами своих вершин. Найти периметр и площадь треугольника.

22. Дано действительное число  $x$ . Не пользуясь никакими другими арифметическими операциями, кроме умножения, сложения и вычитания, вычислить  $2x^4 - 3x^3 + 4x^2 - 5x + 6$ .

23. Даны действительные числа  $x, y$ . Не пользуясь никакими операциями, кроме умножения, сложения и вычитания, вычислить  $3x^2 y^2 - 2xy^2 - 7x^2 y - 4y + 15xy + 2x^2 - 3x + 10y + 6$ .

### Вопросы для самоконтроля

1. Какой диапазон имеет тип данных Byte?
2. Какие типы данных описывает вещественный тип?
3. В каком разделе программы описываются константы?
4. В каком разделе программы описываются переменные?
5. Как называется оператор состоящий из нескольких операторов?
6. Приведите пример, в котором используется оператор выбора Case.
7. Какой результат выполнения программы будет получен при  $a=4$  и  $b=5$  для программного кода:

```

If ((a+5)>15) or (a-b)<0 then
begin
a:=a-8;
if a*b>8 then
c:=a/(a+5)

```

```
else
c:=(a+7)/a;
end;
else
c:=a+b;
```

4. Определите константу выражения  $\text{not}((x > 5) \text{and} (a + x < 8))$  при следующих наборах переменных:

а)  $a=4$ ;  $x=-8$ .

б)  $a=14$ ;  $x=-18$ .

в)  $a=2$ ;  $x=-8$ .

5. Определите константу выражения  $\text{not}((y/c \geq 0) \text{or} (a * x < 0) \text{and} y + x = 6)$  при следующих наборах переменных:

а)  $a=-4$ ;  $x=6$ ;  $y=0$ ;  $c=9$ .

б)  $a=4$ ;  $x=0$ ;  $y=-3$ ;  $c=19$ .

в)  $a=0$ ;  $x=16$ ;  $y=0$ ;  $c=3$ .



## Практическая работа №20

### Тема «Составление и отладка циклических программ».

#### Цель:

- развить навыки составления структурных схем циклических алгоритмов;
- развить навыки составления программного кода по структурной схеме;
- развить навыки тестирования программных кодов.

#### Обеспечение практической работы:

- тетрадь для выполнения практических работ;
- канцелярские принадлежности;
- компилятор Паскаль ABC;
- калькулятор;
- задания для выполнения работы;
- материал лекций.

#### В результате выполнения данной работы студент должен уметь:

- составлять структурные схемы циклических алгоритмов;
- составлять программы по структурной схеме;
- тестировать программные коды.

#### Методические указания

**Пример 1.** Составить таблицу значений функции  $y = e^{\sin(x)} \cos(x)$  на отрезке  $[0; \pi]$  с шагом 0,1.

#### Решение.

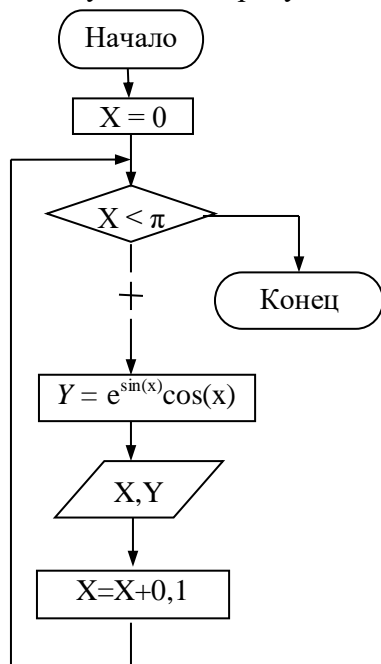
1. Построение математической модели.

*Входные данные:* начальное значение аргумента - 0, конечное значение аргумента -  $\pi$ , шаг изменения аргумента - 0,1.

*Выходные данные:* множество значений аргумента  $X$  и соответствующее им множество значений функции  $Y$ .

2. Составление структурной схемы.

В условии задачи количество повторений цикла явно не задано, но известно, как изменяется параметр цикла  $X$  и каковы его начальное и конечное значения. Поэтому решим эту задачу, используя цикл с предусловием, предварительно построив структурную схему:



### 3. Программный код.

```
program Example_11;  
const pi=3.14;  
var x,y:real;  
begin  
x:=0; { Присвоение начального значения параметру цикла. }  
{ Перед каждым выполнением тела цикла проверяется условие  $x \leq \pi$ , если оно  
истинно, то тело цикла выполняется, иначе управление передается  
следующему за циклом оператору }  
while x<=pi do  
  begin  
    {Расчет значения функции для соответствующего значения аргумента  
X} y:=exp(sin(x))*cos(x);  
    writeln (x:4:2, ' ' \y:8:5); { Печать значений функции и аргумента. }  
    x:=x+0.1; {Вычисление нового значения аргумента.}  
  end; {Конец цикла while} .  
readln;  
end.
```

2 способ.

Теперь решим задачу другим способом, используя циклический оператор **for ... do**, предварительно определив количество повторений тела цикла  $n$ .

#### 1. Построение математической модели.

Если параметр цикла  $x$  принимает значения в диапазоне от  $x_n$  до  $x_k$ , изменяясь с шагом  $dx$ , то количество повторений тела цикла можно определить по формуле:

$$n = \frac{x_k - x_n}{dx} + 1$$

округлив результат деления до целого числа.

#### 2. Программный код.

```
program Example_12;  
var x,y:real,-n,i:integer;  
begin  
x:=0; { Присвоение начального значения аргументу. }  
n:=round((pi-0)/0.1)+1; {Расчет количества повторений цикла. Функция round округляет результат деления  
до целого}  
for i:-1 to n do { Тело цикла выполнится ровно n раз }  
  begin  
    Y:=exp (sin(x))*cos(x) ; {Расчет значения функции для соответствующего значения аргумента  
X. }  
    Writeln (x:4:2, ' ',y:8:4); {Печать значений функции и аргумента. }  
    x:=x+0.1; {Вычисление нового значения аргумента.}  
  end; { Конец цикла for ... do }  
end.
```

**Пример 2.** Дано натуральное число  $n$ . Подсчитайте количество цифр данного числа.

**Решение.**

#### 1. Построение математической модели.

Подсчет количества цифр начнем с последней цифры числа. Увеличим счетчик цифр на единицу. Число уменьшим в 10 раз (тем самым мы избавляемся от последней цифры числа). Далее с получившимся числом сделаем ту же последовательность действий и т.д., пока число не станет равным нулю.

## 2. Программный код.

**Program** Example\_2;

```

var m, n: Longint;
    k: Integer;           { счетчик
цифр}
begin
  writeln('Введите целое число'); { вводим целое число n}
  readln(n);
  m:=n;
  k:=0;
  while m<>0 do          { пока число m<>0 делать }
  begin
    Inc(k); {или k:=k+1;}
    m:=m Div 10;        {«уменьшаем» число на последнюю цифру}
  end;
  writeln ('В числе ', n, ' - ', k, ' цифр. '); { вывод количества цифр}
  readln;
end.

```

## 3. Тестирование программы.

Рассмотрим выполнение этой программы в пошаговом режиме для числа 65387.

В результате работы программы на экране появится предложение:

В числе 65387 – 5 цифр.

n	m	k
65387	65387	0
65387	6538	1
65387	653	2
65387	65	3
65387	6	4
65387	0	5

**Пример 3.** Вычислить факториал числа  $N$  ( $N! = 1 \cdot 2 \cdot \dots \cdot N$ ).

### 1. Построение математической модели.

*Входные данные:*  $N$  - целое число, факториал которого необходимо вычислить.

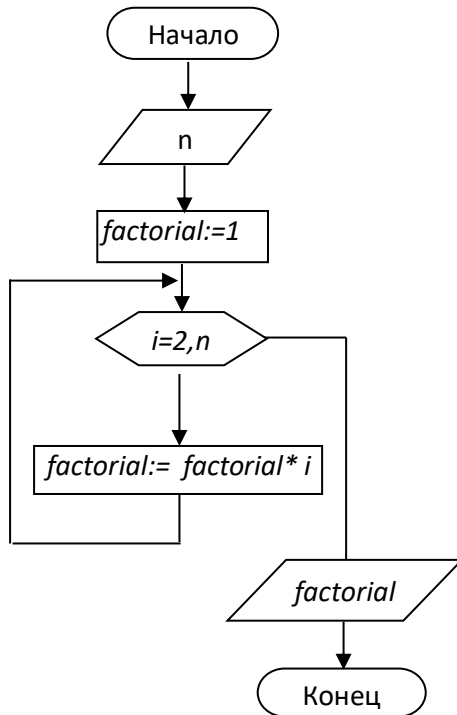
*Выходные данные:* factorial - значение факториала числа  $N$ .

*Промежуточные данные:*  $i$  - целочисленная переменная, принимающая значения от 2 до  $N$  с шагом 1, параметр цикла.

Итак, вводится число  $N$ . Переменной factorial, предназначенной для хранения значения произведения последовательности чисел, присваивается начальное значение, равное единице. Затем организуется цикл, параметром которого выступает переменная  $i$ . Если значение параметра цикла меньше или равно  $N$ , то выполняется оператор тела цикла, в котором из участка памяти с именем factorial считывается предыдущее значение произведения, умножается на текущее значение параметра цикла, а результат снова помещается в участок памяти с

именем factorial. Когда параметр  $i$  становится больше  $N$ , цикл заканчивается и на печать выводится значение переменной factorial, которая была вычислена в теле цикла.

2. Структурная схема:



3. Программный код.

**Program Example\_3;**

**var**

factorial, n, i: integer;

**begin**

**write**('n='); **readln** (n);

factorial:=1; {Стартовое значение.}

**for** i:=2 **to** n **do** factorial:=factorial\*i;

**writeln** (factorial);

**end.**

**Пример 4.** Вычислить  $a^n$  ( $n > 0$ ).

1. Построение математической модели.

**Входные данные:**  $a$  - вещественное число, которое необходимо возвести в целую положительную степень  $n$ .

**Выходные данные:**  $p$  (вещественное число) - результат возведения вещественного числа  $a$  в целую положительную степень  $n$ .

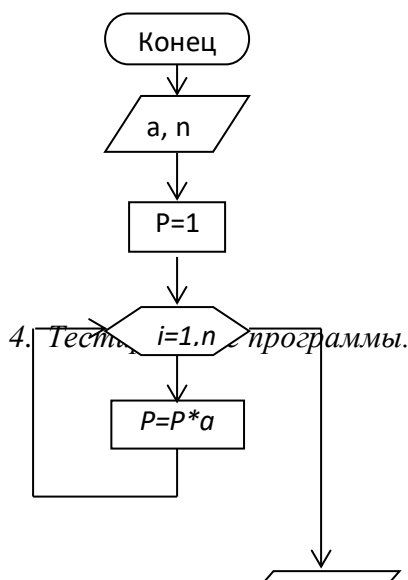
**Промежуточные данные:**  $i$  - целочисленная переменная, принимающая значения от 1 до  $n$  с шагом 1, параметр цикла.

Известно, что для того чтобы получить целую степень  $n$  числа  $a$ , нужно умножить его само на себя  $n$  раз. Результат будет храниться в участке памяти с именем  $p$ . При выполнении очередного цикла из этого участка предыдущее значение будет считываться, умножаться на основание степени  $a$  и снова записываться в участок памяти  $p$ . Цикл выполняется  $n$  раз.

В таблице отображен протокол выполнения алгоритма при возведении числа 2 в пятую степень:  $a = 2, n = 5$ .

2. Структурная схема.

3. Программный код.



4. Тесты программы.

**Program Example\_4;**

**var**

a,p:real;

i,n:word;

**begin**

**write**('Введите основание степени a=');

**readln**(a);

**write**('Введите показатель степени n=');

**readln**(n);

p:=1;

**for** i:=1 **to** n **do**

P:=P\*a;

**writeln** ('P=',P:1:3);

**readln**;

**end.**



**Пример 5.** Вычислить сумму натуральных четных чисел, не превышающих N.

1. Построение математической модели.

Входные данные: N - целое число.

Выходные данные: S - сумма четных чисел.

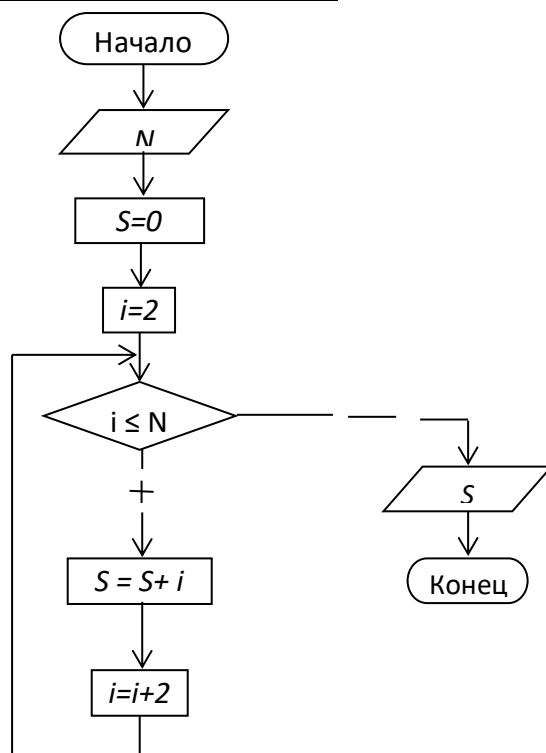
Промежуточные данные: i - переменная, принимающая значения от 2 до N с шагом 2, следовательно, также имеет целочисленное значение.

При сложении нескольких чисел необходимо накапливать результат в определенном участке памяти, каждый раз считывая оттуда предыдущее значение суммы и прибавляя к нему следующее слагаемое. Для выполнения первого оператора накапливания суммы из участка памяти необходимо взять такое число, которое не влияло бы на результат сложения. Иными словами, перед началом цикла переменной, предназначенной для накапливания суммы, необходимо присвоить значение нуль.

Так как параметр цикла i изменяется с шагом 2, в блок-схеме использован цикл с предусловием, который реализуется при составлении программы с помощью оператора while ... do.

i	1	2	3	4	5
P	2	4	8	16	32

2. Структурная схема.:



3. Программный код.

```
Program Example_5;
var n, i, S:word;
begin
  write( 'n=' ); readln (n);
  S:=0; i:=2;
  while i<=n do
    begin
      S:=S+i;
```

```

        i:=i+2;
    end;
writeln ('S=', S) ;
end.

```

Эту же задачу можно решить иначе, используя цикл for ... do:

```

Program Example_51;
var n, i, S: word;
begin
    write ('n=');
        if i mod 2 = 0 then           {Если остаток от деления параметра цикла на 2 равен 0, то
это число
        S:=S+i;                       четное. Следовательно, происходит накопление
суммы}
    writeln (' S=' , S);
end.

```

#### 4. Тестирование программы.

В таблице приведены результаты тестирования программы для  $n = 7$ . Несложно заметить, что при нечетных значениях параметра цикла значение переменной, предназначенной для накопления суммы, не изменяется.

<b>i</b>	1	2	3	4	5	6	7
<b>s</b>	0	2	2	6	6	12	12

**Пример 6.** Даны натуральные числа  $n$  и  $k$ . Составить программу вычисления выражения  $1^k + 2^k + \dots + n^k$ .

#### 1. Построение математической модели.

Для вычисления указанной суммы целесообразно организовать цикл с параметрами  $i$ , в котором, во-первых, вычислялось бы очередное значение  $y=i^k$  и, во-вторых, осуществлялось бы накопление суммы прибавлением полученного слагаемого к сумме всех предшествующих ( $s=s+y$ ).

#### 2. Программный код.

```

program Example_6;
var n, k, y, i, s, m: integer;
begin
    writeln ('Введите исходные данные n и k');
    readln (n, k);
    s:=0
    for i:=1 to n do
        begin
            y:=1
            for m:=1 to k do y:= y*I;           {нахождение степени k числа i}
            s:=s+y;
        end;
    writeln (' Ответ: ', s);
end.

```

Таким образом, для решения задачи потребовалось организовать два цикла, один из которых пришлось поместить внутрь другого. Такие конструкции называют вложенными циклами.

**Пример 7.** Модифицировать предыдущую программу так, чтобы она вычисляла сумму  $1^1 + 2^2 + \dots + n^n$ .

#### 1. Построение математической модели.

Данная задача отличается от предыдущей тем, что показатель степени очередного слагаемого совпадает со значением ее основания, следовательно, параметры внутреннего цикла (цикла,

в котором вычисляется очередное слагаемое) совпадают с параметрами внешнего цикла.

2. Программный код.

```
program Example_7;  
var n, y, i, s, m: integer;  
begin  
  writeln ('Введите начальное значение n` );  
  readln (n) ;  
  s:=0;  
  for i:=1 to n do  
    begin y:=1;  
      for m:=1 to i do y:=y*i;    {нахождение степени k числа i }  
      s:=s+y;  
    end;  
  writeln ('Ответ: ',s);  
end.
```

Внутренний и внешний циклы могут быть любыми из трех рассмотренных ранее видов: циклами с параметром, циклами с предусловием или циклами с постусловием. Правила организации как внешнего, так и внутреннего циклов такие же, как и для простого цикла каждого из этих видов. Но при использовании вложенных циклов необходимо соблюдать следующее условие: внутренний цикл должен полностью укладываться в циклическую часть внешнего цикла.

**Пример 8.** Старинная задача. Сколько можно купить быков, коров и телят, если плата за быка 10руб., за корову — 5руб., за теленка — полтинник (0,5руб.), если на 100 руб. надо купить 100 голов скота.

1. Построение математической модели.

Обозначим  $b$  — число быков;  $k$  — число коров;  $t$  — число телят.

После этого можно записать два уравнения:

$10b + 5k + 0,5t = 100$  и  $b + k + t = 100$ .

Преобразуем их:  $20b + 10k + t = 200$  и  $b + k + t = 100$ .

На 100 рублей можно купить:

- не более 10 быков, т. е.  $0 < b < 10$ ;
- не более 20 коров, т.е.  $0 < k < 20$ ;
- не более 200 телят, т.е.  $0 < t < 200$ .

2. Программный код.

```
program Example_8;  
var b, k, t: integer;  
begin  
  for b:=0 to 10 do  
    for k:=0 to 20 do  
      for t:=0 to 200 do  
        if (20*b+10*k+t=200) and (b+k+t=100) then writeln (`быков `, b, ` коров`, k,`  
  телят `, t);  
end.
```

Сколько раз будет проверяться условие в данной программе? Значение переменной  $b$  изменяется 11 раз (от 0 до 10), для каждого её значения переменная  $k$  изменяется 21 раз, а для каждого значения переменной  $k$  переменная  $t$  изменяется 201 раз. Таким образом, условие будет проверяться  $11 \cdot 21 \cdot 201$  раз. Но если известно число быков и коров, то число телят можно вычислить по формуле  $t = 100 - (b + k)$  и цикл по переменной  $t$  исключается.

```
program Example_81;  
var b, k, t: integer;  
begin  
  for b:=0 to 10 do  
    for k:=0 to 20 do  
      begin  
        t:=100-(b+k);  
        if (20*b+10*k+t=200) then writeln (`быков `, b, ` коров`, k,` телят `, t);  
      end;  
end;
```

*end.*

**Пример 9.** Дана непустая последовательность натуральных чисел, за которой следует 0. Составить программу поиска в данной непустой последовательности порядкового номера наименьшего элемента.

1. Построение математической модели.

Обозначим через  $x, i$  – очередной член последовательности и его номер;  $min, k$  - минимальный член последовательности и его номер. Считывание членов последовательности производится до тех пор, пока не будет наеден 0, то есть пока  $x \neq 0$ . Начальное значение минимума определяется значением первого члена последовательности. Очередное вводимое число требуется сравнивать с текущим значением минимума, и если текущее значение  $min$  окажется больше очередного члена последовательности, то его надо изменить.

2. Программный код.

```
program Example_9;  
var x, i, min, k: Longint;  
begin  
  writeln ('Введите первый член последовательности');  
  read (x); k:=1;  
  min:=x; i:=2;  
  while x<>0 do  
  begin  
    if x<min then  
    begin  
      min:=x;  
      k:=i  
    end;  
    writeln ('Введите ', i, ' элемент последовательности');  
    read (x);  
    inc (i);  
  end;  
writeln ('Номер минимального элемента - ', k);  
end.
```

**Пример 10.** Составить программу планирования закупки товара в магазине на сумму, не превышающую заданной величины.

1. Построение математической модели.

Обозначим через  $x, k$  соответственно цену и количество товара, через  $p$  – заданную предельную сумму, через  $s$  — общую стоимость покупки. Начальное значение общей стоимости покупки ( $s$ ) равно нулю. Значение предельной суммы донесся с клавиатуры. Необходимо повторять запрос цены и количества выбранного товара, вычислять его стоимость, суммировать ее с общей стоимостью и выводить результат на экран до тех пор, пока она не превысит предельную сумму  $p$ . В этом случае на экран надо вывести сообщение о превышении.

2. Программный код.

```
program Example_10;  
var x, k, p, s: integer;  
begin  
  writeln ('Предельная сумма - '); readln (p);  
  s:=0;  
  repeat  
    writeln ('Введите цену товара и его количество');  
    readln (x, k);  
    s:=s+x*k;
```

```

    writeln ('Стоимость покупки равна ', s);
until s>p;
writeln ('Суммарная стоимость покупки превысила предельную сумму');
end.

```

**Пример 11.** Дано натуральное число  $N$ . Определить  $K$ - количество делителей этого числа, не превышающих его. (Например,  $N=12$ , его делители 1, 2, 3, 4, 6,  $K=5$ ).

1. Построение математической модели.

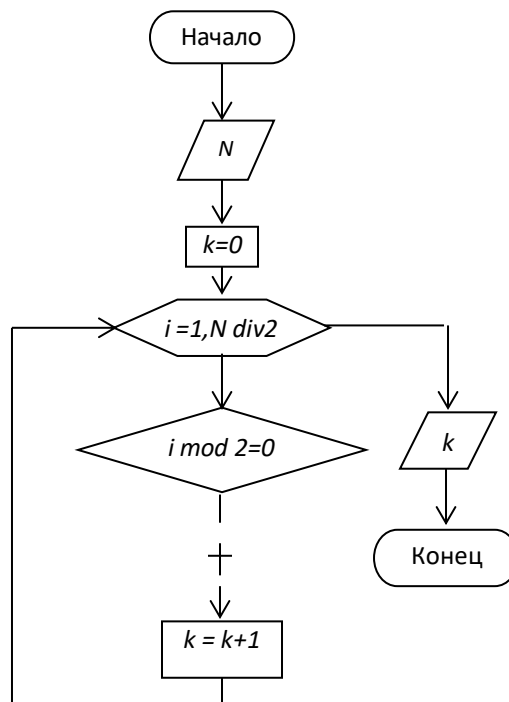
Входные данные:  $N$  - целое число.

Выходные данные: целое число  $K$  - количество делителей  $N$ .

Промежуточные данные:  $i$  - параметр цикла, возможные делители числа  $N$ .

В блок-схеме реализован следующий алгоритм: в переменную  $K$ , предназначенную для подсчета количества делителей заданного числа, помещается значение, которое не влияло бы на результат, то есть нуль. Далее организовывается цикл, в котором изменяющийся параметр  $i$  выполняет роль возможных делителей числа  $N$ . Если заданное число делится нацело на параметр цикла, это означает, что  $i$  является делителем  $N$ , и значение переменной  $K$  следует увеличить на единицу. Цикл необходимо повторить  $N/2$  раз.

2. Составление структурной схемы.



В таблице отображены результаты тестирования алгоритма при определении делителей числа  $N = 12$ .

<b>i</b>		1	2	3	4	5	6
<b>K</b>	0	1	2	3	4	4	5

2. Программный код.

```
var N, i, K:word;
```

```
begin
```

```
  write ('N='); readln (N);
```

```
  K:=0; { Присвоение начального значения. }
```

```

for i:=1 to N div 2 do
  if N mod i= 0 then { Если N делится нацело на i, то }
    k:=K+1; { увеличить счетчик на единицу. }
writeln (' K= ', K); end.

```

**Пример 12.** Дано натуральное число  $N$ . Определить, является ли оно простым. Натуральное число  $N$  называется простым, если оно делится нацело без остатка только на единицу и  $N$ . Число 13 - простое, так как делится только на 1 и 13,  $N= 12$  не является простым, так как делится на 1, 2, 3, 4, 6 и 12.

1. Построение математической модели.

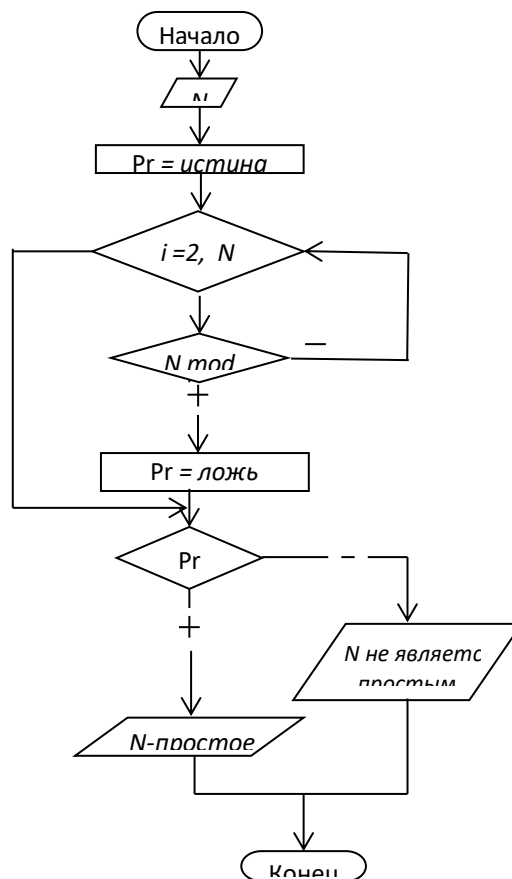
Входные данные:  $N$  - целое число.

Выходные данные: сообщение.

Промежуточные данные:  $i$  - параметр цикла, возможные делители числа  $N$ .

Алгоритм решения этой задачи заключается в том, что число  $N$  делится на параметр цикла  $i$ , изменяющийся в диапазоне от 2 до  $N/2$ . Если среди значений параметра не найдется ни одного числа, делящего заданное число нацело, то  $N$  - простое число, иначе оно таковым не является. Обратите внимание на то, что в алгоритме предусмотрено два выхода из цикла. Первый - естественный, при исчерпании всех значений параметра, а второй - досрочный. Нет смысла продолжать цикл, если будет найден хотя бы один делитель из указанной области изменения параметра.

2. Составление структурной схемы.



3. Программный код.

При составлении программы на языке Турбо Паскаль досрочный выход из цикла удобно выполнять при помощи оператора **break**:

```
var N, i: integer;
```

```
Pr :boolean;
```

```

begin
  write ('N='),
  readln (N);
  Pr:=true;      { Предположим, что число простое. }
  for i:=2 to N div 2 do
  if N mod i = 0 then {Если найдется хотя бы один
                        делитель, то}

      begin
        Pr:=false;   { число простым не является, и }
        break;      { досрочный выход из цикла. }
      end;
  if Pr then {Проверка значения логического
              параметра и вывод на печать
              соответствующего сообщения.}

      writeln ('Число ',N,' - простое')
    else
      writeln ('Число ',N,' простым не является');
  end.

```

**Пример 3.** Дано натуральное число  $N$ . Определить самую большую цифру и ее позицию в числе ( $N=573863$ , наибольшей является цифра 8, ее позиция - четвертая слева).

1. Построение математической модели.

Входные данные:  $N$  - целое число.

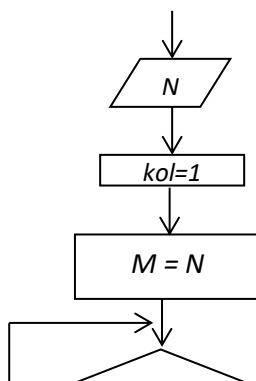
Выходные данные:  $max$  - значение наибольшей цифры в числе,  $pos$  - позиция этой цифры в числе.

Промежуточные Данные:  $i$  - параметр цикла,  $kol$  - количество цифр в числе,  $M$  - переменная для временного хранения значения  $N$ .

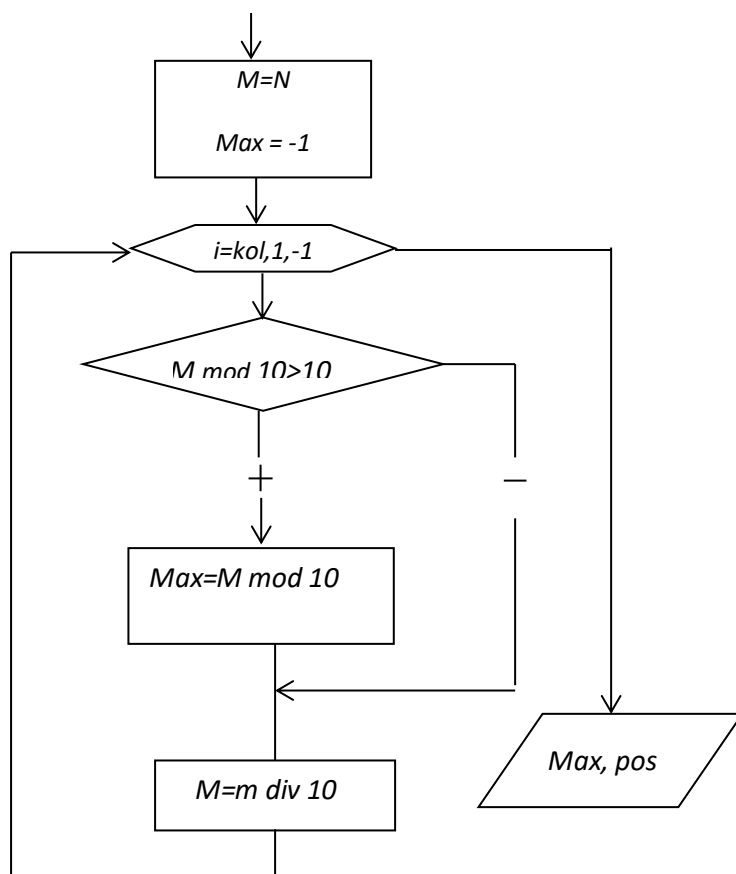
Разобьем решение этой задачи на два этапа. Вначале найдем количество цифр в заданном числе (рис. 1), а затем определим наибольшую цифру и ее позицию (рис. 2).

Для того чтобы подсчитать количество цифр в числе, необходимо определить, сколько раз заданное число можно разделить на десять нацело. Например, пусть  $N = 12345$ , тогда количество цифр  $kol = 5$ .

2.1. Структурная схема. Определение количества цифр в числе



2.2. Структурная схема. Определение максимальной цифры и её позиции в цифре.



Результаты вычислений приведены в таблицах:

**Определение количества цифр числа**

Kol	N
1	12345
2	12345 div 10 = 1234
3	1234 div 10 = 123
4	123 div 10 = 12
5	12 div 10 = 1
	1 div 10 = 0



### Определение текущей цифры числа

<b>i</b>	<b>Число M</b>
1	12345
2	$12345 \text{ div } 10 = 1234$
3	$1234 \text{ div } 10 = 123$
4	$123 \text{ div } 10 = 12$
5	$12 \text{ div } 10 = 1$

Если цифры числа известны, определить наибольшую из них не составит труда. Алгоритм поиска максимального значения в некоторой последовательности цифр заключается в следующем. В ячейку, в которой будет храниться максимальный элемент (*max*), записывают значение, меньшее любого из элементов последовательности (в нашем случае *max* = -1, так как цифры числа находятся в диапазоне от 0 до 9). Затем сравнивают элементы последовательности с значением ячейки *max*, если найдется элемент, превышающий значение предполагаемого максимума, то ячейке *max* необходимо присвоить значение этого элемента и, соответственно, запомнить его номер в последовательности (в нашем случае переменной *pos* присваивается значение параметра цикла). В алгоритме поиска минимума вначале в переменную *min* записываем значение, заранее большее любого элемента последовательности. Затем все элементы последовательности сравниваем с *min*. Если встретится значение меньше, чем *min*, переписываем его в переменную *min*.

### 3. Программный код.

```
var M,N,max:longint;
    i,kol,pos:word;
begin
  { Так как речь идет о натуральных числах, }
  { при вводе предусмотрена проверка. }
  { Закончить цикл, если введено положительное число, }
  { иначе повторить ввод. }
  repeat
    write ('N=') ; readln (N);
  until N>0;
  M:=N; { Сохранить значение переменной N. }
  kol:=1; { Предположим, что число состоит из одной цифры. }
  while M div 10 > 0 do {Выполнять тело цикла, пока число делится нацело на 10. }
  begin
    kol:=kol+1; { Счетчик количества цифр. }
    M:=M div 10; { Изменение числа. }
  end;
  max:=-1; { Присвоение максимуму начального значения. }
  pos:=1; { Предполагаемая позиция максимума в числе. }
  M:=N;
  For i:=kol downto 1 do { Нумерация цифр в числе слева }
  begin
    if M mod 10 > max then { Если цифра больше предполагаемого максимума, то }
```

```

begin
max:=M mod 10;      { заменить максимум этой цифрой }
pos:=i;             { запомнить 'ее позицию в числе}
end;
M:=M div 10;        { изменение числа. }
end;
writeln ( 'max=',max,' pos=',pos);
end.

```

### Задание для практической работы

#### Выполнение практических заданий.

Вариант	№ задач	Вариант	№ задач	Вариант	№ задач
1	1, 7, 37, 47, 71, 81	11	5, 17, 47, 57, 71, 91	21	3, 27, 46, 67, 71, 111
2	2, 8, 38, 48, 72, 82	12	6, 18, 37, 58, 72, 92	22	4, 28, 37, 68, 72, 112
3	3, 9, 39, 49, 73, 83	13	1, 19, 38, 59, 73, 93	23	5, 29, 38, 69, 73, 113
4	4, 10, 40, 50, 74, 84	14	2, 20, 39, 60, 74, 94	24	6, 30, 39, 70, 74, 85
5	5, 11, 41, 51, 75, 85	15	3, 21, 40, 61, 75, 95	25	1, 31, 40, 47, 75, 86
6	6, 12, 42, 52, 76, 86	16	4, 22, 41, 62, 76, 96	26	2, 32, 41, 48, 76, 87
7	1, 13, 43, 53, 77, 87	17	5, 23, 42, 63, 77, 97	27	3, 33, 42, 49, 77, 88
8	2, 14, 44, 54, 78, 88	18	6, 24, 43, 64, 78, 98	28	4, 34, 43, 50, 78, 100
9	3, 15, 45, 55, 79, 89	19	1, 25, 44, 65, 79, 99	29	5, 35, 44, 51, 79, 105
10	4, 16, 46, 56, 80, 90	20	2, 26, 45, 66, 80, 100	30	6, 36, 45, 52, 80, 113

1. Используя операторы цикла, составить таблицу значений функции (протабулировать функцию):

$$y = \sqrt{\operatorname{tg}^2 \frac{x}{5} + 2x^2 + 3}, \text{ если } 2 \leq x \leq 7, \Delta x = 0,56$$

где  $\Delta x$  – шаг, с которым изменяется аргумент  $x$ .

$$2. y = \sqrt{\sin^2 \frac{2\pi x}{5} + 6x^2 + |x|}, \text{ если } 3 \leq x \leq 9, \Delta x = 0,74$$

$$3. y = \sqrt{\cos^2 \frac{28^0}{5x} + 8x^2}, \text{ если } 1 \leq x \leq 5, \Delta x = 0,46.$$

$$4. y = \sqrt{\frac{\sin 12^0 + 3x^2}{\operatorname{tg} \frac{\pi}{7}}}, \text{ если } 1 \leq x \leq 4, \Delta x = 0,35.$$

5.  $y = \sqrt{\frac{\cos 22^\circ + 3x^2}{\operatorname{tg} \frac{\pi}{7} + 7x}}$ , если  $2 \leq x \leq 6$ ,  $\Delta x = 0,34$ .

6.  $y = \sqrt{\frac{3x^2 + \sin 20^\circ}{5x}}$ , если  $1 \leq x \leq 6$ ,  $\Delta x = 0,42$ .

7. Вычислить квадратный корень  $y = \sqrt{x}$  по рекуррентной формуле Герона

$$y = \frac{1}{2} \left( y + \frac{x}{y} \right) \text{ с точностью } \xi = 10^{-6}.$$

8. Составить схему алгоритма вычисления значения полинома

$$P = 11x^5 + 9x^4 + 7x^3 + 5x^2 + 3x + 1 \text{ при } x = 1.25$$

9. Подсчитать количество цифр в целом числе X.

10. Найти минимальное значение функции  $Y = x \sin x$ , на отрезке  $[c, d]$  с шагом  $\nabla x = 0.001$ .

11. Дано натуральное число n. Вычислить  $a(a+1)(a+2)\dots(a+n-1)$ .

12. Дано натуральное число n. Вычислить  $\left(1 + \frac{1}{a}\right)\left(1 + \frac{1}{2a}\right)\dots\left(1 + \frac{1}{na}\right)$

13. Дано натуральное число n. Вычислить  $a(a-n)(a-2n)\dots(a-nn)$

14. Дано натуральное число x. Подсчитать, сколько цифр в этом числе и чему равна их сумма.

15. Дано натуральное число x. Найти первую цифру этого числа.

16. Даны натуральные числа m, n. Получить сумму m последних цифр числа n.

17. Дано 50 вещественных чисел. Найти величину наибольшего из них.

18. Дана непустая последовательность различных натуральных чисел, за которой следует 0. Определить порядковый номер наименьшего из них.

19. Даны целое  $n > 0$  и последовательность из n вещественных чисел, среди которых есть хотя бы одно отрицательное число. Найти величину наибольшего среди отрицательных чисел этой последовательности.

20. Дано n вещественных чисел. Вычислить разность между максимальным и минимальным из них.

21. Составить программу, определяющую, является ли данное число простым.

22. Найти сумму первых чисел Фибоначчи, не превышающих числа a.

23. Дано n вещественных чисел. Определить, образуют ли они возрастающую последовательность.

Из истории. Грекам были известны первые четыре совершенных числа: 6, 28, 496, 8128. Эти числа высоко ценились. Даже в 12 веке церковь утверждала, что для спасения души необходимо найти пятое совершенное число. Это число было найдено только в 15 веке. До сих пор совершенные числа полностью не исследованы – не известно, имеется ли конечное число совершенных чисел или их число бесконечно, кроме того, не известно ни одного нечетного числа, но и не доказано, что таких чисел нет.

### Вопросы для самоконтроля

1. Какие циклы называются условными?
2. Дайте определение понятию «итерация».
3. Какой вид цикла можно использовать, если заранее не известно число итераций?
4. При использовании какого вида цикла, тело цикла может не вычисляться не разу?
5. При использовании какого вида цикла, тело цикла обязательно вычисляется хотя бы один раз?
6. Напишите программу вычисления  $N!$
7. Напишите программу вычисления  $a^n$ .

## Практическая работа №21

### Тема «Составление программ с одномерными и двумерными массивами».

#### Цель:

- развить навыки обработки одномерных массивов;
- развить навыки обработки двумерных массивов;
- развить навыки тестирования программных кодов.

#### Обеспечение практической работы:

- тетрадь для выполнения практических работ;
- канцелярские принадлежности;
- компилятор Паскаль ABC;
- калькулятор;
- задания для выполнения работы;
- материал лекций.

#### В результате выполнения данной работы студент должен уметь:

- составлять программы ввода, вывода массивов;
- составлять программы сортировки массивов;
- составлять программы поиска элементов массива по заданным критериям;
- тестировать программные коды.

## Теоретический материал

### Пример 1.

Рассмотрим алгоритм пузырьковой сортировки.

#### 1. Построение математической модели.

Сравним первый элемент массива со вторым. Если первый окажется больше второго, то поменяем их местами. Те же действия выполним для второго и третьего, третьего и четвертого,  $i$ -го и  $(i + 1)$ -го,  $(n - 1)$ -го и  $n$ -го элементов. В результате этих действий самый большой элемент станет на последнее  $(n$ -е) место. Теперь повторим данный алгоритм сначала, но последний  $(n$ -й) элемент, рассматривать не будем, так как он уже занял свое место. После проведения данной операции самый большой элемент оставшегося массива встанет на  $(n - 1)$ -е место. Так повторяем до тех пор, пока не упорядочим весь массив.

Обратите внимание на то, что для перестановки элементов (блок 4) используется буферная переменная  $b$ , в которой временно хранится значение элемента, подлежащего замене.

Процесс упорядочивания элементов в массиве по возрастанию представим в таблице:

	Номер элемента				
	1	2	3	4	5
Исходный массив	7	3	5	4	2
Первый просмотр	3	5	4	2	7
Второй просмотр	3	4	2	5	7
Третий просмотр	3	2	4	5	7
Четвертый просмотр	2	3	4	5	7

## 2. Программный код.

Листинг программы упорядочивания элементов в массиве по возрастанию их значений.

```
var
i, n, j: integer;
b: word;
y: array [1 .. 100] of word;
begin
  writeln ( 'введите размер массива' );
  readln (n);
  for i :=1 to n do
    begin
      write ('y[', i, '=');
      readln (y[i]);
    end;
  writeln ('массив y');
  for i:=1 to n do write (y [ i ], ' ');
  writeln;
  repeat
    f:=0;
    if y[ i ] > y [ i + 1 ] then           { Если текущий элемент больше следующего, то }
      begin                               { поменять их местами. }
        b:=y[ i ] ;                       { Сохранит значение текущего элемента. }
        y[ i ]:=y[i + 1] ;                 { Заменить текущий элемент следующим. }
        y[i + 1]:=b ;                     { Заменить следующий элемент текущим. }
        f:=1;
      end;
    writeln ( 'упорядоченный массив' ) ;
  for i :=1 to n do
    write (y[ i ], ' ');
  writeln;
end.
```

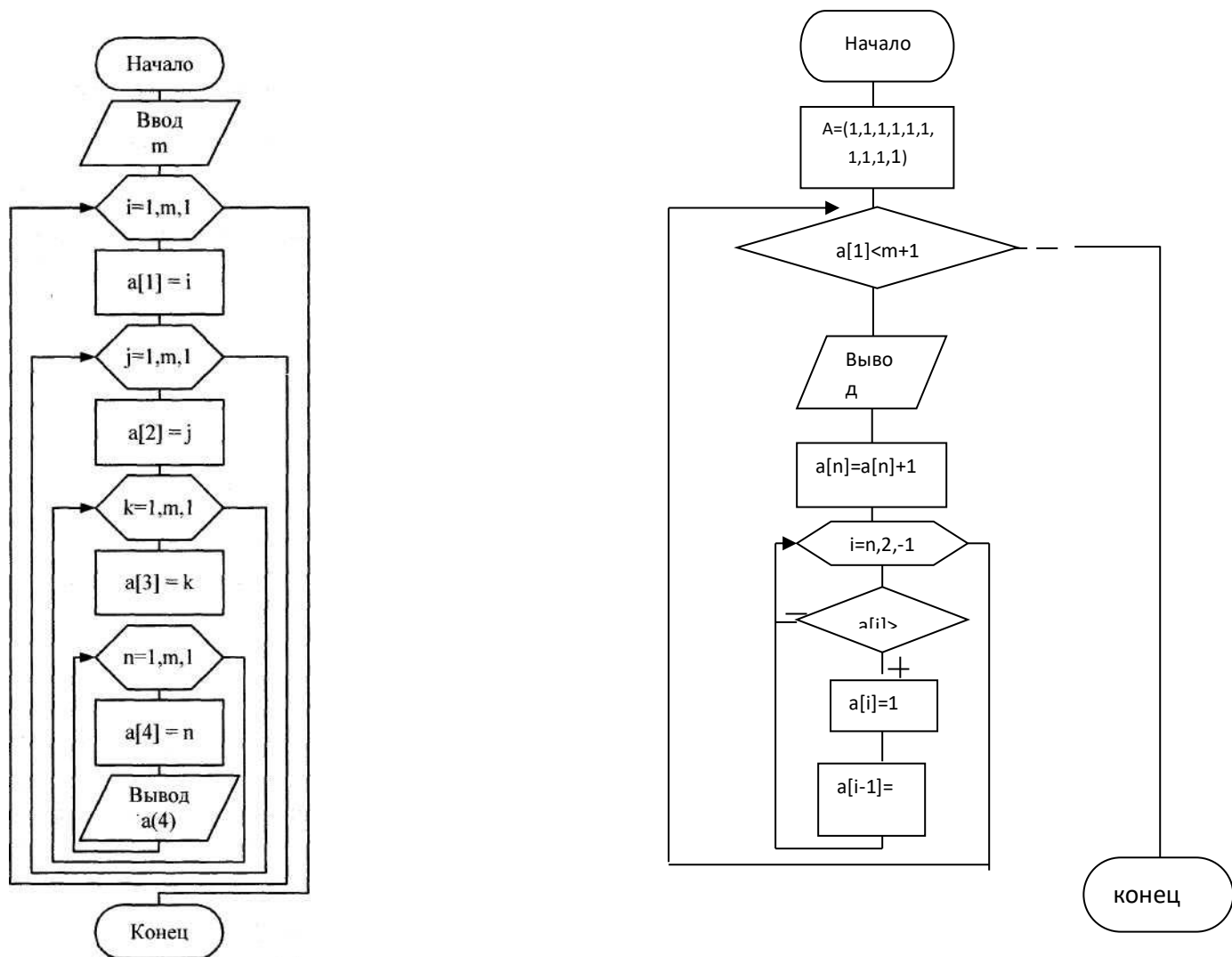
**Пример 2.** Разработать программу генерации следующих комбинаций: 1111, 1112, 1113, 1121, 1122, 1123, 1131, 1132, 1133, 1211, ..., 3333.

**Вариант 1.** Для генерации комбинаций используем вложенные циклы. Количество разрядов - 4. Следовательно, для генерации всех комбинаций понадобится 4 цикла, вложенных один в другой (рис. а). Если ввести параметр m - максимальное значение в каждом разряде, то эта же программа будет генерировать комбинации от 1111 до mmmm.

Если количество разрядов генерируемой комбинации изменить, то придется переписывать программу, увеличивая или уменьшая количество вложенных циклов.

**Вариант 2.** С точки зрения увеличения универсальности программы для генерации всех комбинаций лучше использовать программно реализуемый счетчик в общем случае на n разрядов, который в каждом разряде считает от 1 до n. Каждое состояние такого счетчика соответствует комбинации. Для генерации следующего варианта в младший разряд счетчика добавляют 1 и осуществляют межразрядные переносы в соответствии с характеристиками каждого разряда.

Ниже представлена программа, реализующая данный алгоритм для  $n < 10$ .



Два варианта алгоритма генерации комбинаций:

а - с использованием вложенных циклов; б - программно реализуемый счетчик.

**Program ex2;**

**const**

$a$ :array[1..10] of byte=(1, 1,1,1,1,1,1,1,1,1);

**var** i,m,n:integer;

**begin**

**readLn** (n,m);

**while**  $a[1] < m+1$  **do** {условие «сброса» счетчика}

**begin**

**for**  $i:=1$  **to**  $n$  **do** **write** ( $a[i]$ ); {вывод комбинации}

**write** (' ');

$a[n]:=a[n]+1$ ; {добавление 1 в последний разряд}

**for**  $i:=n$  **downto** 2 **do** {анализ и осуществление переносов}

**if**  $a[i] > m$  **then**

**begin**

$a[i]:=1$ ;

$a[i-1]:=a[i-1]+1$ ;

**end**;

*end;*

*End.*

Ассоциируя комбинации счетчика с вариантами данных и проверяя полученные комбинации по критериям конкретной задачи, можно осуществить полный перебор вариантов.

Как уже упоминалось выше, для реализации ограниченного перебора используется вторая стратегия, при которой генерацию и анализ комбинаций исходных данных выполняют поэтапно. С этой целью удобно использовать рекурсию, при которой процесс генерации комбинации продолжается, пока полученная часть комбинации перспективна, или не исчерпаны все варианты.

Для генерации всех комбинаций исходных данных можно использовать вложенные циклы или программно реализуемый счетчик (по типу, например, электросчетчика). Второй способ позволяет получить более общее решение.

**Ограниченный перебор, реализованный с использованием рекурсии.** Для генерации комбинаций будем использовать древовидную рекурсию, на каждом уровне которой ставится один ферзь  $m$  различными способами. При этом появляется возможность проверки перспективности уже имеющейся комбинации. Поясним сказанное на примере доски  $4 \times 4$ .

На первом шаге первого ферзя можно поставить на одно из четырех полей первого вертикального ряда, что соответствует комбинациям

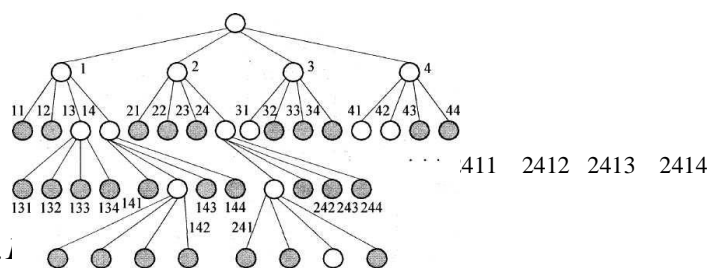
1..., 2..., 3..., 4....

На втором шаге второго ферзя можно поставить на одно из четырех полей второго вертикального ряда. При этом мы получим 4 комбинации для каждого варианта установки первого ферзя:

11.., 12.., 13., 14.. 21., 22., 23., 24.. 31., 32., 33., 34.. 41., 42., 43., 44..

На этом этапе уже можно исключить все варианты, для которых первые два ферзя бьют друг друга: 11., 12., 21., 22., 23., 32., 33., 34., 43., 44.. Генерацию остальных комбинаций необходимо продолжить, установив третьего ферзя одним из четырех способов и исключив неперспективные решения. На последнем этапе необходимо установить четвертого ферзя и опять исключить варианты, в которых ферзи бьют друг друга. Оставшиеся варианты являются решениями задачи.

На рисунке показан фрагмент дерева генерации вариантов для  $m = 4$  с отсечением неперспективных ветвей. Первое найденное решение - комбинация 2413.



**Program** ex21;

**Type**  $p = \text{array}[l \dots]$

**Var**

$Pole: p;$

$k, m: \text{integer};$

{ функция проверки перспективности комбинации }

**function**  $\text{new\_r}(n: \text{integer}; pole: p): \text{boolean};$

$\text{var } j: \text{integer};$

**begin**

$\text{new\_r} := \text{false};$

**for**  $j := l$  **to**  $n - 1$  **do**

**if**  $(pole[j] = pole[n])$  **or**  $(\text{abs}(pole[j] - pole[n]) = n - j)$  **then exit;**

$\text{new\_r} := \text{true};$

**End;**



{рекурсивная функция генерации комбинации}

```

procedure ferz (n,m:integer; var pole.p);
  var i.integer;

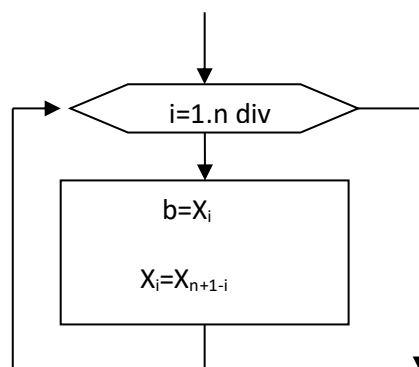
  begin
    if n=m+l then {если установлено m ферзей, то вывести решение}
      begin
        for i:=l to m do write(polefi]:2);
        writeln;
      end
    else {иначе - пытаемся установить следующего ферзя}
      for i:=l to m do {m способами}
        begin
          pole[n]:=i; {установка n-го ферзя}
          if new_r(n,pole) {проверка перспективности комбинации}
            thenferz (n+l,m,pole); {рекурсивный вызов установки
            следующего ферзя}
        end;
      end;
  end;
  {основная программа}
  begin
    writeLn ('Введите размер доски:');
    readLn (m);
    k:=0;
    ferz (l,m,pole); end.
  
```

Размер доски	Количество вариантов, которое проверяется при	Количество вариантов, рассмотренных при	Количество полученных решений
4x4	$4^4 = 256$	17	2
5x5	$5^5 = 3125$	54	10
6x6	$6^6 = 46656$	153	4
7x7	$7^7 = 823543$	552	40
8x8	$8^8 = 16777216$	2057	92

**Пример 3.** Переписать элементы массива x в обратном порядке. Блок-схема Представлена на рис.4.7. Алгоритм состоит в следующем: меняем местами 1-й и n-й элемент, затем 2-й и (n-1)-й и т.д. середины массива (элемент с номером I следует обменять с элементом (n+1-i)).

```

program mas_5;
  type
    massiv=array[1..100]of real;
  var
    x:massiv;
    i, n : integer;
    b : real;
  begin
    writeLn ('введите размер массива');
    readLn(n);
    for i:=1 to n do
      begin
        write ('x[',i,']=');
        readln (x [ i ]);
      end;
  end;
  
```



```

{Элементы массива меняются местами : 1-й – с n-м, 2-й- с (n-1) и т.д.}
for i :=1 to n div 2 do
begin
  b:=x[n+1-i];
  x[n+1-i]:=x[i];
  x[i]:=b;
end;
writeln ( ' преобразованный массив');
for i :=1 to n do
  write ( x [i]:1:2,');
end.

```

**Пример 4.** Найти сумму элементов матрицы, лежащих выше главной диагонали.

Алгоритм решения главной задачи построен следующим образом: обнуляется ячейка для накопления суммы (переменная s). Затем с помощью двух циклов (первый по строкам, второй по столбцам) просматривается каждый элемент матрицы, но суммирование происходит только в том случае, если этот элемент находится выше главной диагонали, то есть выполняется свойство  $i < j$ .

```

program ex6;
var
a:array [1..15,1..10] of real;
i,j,n,m: integer;
s: real;
begin
writeln(' введите размеры матрицы');
writeln ('n-количество строк, m-количество столбцов');
readln (n,m);
for i:=1 to n do
for j:=1 to n do
begin
write ('a[',i, ', ',j, '=');
readln (a[i,j]);
end;
S:=0;
for i:=1 to n do
for j:=1 to n do
if j>i then {Если элемент лежит выше главной диагонали, то }
s:=s+a [i,j]; {наращиваем сумму. }
writeln ('матрица A');
for i:=1 to m do
{Здесь важен формат, особенно общая ширина поля! }
write (a[i, j] :8:3, ' ');
writeln
end;
writeln ('сумма элементов матрицы' , s:8:3);
end.

```

**Пример 5.** Преобразовать матрицу A (m,n) так, чтобы строки с нечетными номерами были упорядочены по убыванию, с четными – по возрастанию.

```

program ex7;
var

```

```

a: array [1..15,1..15] of real;
j, i, k, m, n: byte;
b: real;
begin
  writeln ('введите m и n');
  readln (m,n);
  for i:=1 to m do
    for j:=1 to n do
      begin
        write ('a[',i,',',j,'] =');
readln (a[i,j]);
        end;
      writeln ('матрица a');
    for i:=1 to m do
      begin
        for j:=1 to n do
          write (a[i,j]:7:3, ' ');
          writeln;
        end;
      for i:=1 to m do
        if (I mod 2)=0 then {если номер строки четный, то упорядочить ее элементы по
возрастанию.}
        begin
          for k:1 to n-1 do
            for j:1 to n-k do
              if a[i,j] >a[i,j+1] then
                begin
                  b:=a[i,j];
                  a [i,j]:=a [i,j+1];
                  a[i,j+1]:=b;
                end;
            end;
          else {если номер строки нечетный, то упорядочить ее элементы по убыванию.}
            for k:=1 to n-1 do
              for j:=1 to n-k do
                if a[i,j]< a[i,j+1] then
                  begin
                    b:=a[i,j];
                    a[i,j]:=a [i,j+1];
                    a[i,j+1]:=b;
                  end;
                writeln ('преобразованная матрица a');
              for i:=1 to m do
                begin
                  for j:=1 to m do
                    write (a [i,j]:7:3, ' ');
                  writeln;
                end;
              end.

```

**Пример 6.** Написать программу умножения двух матриц  $A(n, m)$  и  $B(m, l)$ . Например, необходимо перемножить две матрицы:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} * \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix}.$$

Воспользуемся правилом «строка на столбец», получим матрицу

$$\begin{pmatrix} a_{11} * b_{11} + a_{12} * b_{21} + a_{13} * b_{31} & a_{11} * b_{12} + a_{12} * b_{22} + a_{13} * b_{32} \\ a_{21} * b_{11} + a_{22} * b_{21} + a_{23} * b_{31} & a_{21} * b_{12} + a_{22} * b_{22} + a_{23} * b_{32} \\ a_{31} * b_{11} + a_{32} * b_{21} + a_{33} * b_{31} & a_{31} * b_{12} + a_{32} * b_{22} + a_{33} * b_{32} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ c_{31} & c_{32} \end{pmatrix}.$$

В общем виде формула для нахождения элемента  $C_{ij}$  матрицы выглядит следующим образом:

$$C_{ij} = \sum_{k=1}^m A_{ik} B_{kj},$$

где  $i = 1, N$  и  $j = 1, L$ . Обратите внимание, что проводить операцию умножения можно только в том случае, если количество строк левой матрицы совпадает с количеством столбцов правой. Кроме того,  $A \times B \neq B \times A$ .

**var**

$a, b, c : \text{matrica};$

$i, j, m, n, l, k : \text{bute};$

$s : \text{real};$

**begin**

*writeln* (' $\hat{A} \hat{a} \hat{a} \hat{a} \hat{e} \hat{o} \hat{n}, m \hat{e} \hat{l}$ ');

*readln* ( $n, m, l$ );

*for*  $i := 1$  *to*  $n$  *do*

*begin*

*write* (' $a[', i, ', ', j, ']='$ );

*readln* ( $a[i, j]$ );

*end*;

*for*  $i := 1$  *to*  $m$  *do*

*for*  $j := 1$  *to*  $l$  *do*

*begin*

*write* (' $b[', i, ', ', j, ']='$ );

*readln* ( $b[i, j]$ );

*end*;

*for*  $i := 1$  *to*  $n$  *do*

*for*  $j := 1$  *to*  $l$  *do*

*begin*

{*B* переменной *S* будет храниться результат скалярного произведения *i*-й строки на *j*-й столбец.}

$S := 0$ ;

*for*  $k := 1$  *to*  $m$  *do*  $s := s + a[i, k] * b[k, j]$ ;

$c[i, j] := s$ ;

*end*;

*writeln* ('Матрица  $a$ ');

*for*  $i := 1$  *to*  $n$  *do*

**begin**

*for*  $j := 1$  *to*  $m$  *do* *writeln* ( $a[i, j]:7:3, ' '$ );

*writeln*;

**end**;

*writeln* ('Матрица  $b$ ');

*for*  $i := 1$  *to*  $m$  *do*

**begin**

```

        for j:=1 to l do write (b[i, j]:7:3, ' ');
        writeln;
end;
writeln ('Матрица c=a*b');
for i:=1 to n do
    begin
        for j:=1 to l do write (c[I,j]:7:3, ' ');
        writeln;
    end;
end.

```

**Пример 7.** Поменять местами n-й и 1-й столбцы матрицы A(k,m).

```

program ex9;
type
matrica = array [1..15,1..15] of real;
var
a : matrica;
i, j, k, m, n, l : byte;
b : real;
begin
write ('k=');
readln (k);
write ('m=');
readln (m);
for i:=1 to k do
for j:=1 to m do
begin
write ('a [', i, ', ', j, ']=');
readln (a[i, j])
end;
repeat
write ('n=');
readln (n);
write ('l=');
readln (l);
until (n<=m) and (l<=m) and (n<=l); {Ввод считается верным, если n и l меньше m и не верны
друг другу.}

for I:=1 to k do {Элементы столбца с номером 1 заменить элементами столбца с номером
n.}
begin
b :=a[i,n];
a[i,n]:= a[i,l];
a[i,l]:=b
end;
for I :=1 to k do
begin
for j:=1 to m do write ('a [', i, ', ', j, ']:7:3, ' ');
writeln;
end;

```

*end.*

**Пример 8.** Преобразовать матрицу  $A (m,n)$  таким образом, чтобы каждый столбец был упорядочен по убыванию. Алгоритм решения этой задачи сводится к тому, что уже известный нам по предыдущей главе алгоритм упорядочивания элементов в массиве выполняется для каждого столбца матрицы.

```
program ex10;  
type  
  matrica = array [1..15,1..15] of real;  
var  
  a : matrica ;  
  I,j,k,m,n : byte ;  
  b: real;  
begin  
  write ( 'm=' );  
  readln ( m );  
  write ( 'n=' );  
  readln ( n );  
  for i:=1 to m do  
    for j :=1 to n do  
      begin  
        write ( 'a[',i,',',j,'] =' );  
        readln ( a[I,j] );  
      end;  
  writeln ( 'матрица a' );  
  for i:=1 to m do  
    begin  
      for j:=1 to n do write ( a[I, j]:7:3, ' ');  
    writeln;  
    end;  
  for j:=1 to n do    {j- номер столбца.}  
    for k:=1 to m-1 do {k- номер просмотра.}  
      for i:=1 to m-k do {I – номер строки .}  
        if a [I,j] <a [i+1,j] then  
          begin  
            b:= [a , j];  
            a [I,j] :=a[i+1,j];  
            a [i+1,j] :=b;  
          end;  
        writeln ( ' преобразованная матрица a' );  
  for i:= to n do  
    begin  
      for j:= to n do write ( a[I,j] :7:3, ' ');  
    writeln;  
    end;  
end.
```

**Задание для практической работы**

Вариант	№ задач	Вариант	№ задач	Вариант	№ задач
1	1, 19, 37, 62, 82, 100, 120	11	11, 29, 47, 72, 92, 110, 130	21	3, 21, 57, 62, 99, 105, 121
2	2, 20, 38, 63, 83, 101, 121	12	12, 30, 48, 73, 93, 111, 131	22	4, 22, 58, 63, 82, 106, 122
3	3, 21, 39, 64, 84, 102, 122	13	13, 31, 49, 74, 94, 112, 132	23	5, 23, 59, 64, 83, 107, 123
4	4, 22, 40, 65, 85, 103, 123	14	14, 32, 50, 75, 95, 113, 133	24	6, 24, 60, 65, 84, 108, 124
5	5, 23, 41, 66, 86, 104, 124	15	15, 33, 51, 76, 96, 114, 134	25	14, 25, 61, 66, 85, 109, 125
6	6, 24, 42, 67, 87, 105, 125	16	16, 34, 52, 77, 97, 115, 135	26	15, 26, 62, 67, 86, 110, 126
7	7, 25, 43, 68, 88, 106, 126	17	17, 35, 53, 78, 98, 116, 136	27	16, 27, 61, 68, 87, 111, 127
8	8, 26, 44, 69, 89, 107, 127	18	18, 36, 54, 79, 99, 117, 137	28	17, 28, 40, 69, 88, 112, 128
9	9, 27, 45, 70, 90, 108, 128	19	1, 19, 55, 80, 86, 118, 138	29	18, 29, 39, 70, 89, 113, 129
10	10, 28, 46, 71, 91, 109, 129	20	2, 20, 56, 81, 93, 119, 120	30	10, 30, 46, 71, 90, 114, 130

1. Подсчитать и вывести на экран количество элементов массива, равных заданному значению. Заданное значение вводится с клавиатуры.
2. Составить и вывести на экран новый массив с номерами элементов исходного массива, которые равны заданному значению. Заданное значение вводится с клавиатуры.
3. Составить и вывести на экран новый массив со значениями элементов исходного массива, которые не равны заданному значению. Заданное значение вводится с клавиатуры.
4. Поменять местами максимальный и  $x$  минимальный элементы массива. Вывести измененный массив на экран.
5. Подсчитать и вывести на экран количество отрицательных, положительных и нулевых элементов массива.
6. Все элементы массива, меньшие заданного значения, и их номера записать в новые массива. Вывести новые массивы на экран. Заданное значение вводится с клавиатуры.
7. Все элементы массива, большие заданного значения, и их номера записать в новые массивы. Вывести новые массивы на экран. Заданное значение вводится с клавиатуры.
8. Подсчитать и вывести на экран процент нулевых элементов в исходном массиве.
9. Определить и выдать сообщение на экран, какая сумма по абсолютной величине больше, отрицательных или положительных элементов вектора.
10. Вводится дополнительный массив разрешенных значений. Определить и вывести на экран, сколько элементов исходного массива имеют неразрешенные значения.
11. Вводится дополнительный массив разрешенных значений. Определить и вывести на экран, сколько элементов исходного массива имеют разрешенные значения.
12. Вводится дополнительный массив разрешенных значений. Составить массив из элементов исходного массива, имеющих неразрешенные значения. Вывести результирующий массив на экран.
13. Составить и вывести на экран массив с  $M$  минимальными значениями исходного массива.  $M$  вводится дополнительно.
14. Составить и вывести на экран массив с  $M$  максимальными значениями исходного массива.  $M$  вводится дополнительно.

15. Переписать элементы массива в обратном порядке на том же месте. Вывести измененный массив на экран.
16. Составить и вывести на экран массив из различных элементов исходного массива.
17. Составить и вывести на экран массив номеров элементов исходного вектора, встречающихся один раз.
18. Составить и вывести на экран два массива: массив повторяющихся элементов исходного массива и массив их частот.

19 – 36. Ввести размер прямоугольной матрицы и значения ее элементов. Построить вектор В, где В(i) -

19. число неотрицательных элементов в i-ой строке.
20. число нулевых элементов в i-ом столбце.
21. среднее арифметическое отрицательных элементов в i-ой строке.
22. среднее арифметическое положительных элементов в i-ом столбце.
23. минимальное значение в i-ой строке.
24. максимальное значение в i-ом столбце.

### Вопросы для самоконтроля

1. Какими параметрами характеризуется массив?
2. Чем отличается динамический массив от статического?
3. Опишите двумерный массив – константу размерностью 2 x 4.
4. Какие методы сортировки элементов массива Вам известны?
5. Опишите алгоритм сортировки массива методом «пузырька».
6. Опишите алгоритм сортировки массива методом выбора.
7. Составьте программу нахождения минимального элемента одномерного массива.
8. Составьте программу нахождения суммы положительных элементов двумерного массива.
9. С переменными, относящимися к перечисленному типу данных, можно выполнить операции:
  - 1) присваивания;
  - 2) сложения;
  - 3) вывода на экран;
  - 4) отношения;
  - 5) умножения на константу.
10. Выберите все правильные варианты ответа. При описании массива тип индекса может быть:
  - 1) вещественным;
  - 2) byte;
  - 3) интервальным;
  - 4) integer;
  - 5) перечисляемым.
9. Выберите все правильные утверждения. С массивами в целом можно выполнять операции:
  - 1) сравнения;
  - 2) присваивания;
  - 3) сложения;
  - 4) вычитания;
  - 5) умножения.
11. Какое число получится в результате работы программы?
 

```
const a : array[1.. 8] of integer = (3, 8, 0, -6, 0, -1, -9, 3);
var i , k : integer;
begin
```



```
k := abs(a[8]);
for i := 1 to 8 do
  if abs(a[i]) > k then k := abs(a[i]);
  write(k)
end.
```

12. Какую величину вычисляет программа?

```
const a : array[1..8] of integer = (3, 8, 0, -6, 0, -1, -9, 3);
var i, k : integer;
begin
  k := 0; j := 4;
  for i := 1 to 8 do
    if abs(a[i]) > j then k := a[i];
    write(k)
  end.
```

13. Составить алгоритмы для решения следующих задач:

- 1) Первый из элементов массива, больший, чем 4.
- 2) Первый из элементов массива, больший по модулю, чем 4.
- 3) Модуль последнего из элементов массива, больших, чем 4.
- 4) Модуль первого из элементов массива, больших, чем 4.
- 5) Последний из элементов массива, больший по модулю, чем 4.

## **Практическая работа №22**

### **Тема «Самостоятельное составление программ по заданным критериям».**

#### **Цель:**

- закрепить навыки обработки одномерных массивов;
- закрепить навыки обработки двумерных массивов;
- закрепить навыки тестирования программных кодов.

#### **Обеспечение практической работы:**

- тетрадь для выполнения практических работ;
- канцелярские принадлежности;
- компилятор Паскаль ABC;
- калькулятор;
- задания для выполнения работы;
- материал лекций.

#### **В результате выполнения данной работы студент должен уметь:**

- составлять программы ввода, вывода массивов;
- составлять программы сортировки массивов;
- составлять программы поиска элементов массива по заданным критериям;
- составлять программы обработки массивов по сложным критериям;
- тестировать программные коды.

#### **Задание для практической работы**

##### **Практическая работа состоит из пяти задач.**

Задача 1. Задачи, содержащие условие отбора.

Задача 2. Задачи, содержащие множественные условия отбора.

Задача 3. Циклические структуры.

Задача 4. Целочисленная арифметика.

Задача 5. Одномерные массивы.

Время выполнения работы: 80 минут.

Работы выполняются согласно полученной карточки

## Практическая работа № 23

**Тема:** «Создание, ввод, редактирование и форматирование текста, абзаца. Междустрочные и межсимвольные интервалы. Интервалы между абзацами в MSWord».

**Цель работы:** изучение интерфейса текстового процессора MSWord, знакомство с панелями инструментов, изучение основных приемов создания и редактирования документов, основных приемов работы с абзацами.

### **Задание:**

Для получения практических навыков работы в текстовом процессоре *MICROSOFT WORD* выполните следующее задание:

1) Создайте новый документ в папке *Практические студентов*. Для этого откройте необходимую папку, нажмите правой кнопкой мыши (п.к.м.) и в контекстном меню выберите команду *Создать – Документ MicrosoftWord*.

2) Наберите ниже приведенный текст, делая абзацные переходы (нажатие клавиши *Enter*) в конце смысловых предложений и отформатируйте его.


*Текст к заданию:*

### **Закон РФ «О персональных данных»**

Настоящим Федеральным законом регулируются отношения, связанные с обработкой персональных данных, осуществляемой федеральными органами государственной власти, органами государственной власти субъектов Российской Федерации, иными государственными органами (далее - государственные органы), органами местного самоуправления, иными муниципальными органами (далее - муниципальные органы), юридическими лицами и физическими лицами с использованием средств автоматизации, в том числе в информационно-телекоммуникационных сетях, или без использования таких средств, если обработка персональных данных без использования таких средств соответствует характеру действий (операций), совершаемых с персональными данными с использованием средств автоматизации, то есть позволяет осуществлять в соответствии с заданным алгоритмом поиск персональных данных, зафиксированных на материальном носителе и содержащихся в картотеках или иных систематизированных собраниях персональных данных, и (или) доступ к таким персональным данным

Для форматирования текста выполните следующие действия:

1. Оформите заголовок текста следующим образом: шрифт – полужирный, курсивный *Times New Roman*, выравнивание *По центру*, размер 16 пт, вразрядку с интервалом 4 пт. Используйте команды на панели инструментов на вкладке *ГЛАВНАЯ – Шрифт*

 (Рисунок 1), *Шрифт – Дополнительно – Интервал – Разреженный* (Рисунок 2), предварительно выделив текст;

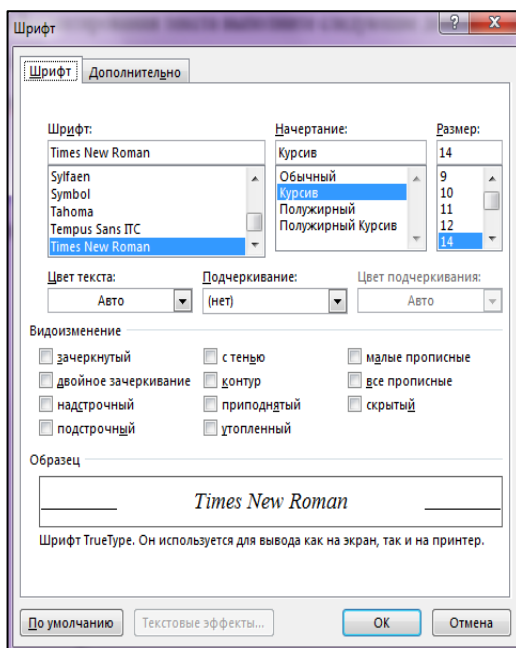


Рисунок 1

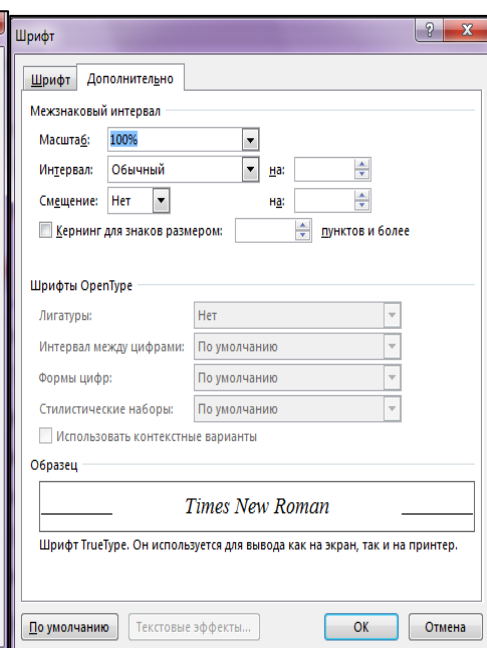


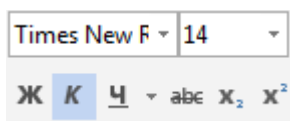





Рисунок 2

Текст оформите следующим образом, установите: отступ первой строки (Красная строка)  (Вид/Линейка); выравнивание текста *По ширине* , шрифт – *Arial*, полужирный курсив, размер 12 пт. Для этого предварительно выделите текст и щелкните последовательно левой кнопкой мыши (л.к.м.) на кнопках панели инструментов *Ж*, *К*, *Размер шрифта - 12*;



2. Наведите указатель мыши на кнопку  «Междустрочный интервал» и установите интервал равный – «1,15»

3. Попробуйте заключить последнее словосочетание в рамку. Для этого предварительно выделив текст, выполните последовательно команды *Граница*  - *Границы и заливка*  *Границы и заливка...* . В диалоговом окне *Границы* (Рисунок 3) установите параметры: *Ширина*: 2,25 пт, *Цвет*: авто. В диалоговом окне *Заливка* установите параметры: *Узор*: 20%.

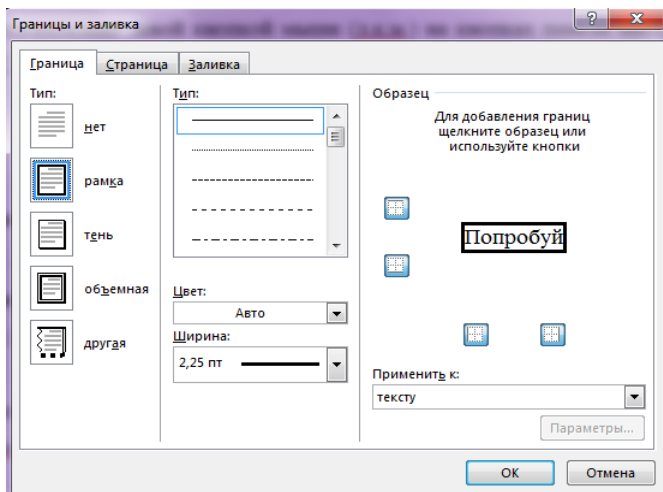


Рисунок 3

1. Задайте установки режима сохранения, выполнив команду *Файл - Параметры*, вкладка *Сохранение*, поле *Автосохранение каждые 10 мин* (Рисунок 4).

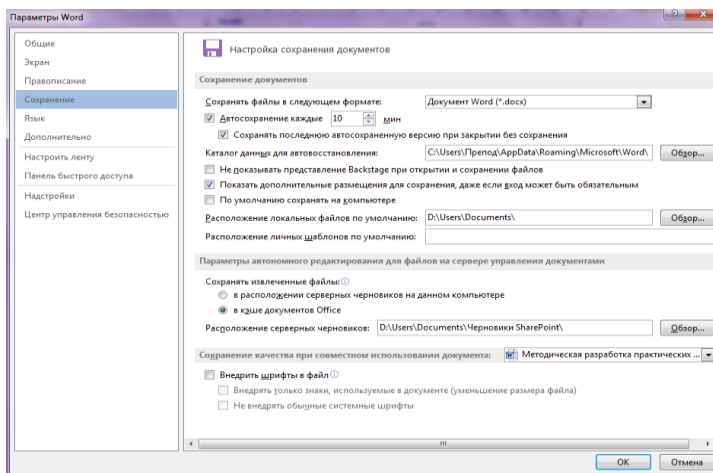


Рисунок 4

3. Сохраните файл под именем *Практическая работа №1 (Фамилия Имя)* в папке *Работы студентов* в формате *Word*, последовательно выполнив команды *Файл - Сохранить как*.

4. Создайте элемент автозамены: *персональные данные*, используя аббревиатуру *пд*. Для этого выполните следующие действия: нажмите на вкладку *ФАЙЛ*, выберите команду *Параметры – Правписание – кнопка Параметры автозамены* (рисунок 7) и в окне *Автозамена* выберите *заменить пд на персональные данные*. Нажмите кнопку *Добавить*, затем выберите новую команду из списка и нажмите *Ок*.

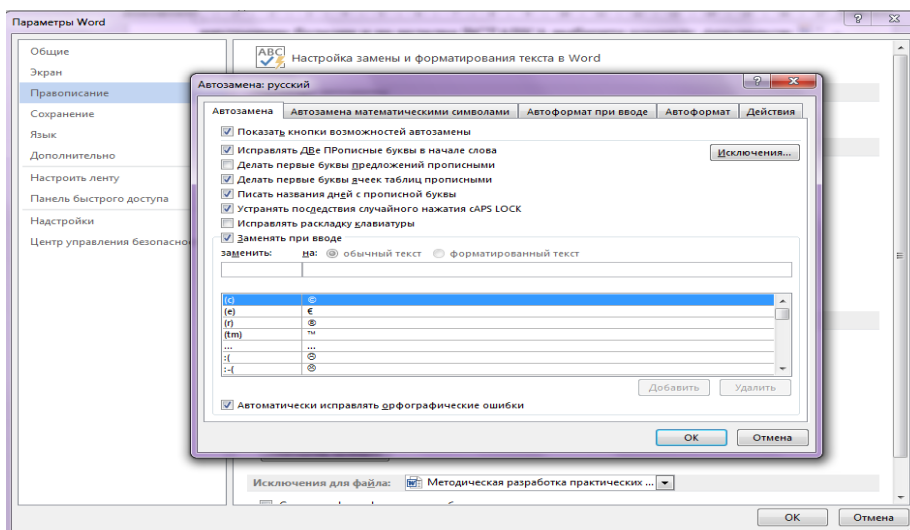
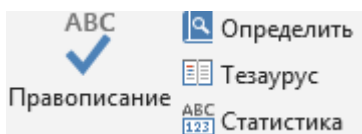
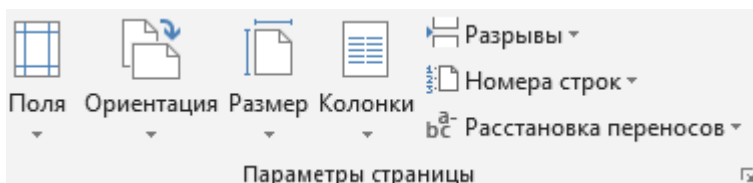


Рисунок 7

5. Предварительно установив курсор в начало текста, проверьте орфографию текста, выполнив команду *Рецензирование - Правописание*.



6. Настройте параметры страницы с помощью последовательности команд *Разметка страницы - Параметры страницы*.



В окне *Параметры страницы* (рисунок 8) выберите вкладку *Поля* и установите следующие значения: *Верхнее* - 2см, *Нижнее* - 2см, *Левое* - 3см, *Правое* - 3см. На вкладке *Источник бумаги* в группе *От края до колонтитула* установите *верхнего* - 1см, *нижнего* - 1см. В поле *вертикальное выравнивание*, выберите *По верхнему краю*, и далее в поле *Применить* укажите *Ко всему документу*.

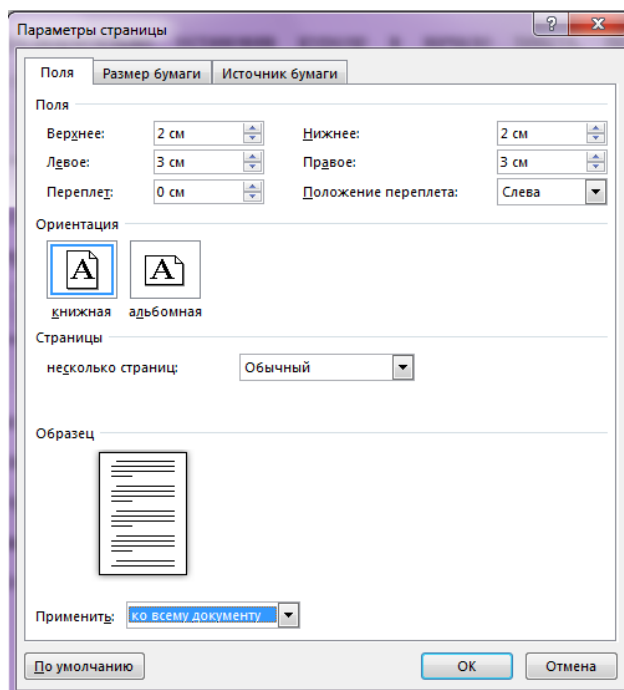
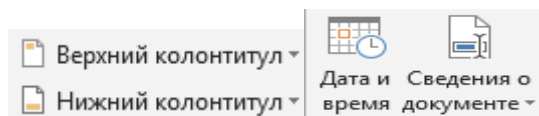


Рисунок 8.

7. На вкладке ВСТАВКА сформируйте верхний колонтитул (текст по центру, с нижним подчеркиванием) и нижний колонтитул (дата, время). В качестве текста верхнего колонтитула введите, например, следующий текст: «Закон РФ «О персональных данных»».



### Задание для самостоятельной работы

\*Перейдите на следующую страницу с помощью комбинации клавиш **CTRL+ENTER** и введите текст:

Набрать приведенный ниже текст, соблюдая оформление, и установив параметры страницы: верхнее поле – 1,5 см, нижнее – 1,5 см, левое – 2 см, правое – 1,5 см, ориентация – альбомная. Параметры абзаца: выравнивание – по ширине, отступ первой строки – 1,25 см, интервал перед – 0 пт, после – 6 пт. Основной текст – шрифт Tahoma, размер 12. Заголовок – выравнивание по центру, полужирный, размер 14.

## История создания и развития компьютерной техники.

Бытующее мнение о том, что компьютеры — изобретение XX столетия, не совсем верно. С тех пор как человек появился на земле и занялся производством и торговлей, он стал нуждаться в системе подсчетов и вычислений. Более тысячи лет эти операции выполнялись либо вручную, либо с помощью устройств вроде абака — счетной доски с камешками, разделенной на полосы. Человеческая мысль не стоит на месте, и вслед за абаком появились счеты, сумматор Паскаля (1642), арифмометр Лейбница (1673) и прочие полезные изобретения, без которых не было бы компьютеров в том виде, в каком мы сейчас привыкли их видеть...

В 1801 году француз Жозеф Жаккард совершил поистине революционное открытие в области производства ткани — изобрел ткацкий станок для тканей со сложной фактурой, читавший инструкции с карточек. Отверстия, пробитые в карточке, составляли цепь последовательных команд: какие нити использовать и что с ними делать. Изобретение было гениальным, но современники Жаккарда не оценили его по достоинству; сам изобретатель сильно пострадал от французских ткачих, утверждавших, что машина Жаккарда отнимет у них рабочие места.

В 1834 году британский изобретатель Чарльз Бэббидж разработал проект аналитической машины, которая должна производить вычисления, используя ту же идею хранения инструкций на карточках, названных впоследствии перфокартами.

Машина, по сути, представляла собой гигантский арифмометр с программным управлением, памятью и арифметическим устройством и таким образом явилась прообразом первого компьютера. Гениальный ученый долгое время пытался уговорить британское правительство профинансировать его проект, но не добился успеха.

### Контрольные вопросы:

1. Что такое редактирование текста?
2. Что такое форматирование текста?
3. Как включить линейку?
4. Как настроить: междустрочный интервал;  
отступ правой строки (красная строка);  
разреженный интервал (например 4 пт);  
выравнивание и какие виды выравнивания существуют;  
параметры страницы;  
параметры автосохранения;  
текст в рамку/границу, и установить узор к (примеру 20%)  
параметры автозамены.
5. Как вставить нижний и верхний колонтитул?



## Практическая работа № 24

**Тема:** «Внедрение объектов в электронный документ»

**Цель:** научить структурировать данные с помощью таблиц, научить добавлять таблицы в текстовый документ; познакомить с основными приемами работы с таблицами.

### План урока:

1. Организационный момент.
2. Актуализация знаний по теме «Вставка рисунков в текст».
3. Обучающая практическая работа.
4. Самостоятельная работа.
5. Подведение итогов.

### Ход урока

I. Организационный момент.

Постановка темы и цели урока.

II. Актуализация знаний по теме «Вставка рисунков в текст».

Опрос:

Как можно добавить рисунок в текст?

Как изменить размер рисунка, не нарушая пропорций?

Что означает термин «обтекание» рисунка текстом?

Что означает термин «рисунок в тексте»?

III. Обучающая практическая работа.

Таблицы в текстовом процессоре используются для упорядочения и более наглядного представления данных. Таблица Microsoft Word состоит из строк и столбцов. Клетка на пересечении строки столбца называется ячейкой. Ячейки могут содержать числа, текст, рисунки.

Большинство команд работы с таблицами находятся в верхнем меню **Работа с таблицами** (разделы: **Макет** и **Конструктор**), которое появляется на ленте, если курсор расположен внутри таблицы. Некоторые команды продублированы на ленте **Главная** и в контекстном меню.

### Практическая работа

**Задание.** Создайте и оформите таблицу по образцу (рис.1).

Рисунки расположены в папке *Страны*



Рис. 1

### Комментарии.

Рассмотрим создание и редактирование таблиц на примере создания таблицы *Великобритания*. Первое, на что следует обратить внимание, это сколько столбцов и строк лежит в основе таблицы. В нашем случае - три строки и три столбца.

1. На ленте **Вставка** в разделе **Таблицы** выберите команду **Таблица**.

2. Перемещая указатель мыши, выберите три строки и три столбца (рис. 2). Также можно воспользоваться командой **Вставить таблицу** из этого же меню.

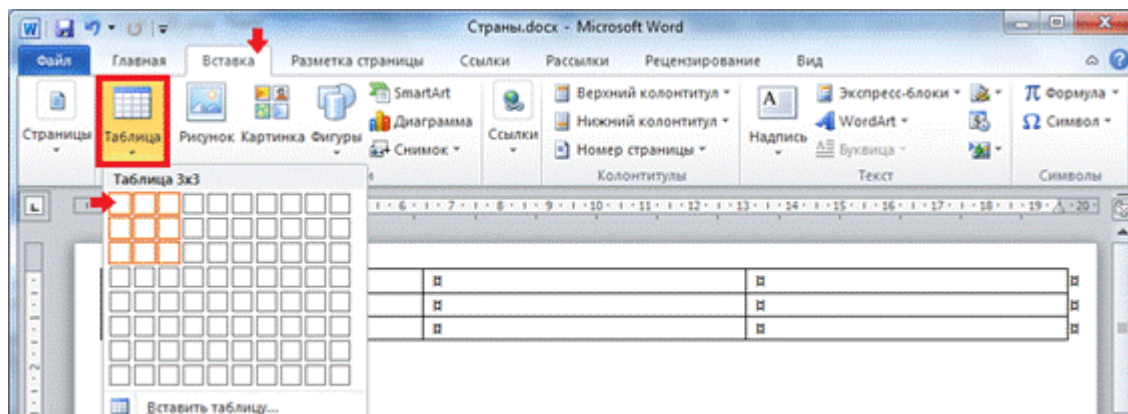


Рис. 2

3. Объедините строчки в первом столбце. Для этого выделите ячейки первого столбца и на ленте **Работа с таблицами: Макет** в разделе **Объединение** выберите команду **Объединить ячейки** (рис. 3).

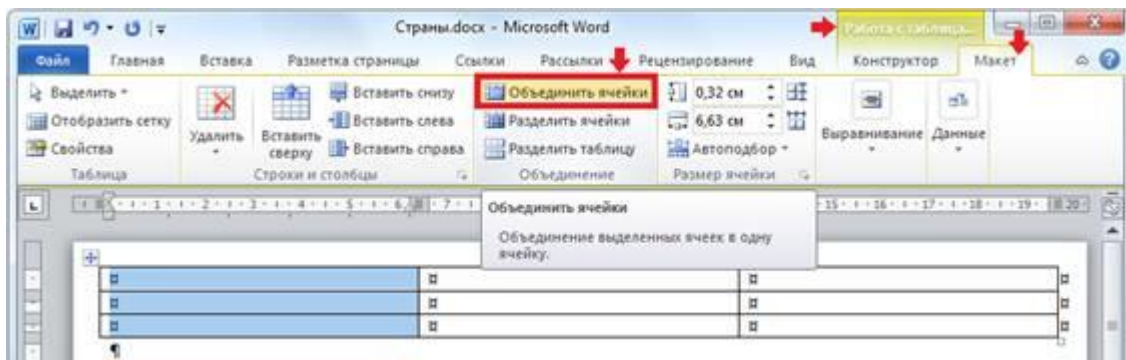


Рис. 3

4. Аналогично объедините вторую и третью строку во втором столбце; а также первые ячейки во втором и третьем столбцах (рис. 4).

5. Заполните верхнюю ячейку таблицы, напечатав по центру *Великобритания* (рис. 4).



Рис. 4

6. Отформатируйте текст в ячейках таблицы по центру как по горизонтали, так и по вертикали: на ленте **Работа с таблицами: Макет** в разделе **Выравнивание** выберите пиктограмму **Выровнять по центру** (рис. 5).

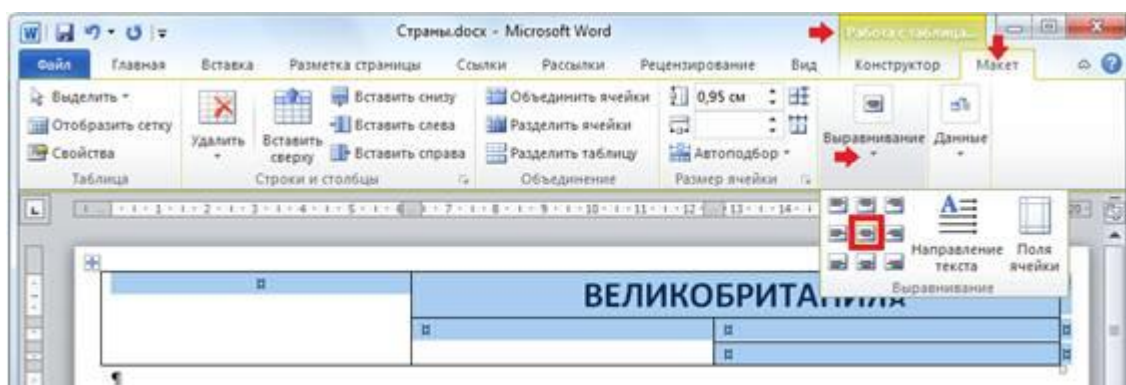


Рис. 5

7. Добавьте в первый столбец рисунок из файла: на ленте **Вставка** в разделе **Иллюстрации** выберите пиктограмму **Рисунок** (рис. 6).

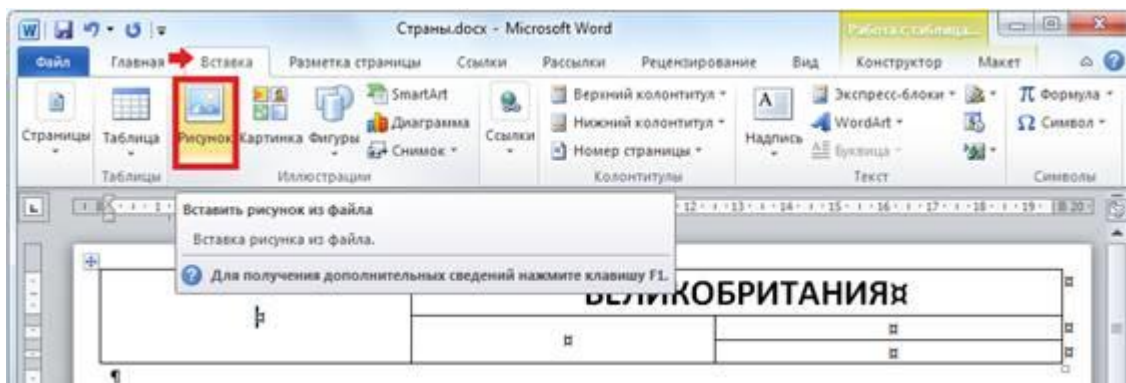


Рис. 6

8. Подкорректируйте ширину первого столбца, сделав ее меньше. Установите указатель мыши на границе между столбцами (рис. 7). Перетащите указатель мыши влево или вправо, чтобы изменить ширину столбца.

Несмотря на то, что высоту строк можно изменить аналогичным образом, старайтесь не использовать этот способ. Изменив высоту «вручную» один раз, вы навсегда теряете возможность автоматического подбора высоты ячейки в зависимости от ее содержимого. Следовательно, при каждом редактировании содержимого ячеек, вам придется следить и регулировать высоту ячейки самому.

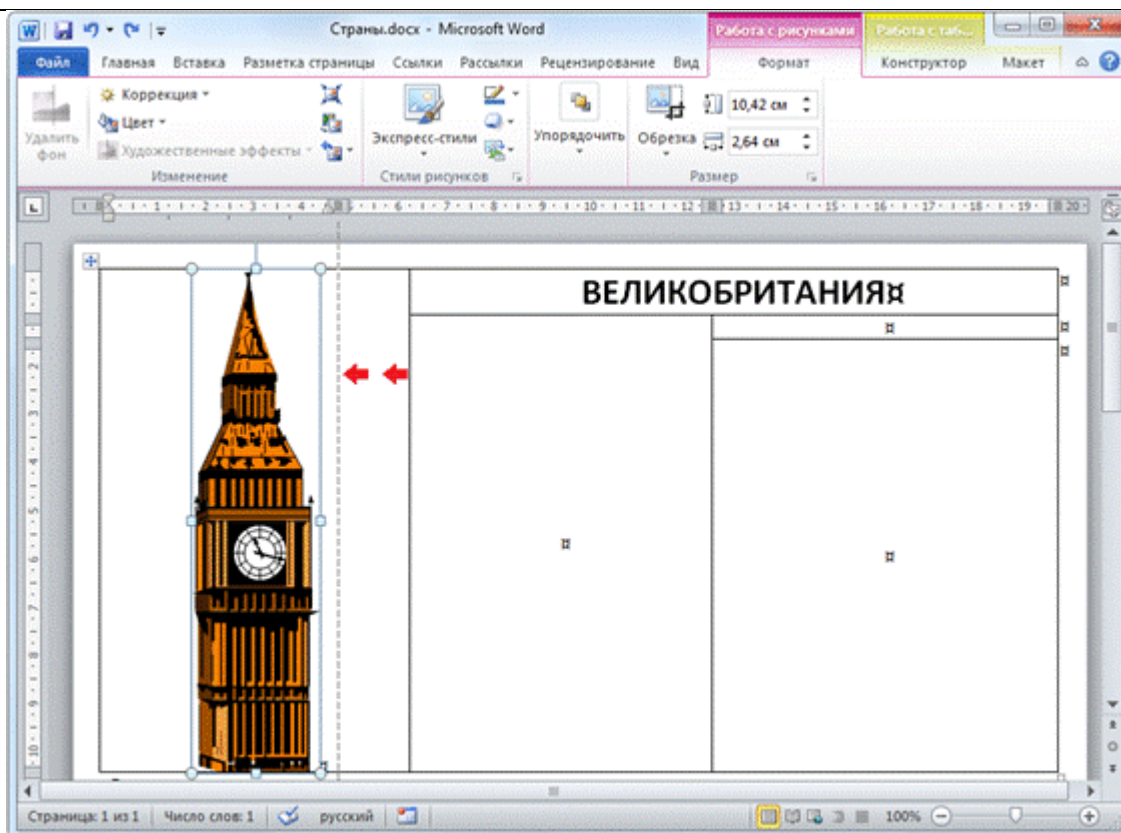


Рис. 7

9. Заполните данными остальные ячейки таблицы (рис. 8).

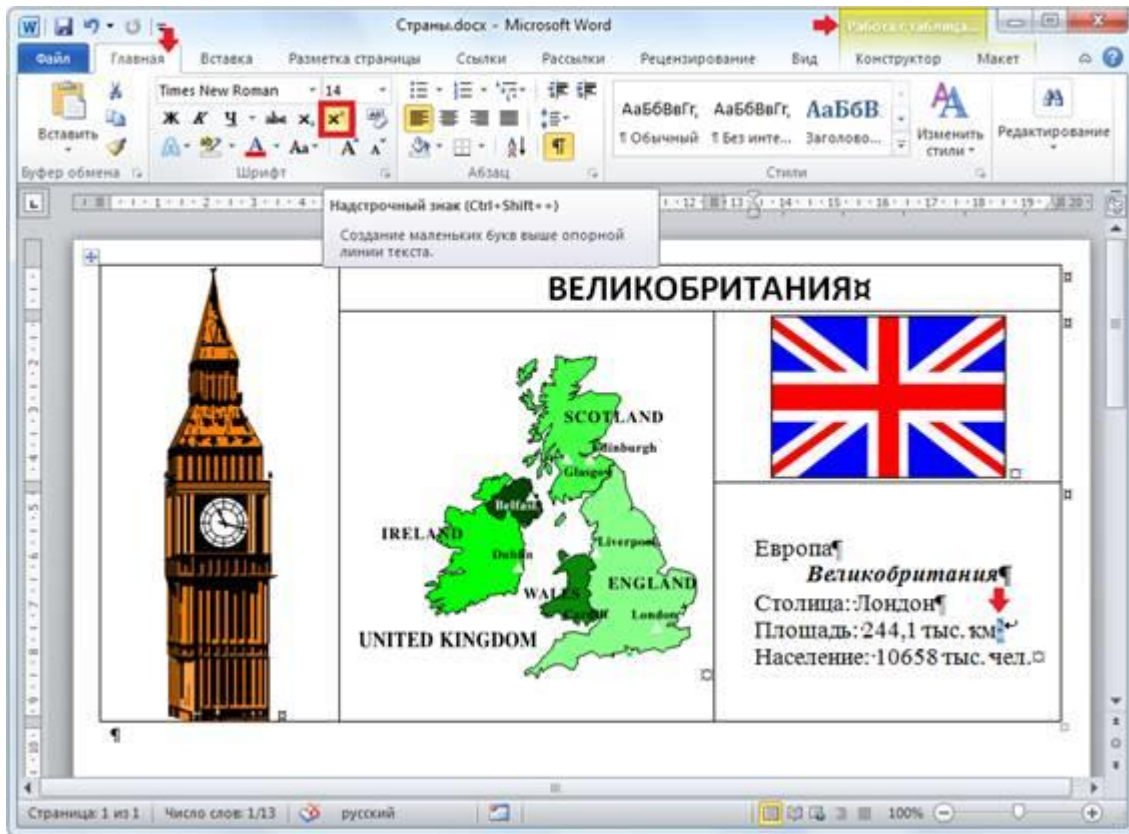


Рис. 8

#### V. Подведение итогов.

Контрольные вопросы:

- Как добавить в текстовый документ таблицу?
- Как объединить несколько ячеек по вертикали или горизонтали?
- Как изменять размер таблицы?
- Как изменить ширину столбцов?
- Почему не рекомендуется изменять высоту строк вручную?
- Какие способ форматирования текста внутри ячеек вы знаете?

## Практическая работа № 25

Тема: «Создание и форматирование таблиц в Excel»

Цель: Получение практических навыков по созданию, редактированию и форматированию таблиц. Формирование навыков работы по оформлению (форматированию) таблиц, построению и редактированию диаграмм в среде MS EXCEL 2007-2013

### Задание 1:

Средствами табличного процессора EXCEL 2007-2013 создайте Таблицу1 на основе ниже приведённого сценария.

1. Запустите табличный процессор EXCEL 2007-2013.
2. Установите курсор в ячейку A1 (щелчком мыши по ячейке) и введите текст: **Выручка от реализации книжной продукции.**
3. Введите таблицу согласно образцу, представленному в таблице1.

Таблица 1

	A1	Выручка от реализации книжной продукции						
	A	B	C	D	E	F	G	H
1	Выручка от реализации книжной продукции							
2	Осень 2014							
3	Объём продаж	июнь	Июль	август	сентябрь	Октябрь	Итого	Процент реализации
4	До 100 пачек	1050	2100	4250	2250	1950		
5	До 200 пачек	330	6160	13530	11200	1100		
6	> 200 пачек		1000	6000	4000			
7	Сумма							
8								

4. Рассчитайте сумму выручки от реализации книжной продукции в июне месяце одним из двух способов:

- 1 способ: установите курсор в ячейку B7, введите в ячейку B7 формулу (формула начинается со знака =, занесение имён ячеек в формулу осуществляем щелчками мыши по соответствующим ячейкам): =B4+B5+B6. Нажмите клавишу **Enter**. В ячейке появится сумма диапазона ячеек B4:B6;


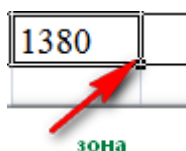
- 2 способ: выделите диапазон ячеек B4:B7 и нажмите кнопку Автосумма  (Вкладка Главная, Группа Редактирование). В ячейке B7 появится сумма диапазона ячеек B4:B6. Сравните полученный результат со значением ячейки B7 в таблице 2.

Таблица 2

	A	B	C	D	E	F	G	H
1	Выручка от реализации книжной продукции							
2	Осень 2014							
3	Объём продаж	июнь	Июль	август	сентябрь	Октябрь	Итого	Процент реализации
4	До 100 пачек	1050	2100	4250	2250	1950		
5	До 200 пачек	330	6160	13530	11200	1100		
6	> 200 пачек		1000	6000	4000			
7	Сумма	1380	9260	23780	17450	3050		
8								

5. Распространите операцию суммирования на диапазон **C7:F7**:

- Скопируйте содержимое ячейки **B7** в ячейки **C7:F7**. Для этого схватите ячейку **B7** за




правый нижний угол (зону автозаполнения) **автозаполнения** и не отпуская кнопку мыши перетащите её в ячейку **F7**. Очистите диапазон **C7:F7**.

6. Убедитесь в правильности выполненной операции:

- выделите ячейку **B7**. В строке формул должно отобразиться выражение: **=СУММ(B4:B6)**;


7. Подсчитайте суммарную выручку от реализации книжной продукции (столбец **Итого**). Для этого:

- выделите в строке ячейки **B4:G4**;
- нажмите кнопку  ;
- в ячейке **G4** появится сумма диапазона **B4:F4**.

8. Подсчитайте суммы в остальных ячейках столбца **Итого**. Для этого: схватите ячейку **G4** за правый нижний угол (зону автозаполнения) и, не отпуская кнопку мыши, протащите её до ячейки **G7**. В ячейках **G5**, **G6**, **G7** появятся суммарная выручка от реализации книжной продукции.

9. Подсчитайте **Процент реализации**, полученной от продажи партий товара. Для этого:

- выделите ячейку **H4**;
- введите формулу **=G4/G7**;
- нажмите клавишу **Enter**;
- выделите ячейку **H4**;

- нажмите кнопку  (Вкладка Главная, Группа Число). В ячейке Н4 появится значение доли выручки в процентах.

10. *Рассчитайте долю выручки для других строк таблицы, используя автозаполнение.*

В результате автозаполнения в ячейках Н5, Н6 и Н7 появится сообщение #ДЕЛ/0! (деление на ноль). Такой результат связан с тем, что в знаменателе формулы введён **относительный** адрес ячейки, который в результате копирования будет смещаться относительно ячейки G7 (G8, G9, G10 — пустые ячейки). Измените **относительный** адрес ячейки G7 на **абсолютный** — \$G\$7, это приведёт к получению правильного результата счёта. Еще раз попробуйте рассчитать доли выручки в процентах.

Для этого:

- очистите диапазон Н4:Н7;
- выделите ячейку Н4;
- введите формулу =G4/\$G\$7;
- нажмите клавишу **Enter**;
- рассчитайте долю выручки для других строк таблицы, используя автозаполнение.

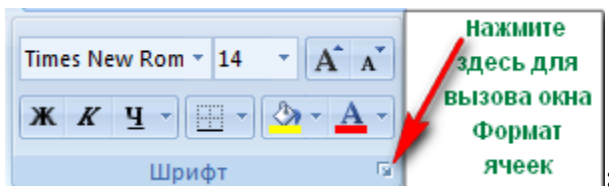
В результате в ячейках диапазона Н4:Н7 появится доля выручки в процентах.

### Задание 2:

Средствами табличного процессора MS Excel измените оформление таблицы, размещенной в файле и постройте диаграмму на основе данных указанной таблицы. Последовательность Ваших действий описана в ниже приведённом сценарии.

1. Измените шрифт заголовка таблицы. Для этого:

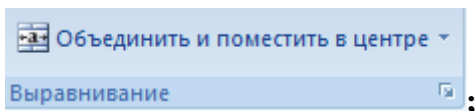
- выделите заголовок таблицы;
- выберите **Вкладка Главная, группа Шрифт**



- установите параметры: тип шрифта (например, **Arial Cyr**), стиль (например, **полужирный**), размер (например **16**), цвет и т. д.;

2. Задайте выравнивание заголовка по **центру**. Для этого:

- выделите диапазон ячеек, в пределах которых будет размещён заголовок (обычно заголовок центрируется по ширине таблицы);
- выберите **Вкладка Главная, группа Выравнивание**





- нажмите кнопку **Объединить и поместить в центре**.

3. Измените цвет фона шапки таблицы (совокупность наименований столбцов):

- выделите диапазон ячеек **A3:H3**;
- выберите **Вкладка Главная, группа Шрифт** (используйте кнопку **Цвет**

**заливки**  для изменения цвета фона).

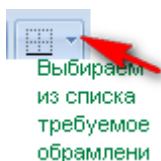
4. Измените цвет текста шапки таблицы с помощью кнопки **Цвет текста** 

5. Уберите линии сетки. Для этого:

6. Нажмите **Вкладку/Вид**, снимаем галочку с пункта **Сетка**

7. Выполните процедуру обрамления таблицы (границы). Для этого:

- выделите нужные строки (например, всю таблицу);



- выберите **Вкладка Главная, группа Шрифт** ;
- установите параметры: границы (например, все границы), вид линии (толщина линии), цвет линии (из палитры).

8. Измените высоту строки: Для этого:

- установите указатель мыши на границу двух любых строк (в зоне заголовков строк),



указатель мыши превратится в двунаправленную стрелку, нажмите левую клавишу мыши и, не отпуская её, протащите границу вверх для уменьшения высоты строки и вниз для увеличения высоты строки, отпустите мышь.

9. На основе данных электронной таблицы **Выручка от реализации книжной продукции** постройте диаграмму. Для этого:

- выделите диапазон ячеек **A3:F6**;
- выберите **Вкладка/Вставка/группа Диаграммы** нажмите кнопку **Гистограмма**.

10. Ведите заголовок диаграммы. Например: **Выручка от реализации книжной продукции**.

11. Отредактируйте полученную диаграмму, которая состоит из:

- области диаграммы;
- области построения;
- области рядов данных (пункт 16);

- легенды (пункт 17);
- заголовка.

12. Отредактируйте область всей диаграммы. Для этого:

- выделите область диаграммы щелчком мыши по ней. Обратите внимание на изменённый вид **Ленты**. Вам стала доступна **Вкладка Работа с диаграммами** и дополнительная **Вкладка Конструктор**.

13. Отредактируйте **область построения**. Для этого:

- выделите область построения щелчком мыши;
- перейдите на **Вкладку Макет**;
- нажмите на **Кнопку Область построения**, выберите **Дополнительные параметры построения**, переберите предложенные варианты оформления **Области построения**.

14. Отредактируйте **область рядов** данных. Для этого:

- выделите один из столбцов щелчком мыши по нему. Одновременно будут выделены все остальные столбцы, относящиеся к одному ряду;
- вызовите контекстное меню области ряда данных, щелкнув правой кнопкой мыши по выделенному столбцу;
- окрасьте столбцы (залейте) одного ряда в выбранный Вами цвет и измените тип границы столбца;
- аналогичным образом окрасьте все остальные группы столбцов.

15. Отредактируйте **легенду**. Для этого:

- выделите легенду, в которой приведены обозначения, используемые для выделения различных групп столбцов;
- измените шрифт и цвет фона в легенде.

16. Отредактируйте заголовок. Для этого:

- Нажмите на **Кнопку Название диаграммы**, выберите способ размещения **Над диаграммой**;
- щелкните мышью по строке формул; наберите новый текст заголовка в строке формул и нажмите клавишу **Enter**; текст появится внутри области диаграммы;
- укажите на рамку, окружающую появившийся текст заголовка, нажмите мышью и перетащите заголовок на нужное место;
- измените цвет текста с помощью кнопки **Цвет шрифта**.

17. Поместите диаграмму рядом с таблицей (справа или внизу). Для этого:

- щёлкните мышью по области диаграммы и, не отпуская кнопки мыши, перетащите её в нужное место.

18. Постройте объемную круговую диаграмму. Для этого:

- выделите в электронной таблице область **A3:F4**;
- выберите **Вкладка Вставка, группа Диаграммы** нажмите кнопку **Круговая,**

#### **Объёмная круговая;**

- щелчком мыши выделите область рядов данных диаграммы, вызовите контекстное меню, выберите **Поворот объёмной фигуры** и поверните объёмную диаграмму на нужный угол;

- выделите область рядов данных диаграммы. С помощью контекстного меню окрасьте всю диаграмму в выбранный Вами цвет;

- выделите передний сектор диаграммы установите цвет точки данных. Например: красный;

- выделите передний сектор диаграммы и, удерживая нажатой левую клавишу мыши, выдвиньте этот сектор вперед, вырезав его из круга.

19. Сохраните созданную таблицу в папке **Отчеты**.

20. Ответьте на **Контрольные вопросы по данной работе**.

## Практическая работа № 26

Тема: «Построение графиков и диаграмм»

Цель: Формирование навыков работы по оформлению (форматированию) таблиц, построению и редактированию диаграмм в среде MS EXCEL 2007-2013

### Методические указания

#### Работа с диаграммами

#### Основные термины

Диаграмма представляет собой графический способ представления табличных данных. При этом используется следующая терминология.

- **Серия данных** – группа данных расположенная внутри одной строки таблицы.
- **Имя серии** – имя строки таблицы, содержимое которой образует данную серию.
- **Легенда** – набор всех имен серий данной таблицы.
- **Категория** – группа значений, расположенных в одном столбце таблицы.

#### Элементы диаграммы

Для осознанной работы с диаграммами необходимо знать название и назначение ее основных элементов.



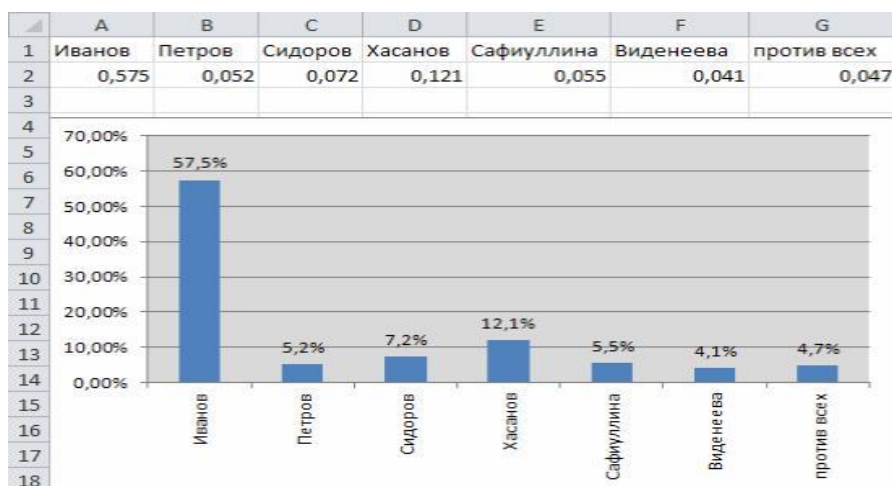
1. **Область диаграммы.** Область размещения диаграммы и всех ее элементов.
2. **Область построения диаграммы.** На плоских диаграммах – ограниченная осями область, содержащая все ряды данных. На объемных диаграммах – ограниченная осями область, содержащая все ряды данных, имена категорий, подписи делений и названия осей.
3. **Точки данных для ряда данных,** отмеченные на диаграмме. **Точки дан-**

**ных** – отдельные значения, отображаемые на диаграмме в виде полос, столбцов, линий, секторов, точек или других объектов, называемых маркерами данных. Маркеры данных одного цвета образуют ряд данных. **Ряд данных** – Набор связанных между собой элементов данных, отображаемых на диаграмме. Каждому ряду данных на диаграмме соответствует отдельный цвет или способ обозначения, указанный на легенде диаграммы. Диаграммы всех типов, кроме круговой, могут содержать несколько рядов данных.

4. **Ось категорий.** Горизонтальная ось, вдоль нее строятся категории.
5. **Ось значений.** Вертикальная ось, вдоль нее строятся данные.
6. **Легенда диаграммы.** Рамка, в которой определяются узоры или цвета рядов или категорий данных на диаграмме.
7. **Названия на диаграммах.** Описательный текст, располагающийся вдоль осей или по центру в верхней части диаграммы.
8. **Подписи данных.** Подпись с дополнительными сведениями о маркере данных, предоставляющем одну точку данных или значение ячейки листа.

## Создание диаграмм

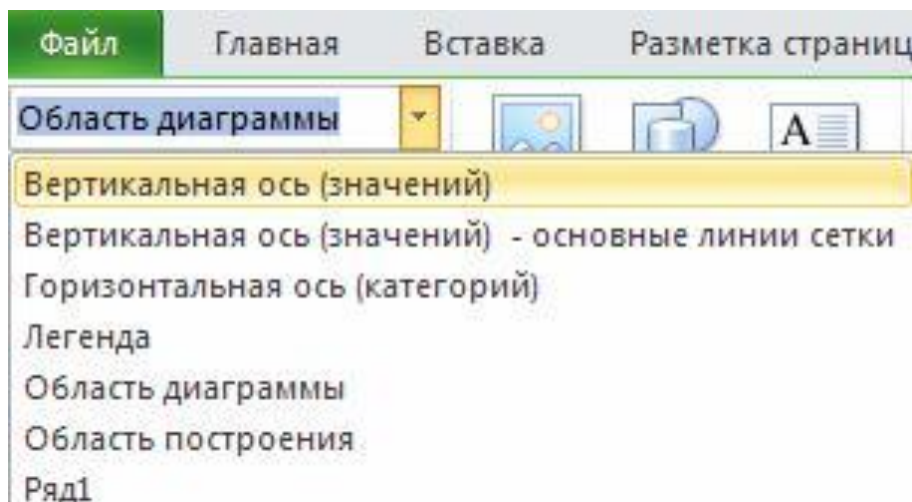
В Microsoft Excel 2010 можно построить диаграмму следующим способом: необходимо предварительно создать в документе таблицу данных, выделить ее, и после этого запускать вставку диаграмм. Диаграмма связана с данными, на основе которых она создана, и обновляется автоматически при изменении данных. Диаграмму можно создать на отдельном листе или поместить в качестве внедренного объекта на лист с данными. Создадим таблицу и диаграмму следующего вида.



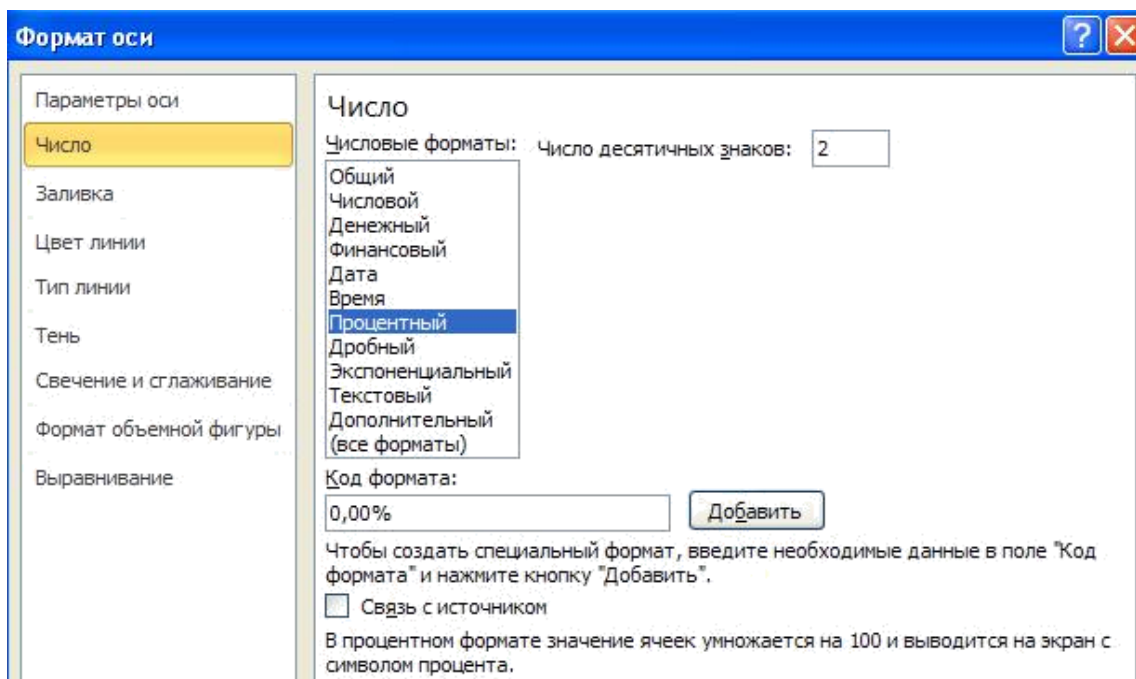
1. Создайте новую книгу. Для этого нажмите на кнопку **Файл**, выполните команду: **Создать/ Новая книга** и нажмите **Создать**.
2. Заполните таблицу, т.е. ячейки **A1-G1, A2-G2**.
3. Выделите таблицу и выполните команду **Вставка/(Диаграммы) Гистограмма**.
4. В списке гистограмм выбираем пункт **Гистограмма с группировкой**.



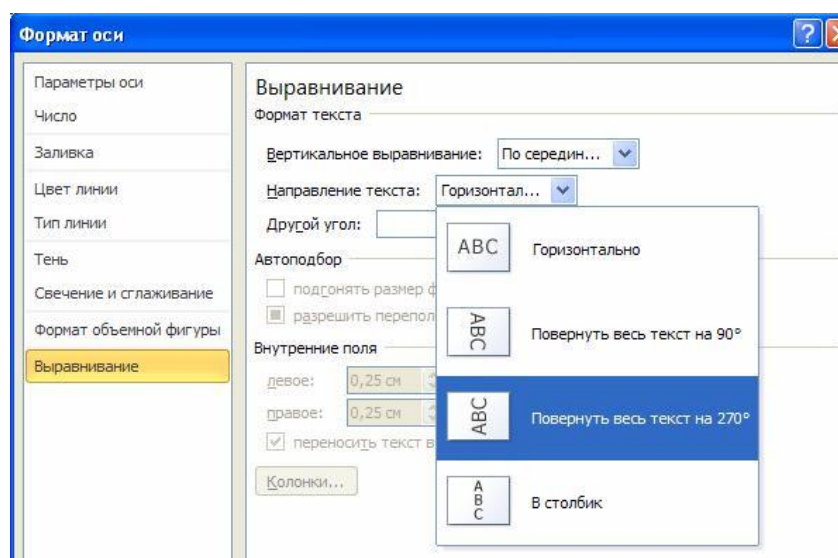
5. Выделите диаграмму, затем выполните команду: **Макет/(Текущий фрагмент) Область диаграммы/Вертикальная ось (значений)**.



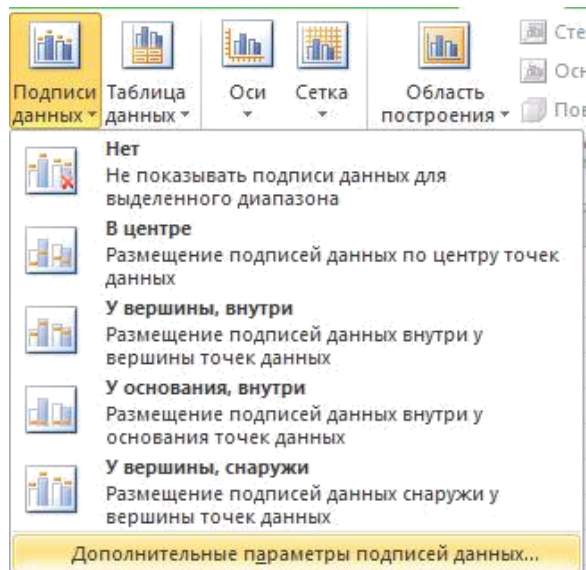
6. Далее выполните команду: **Макет/(Текущий фрагмент)Формат выделенного объекта**.
7. В появившемся окне **Формат оси** выберем вкладку **Число**.
8. Далее выбираем **Числовые форматы: Процентный, число десятичных знаков: 2**. Нажмите **Заккрыть**.



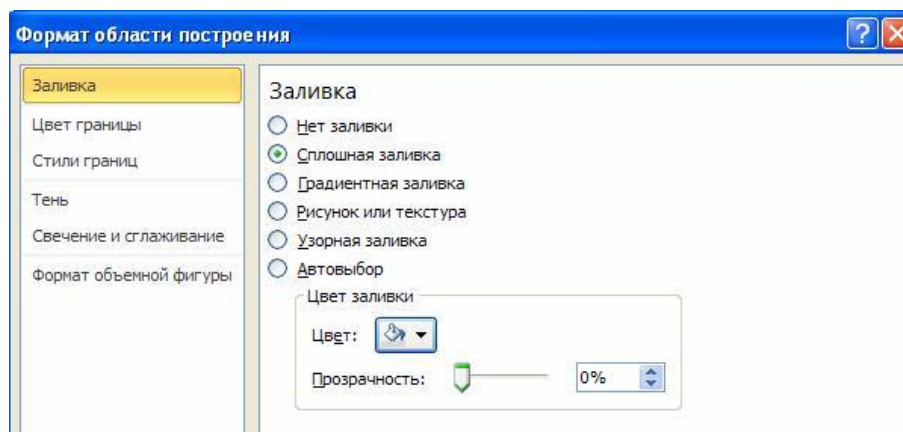
9. Выбираем **Горизонтальную ось (категорий)**, аналогично с пунктом 5.
10. Далее выполните команду: **Макет/(Текущий фрагмент)Формат выделенного объекта**.
11. В появившемся окне **Формат оси** выберем вкладку **Выравнивание**. В строке **Направление текста** в выпадающем списке выбираем пункт **По-вернуть весь текст на 270**. Нажимаем **Заккрыть**.



12. Выбираем **Ряд1**, аналогично с пунктом 5.
13. Далее выполните команду: **Макет/(Подписи) Подписи данных**.



14. В выпадающем списке выберите пункт **Дополнительные параметры подписей данных**.
15. В открывшемся окне, по аналогии с предыдущими пунктами, выбираем также **Числовой формат: процентный, число десятичных знаков: 1**.
16. Во вкладке **Параметры подписи** установите следующие параметры: **Включить в подписи: значения** и **Положение подписи: у вершины, снаружи** Нажимаем **Заккрыть**.
17. Аналогично предыдущим пунктам 5 и 6 выберите **Область построения**. В открывшемся окне **Формат области построения** выбираем: **Залив-ка/сплошная заливка**. В строке **Цвет** выбираем нужный нам цвет. Нажи-маем **Заккрыть**.



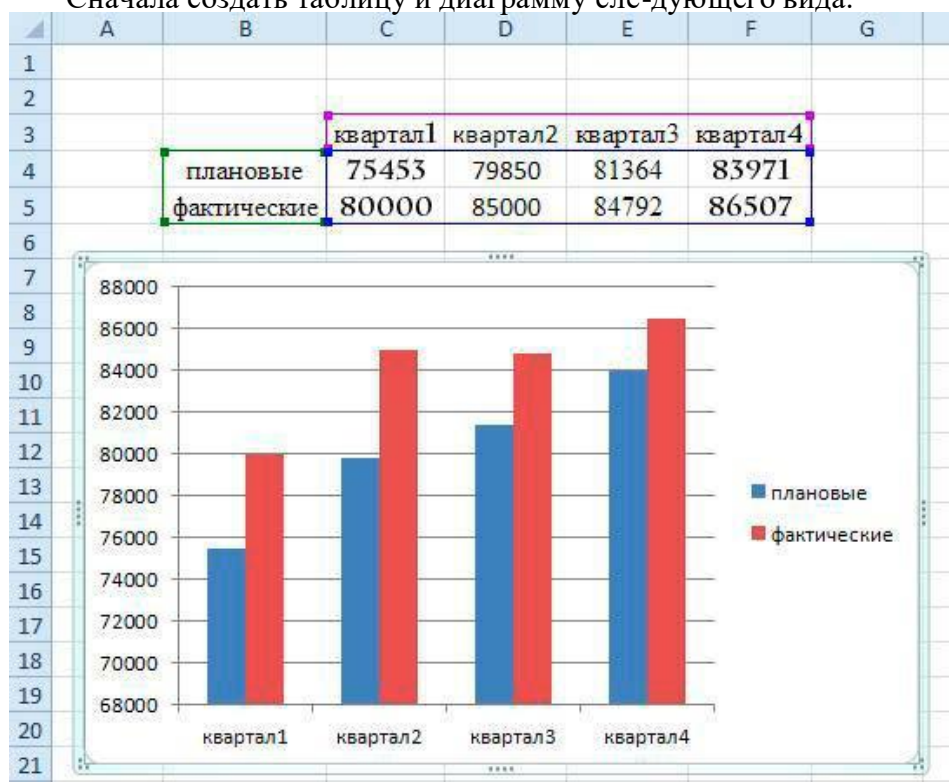
18. При построении диаграммы справа у нас есть надпись: **Ряд1**. Это **Легенда**, уберем ее выполнив команду: **Макет/(подписи)Легенда/Нет**.
19. Расположите диаграмму под таблицей и установите ей соответствующий размер.

### *Использование рисунков в диаграммах*

### **Создание таблицы данных и диаграммы**



С помощью меню **Диаграмма** можно корректировать и изменять любые параметры и свойства диаграмм: изменять тип диаграмм, задавать цвета и стили линий диаграммы, форматировать диаграмму, изменять формат построения диаграммы и т.д. С помощью мастера диаграмм можно также строить графики различных функций. Сначала создать таблицу и диаграмму следующего вида.



Создадим таблицу данных.

	A	B	C	D	E	F
1						
2						
3			квартал1	квартал2	квартал3	квартал4
4		плановые	75453	79850	81364	83971
5		фактические	80000	85000	84792	86507

1. Поместите курсор мыши в рабочую область программы. Наберите в ячейке **C3** имя поля: **квартал1**.
2. Выделите ячейку **C3**, далее нажмите на черный квадратик, находящийся в правом нижнем углу ячейки, затем, не отпуская, кнопку мыши перетащите курсор до ячейки **F3**. У вас должно получиться следующее:

C	D	E	F	G
квартал1	квартал2	квартал3	квартал4	

3. В ячейки **B4** и **B5** введите имена полей: **плановые** и **фактические** соответственно.
4. В диапазон ячеек C4:F5 ввести числовые данные.

	A	B	C	D	E	F
1						
2						
3			квартал1	квартал2	квартал3	квартал4
4		плановые				
5		фактические				

	A	B	C	D	E	F
1						
2						
3			квартал1	квартал2	квартал3	квартал4
4		плановые	75453	79850	81364	83971
5		фактические	80000	85000	84792	86507

5. Выделите ячейки **B3:F5** и установите шрифт **Charlemagne** 12 пунктов, а затем выровняйте по центру. В итоге мы получим исходную таблицу.

4. Выделить введенные данные, далее выполните команду **Вставка/(Диаграммы) Гистограмма/Гистограмма с группировкой**.

Изменение типа диаграммы

Книга2 -

Файл Главная Вставка Разметка страницы Формулы Данные Рецензирование Ви

Сводная Таблица Таблицы Рисунок Картинка Иллюстрации Фигуры SmartArt Снимок

Гистограмма График С областями Точечная Другие

Круговая Линейчатая

Гистограмма

Гистограмма с группировкой

Объем

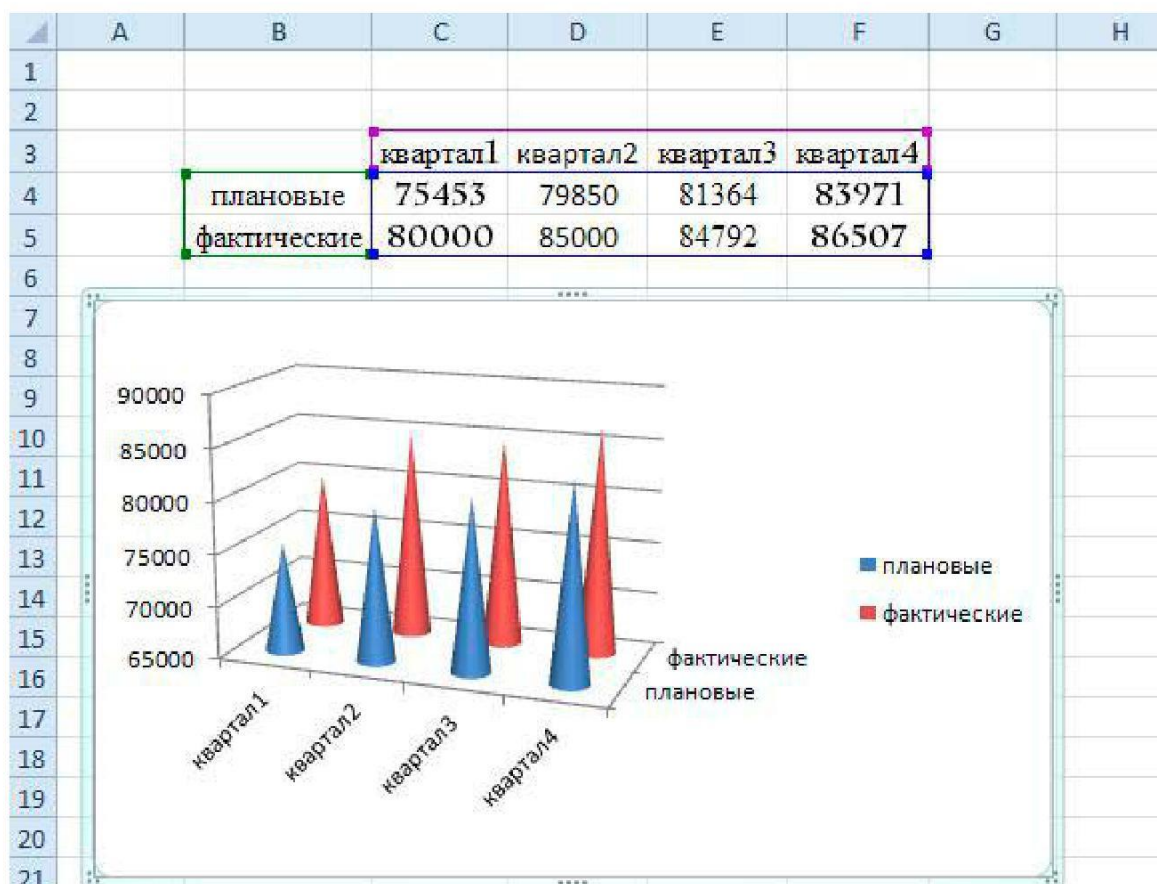
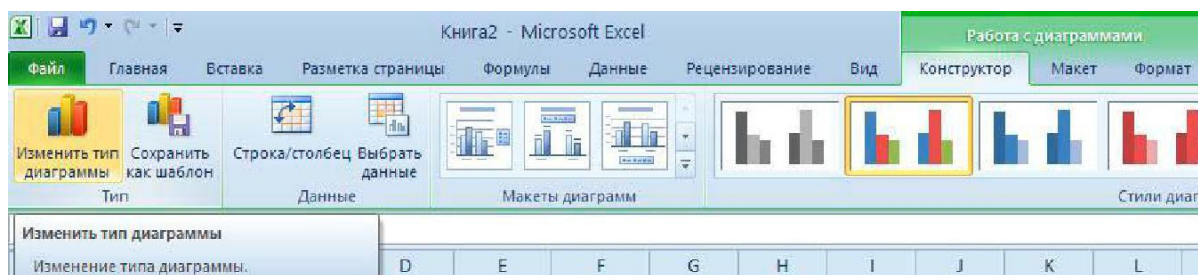
Сравнение значений по категориям с использованием вертикальных прямоугольников.

Применяется в тех случаях, когда порядок категорий не важен или требуется отобразить количества элементов.

Цилинд

	A	B	C	D
1				
2				
3			квартал1	квартал
4		плановые	75453	79850
5		фактические	80000	85000

1. Для того чтобы изменить тип диаграммы, щелкните на диаграмму.
2. Выполните команду **Конструктор / (Тип) Изменить тип диаграммы / Объемная коническая**.



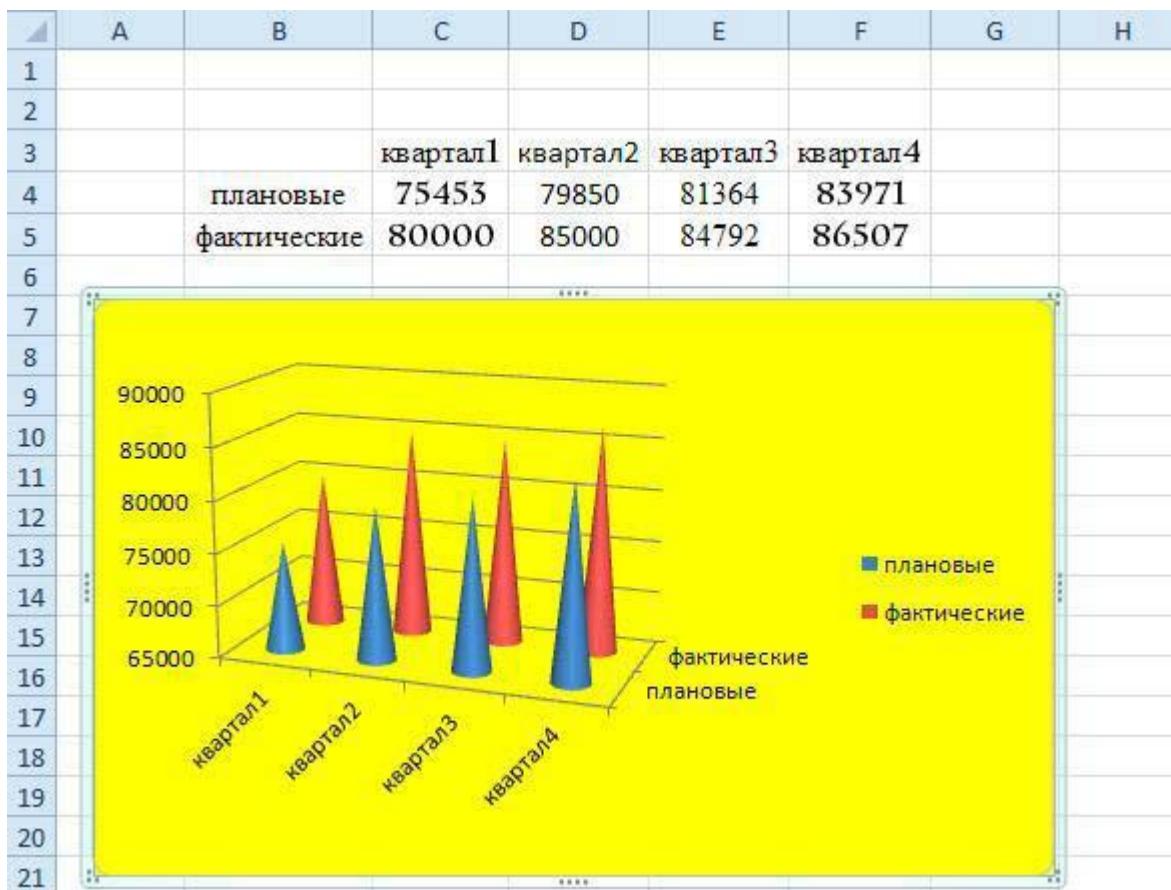
Выбор цвета, границ и заливки допускается в маркерах данных, в области диаграммы, в области построения и в других элементах диаграммы.

Следующая процедура используется для изменения цветов, узора, ширины линии или типа рамки для маркеров данных, области диаграммы, области построения, сетки, осей и подписей делений на плоских и объемных диаграммах:

1. Щелкните левой кнопкой мыши на область диаграммы.

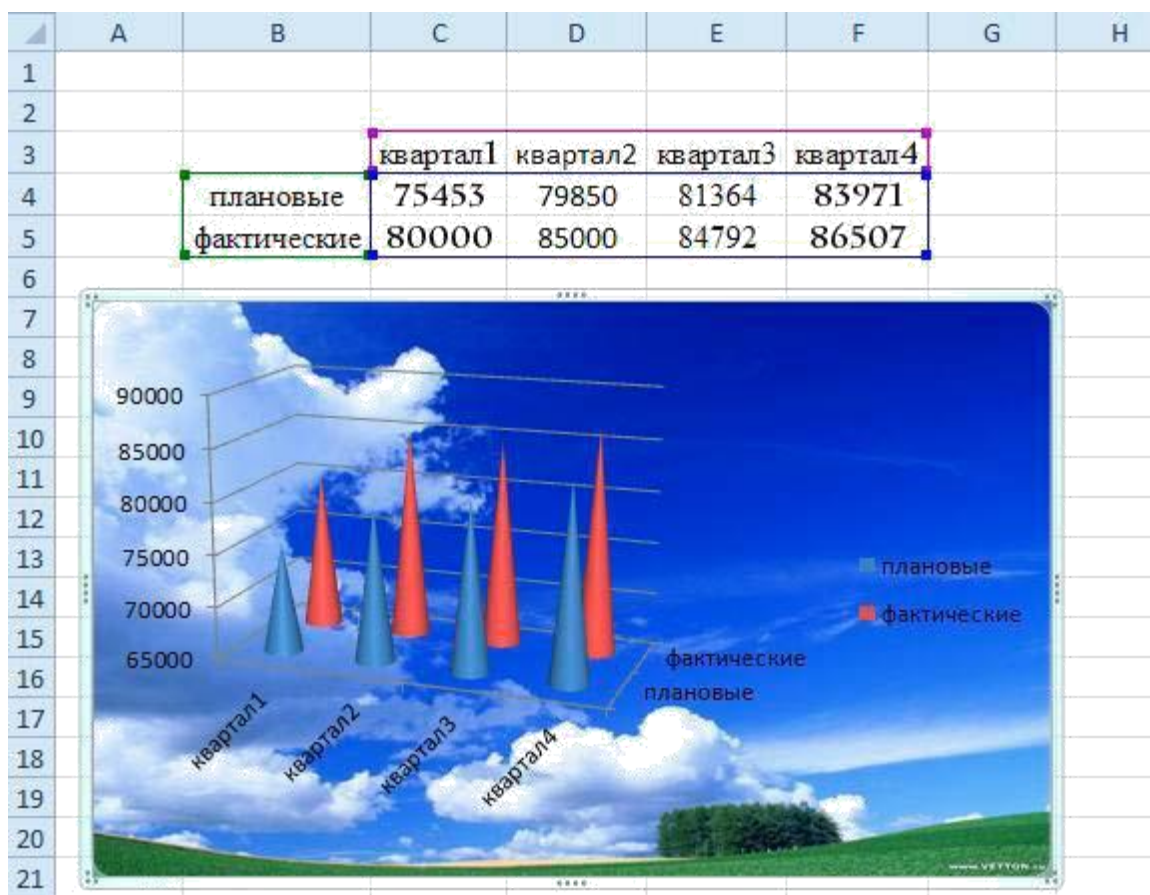
2. При необходимости заливки фигуры выберите **Формат/(Стили фигур)Заливка фигуры** и укажите нужный вид во вкладках **Градиентная** или **Текстура**. (В нашем примере **Формат/(Стили фигур)Заливка фигуры/Желтый цвет**)

3. При необходимости изменения вида диаграммы пользуйтесь вкладками **Конструктор, Макет** или **Формат**.



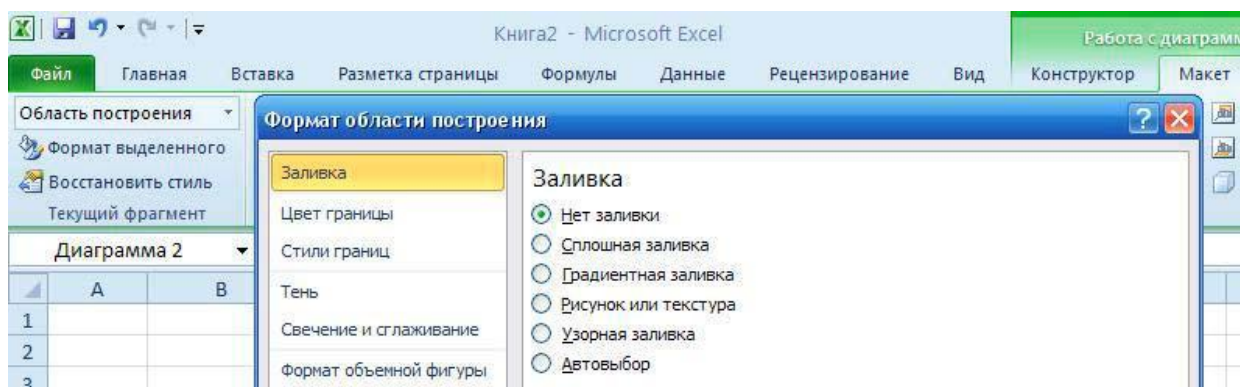
### Использование рисунков в диаграммах.

Следующая процедура используется для добавления рисунка — например, точечного — к определенным типам маркеров данных, области диаграммы, области построения или легенде на плоских и объемных диаграммах.



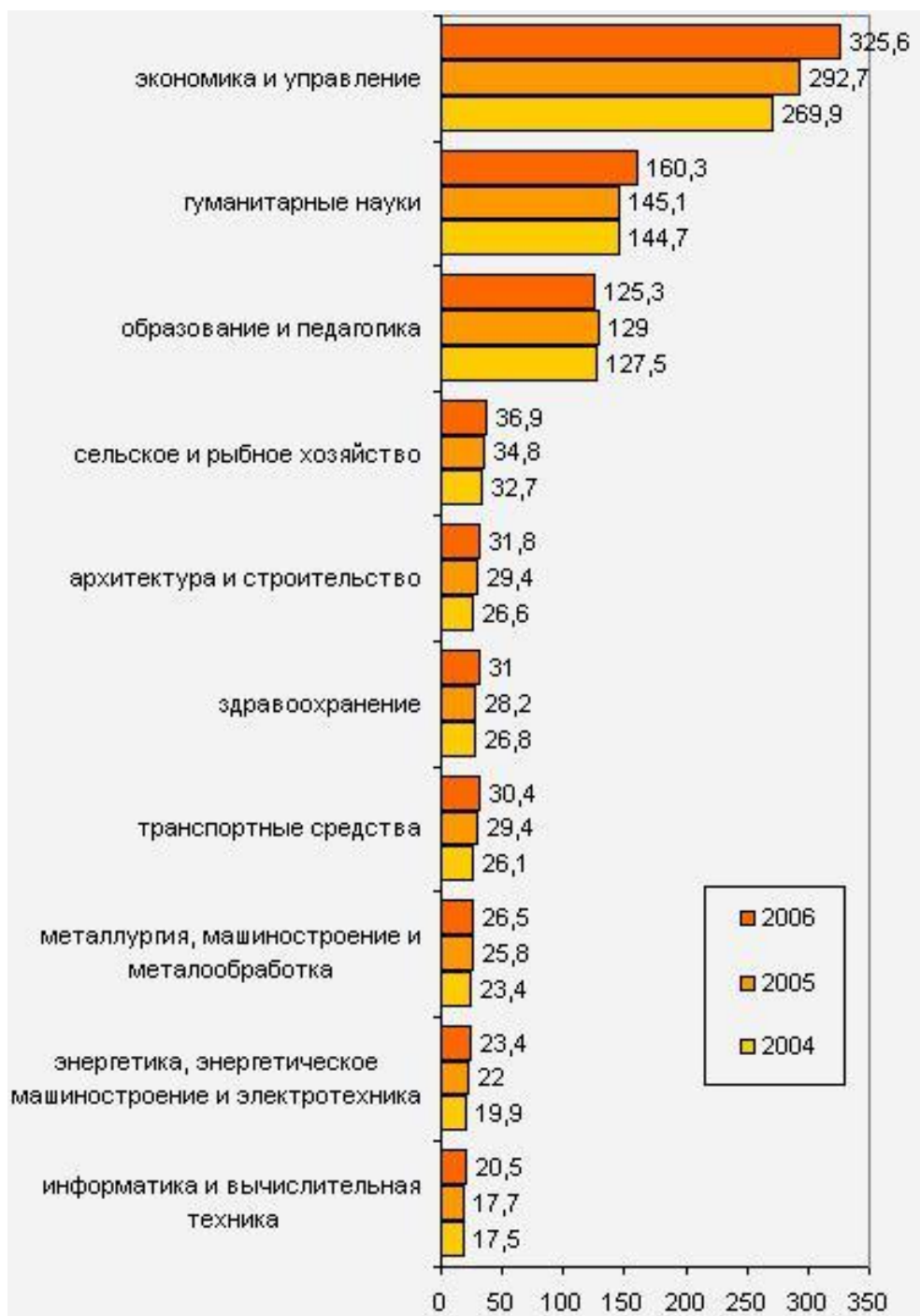
Для того чтобы у вас получилось также, выполните следующие команды.

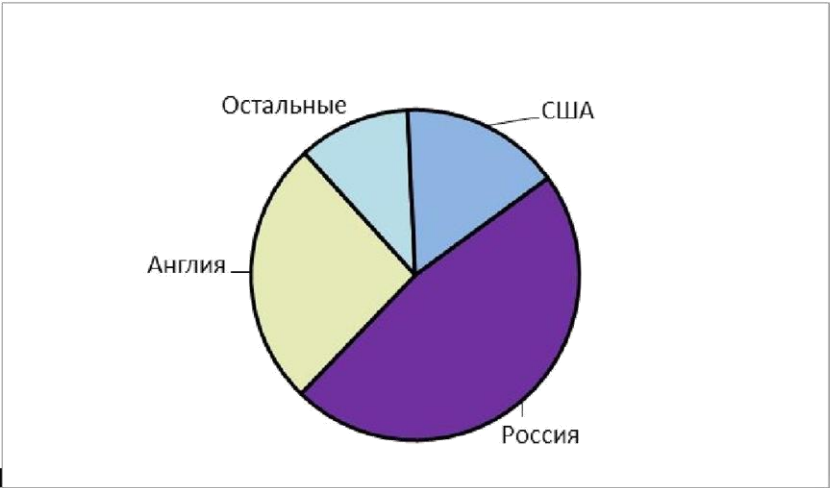
1. Щелкните левой кнопкой мыши на область диаграммы.
2. Выполните команду **Формат/(Стили фигур) Заливка фигуры/Рисунок**.
3. В появившемся окне **Вставка рисунка** выберите папку, где находится нужный рисунок, а затем дважды щелкните его.
4. Далее вы полните команду **Макет/Текущий фрагмент**. Выберите **Область построения**. Далее **Формат выделенного фрагмента/Нет заливки**.



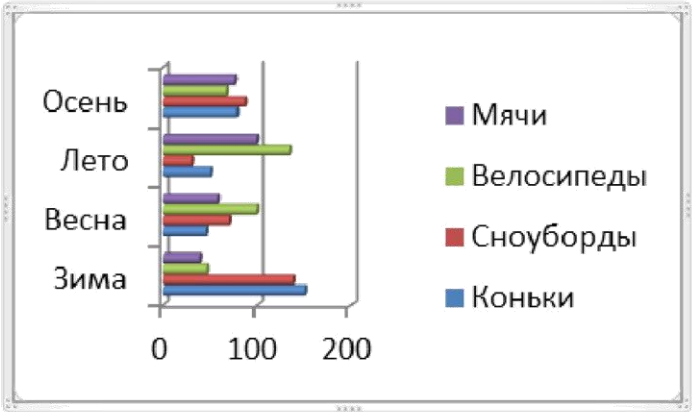
## Задания для самостоятельной работы

1. Создайте три диаграммы.





**Объем продаж в спортивном магазине**



## Лабораторная работа № 27

### Тема: «Создание файла баз данных»

#### Методические указания

##### А) Базы данных

Хранение информации - одна из важнейших функций компьютера. Одним из распространенных средств хранения являются базы данных. *База данных* - файл специального формата, содержащий информацию, структурированную заданным образом. Большинство баз данных имеет *табличную структуру*. В базах данных столбцы называются *полями*, а строки - *записями*.

Поля - это основные элементы структуры базы данных. Они обладают *свойствами*, от которых зависит, какие типы данных можно вносить в поле, а какие - нет, а также что можно делать с данными, содержащимися в поле.

Очевидным уникальным свойством любого поля является его имя. Разумеется, одна база данных не может иметь двух полей с одинаковым именем, поскольку компьютер запутается в их содержимом.

##### 1) СУБД MS Access

Системы управления базами данных (СУБД) - это программные средства, с помощью которых можно создавать базы данных, наполнять их и работать с ними. С помощью MS Access обычные пользователи получили удобное средство для создания и эксплуатации достаточно мощных баз данных без необходимости что-либо программировать.

Исходное окно Access отличается простотой и лаконичностью. Шесть вкладок этого окна представляют шесть объектов, с которыми работает программа.

Таблицы - это основные объекты базы данных. В них хранятся данные.

Запросы - это специальные структуры, предназначенные для обработки данных базы. С помощью запросов данные упорядочивают, фильтруют, отбирают, изменяют, объединяют, то есть обрабатывают.

Формы - это объекты, с помощью которых в базу вводят новые данные или просматривают имеющиеся.

Отчеты - это формы «наоборот». С их помощью данные выдают на принтер в удобном и наглядном виде.

С организационной точки зрения в работе с любой базой данных есть два режима: проектировочный и эксплуатационный (пользовательский). Создатель базы имеет право создавать в ней новые объекты (например, таблицы), задавать их структуру,



менять свойства полей, устанавливая необходимые связи. Он работает со структурой базы и имеет полный доступ к базе.

### **Создание таблиц**

Создание базы начинается с создания первой таблицы. Есть несколько способов создания новой таблицы, отличающихся уровнем автоматизации. Самый «автоматичный» способ состоит в импорте таблиц из другой базы. В зависимости от обстоятельств из импортируемой таблицы может поступить структура полей, их названия и свойства, а также и содержание базы. Если что-то импортируется не совсем так, как надо, необходимые правки (например, в свойства полей) вносят вручную.

Опытные разработчики пользуются Мастером таблиц. Это программа, ускоряющая создание структуры таблицы. Мастер задает ряд вопросов и, руководствуясь полученными ответами, создает структуру таблицы

автоматически. Несмотря на то, что этот режим служит для упрощения работы, начинающим пользоваться им не рекомендуется, поскольку, не владея всей терминологией, легко запутаться в вопросах и ответах. Первые таблицы стоит попробовать создать вручную.

Наиболее универсальный ручной метод предоставляет пункт Конструктор. В этом режиме можно самостоятельно задать имена полей, выбрать их тип и настроить свойства.

### **Практические задания**

Используя программу MSAccess, создать базу данных «Правонарушения».

Пусть известны следующие группы данных.

Сведения о правонарушителях:

1) фамилия, имя, отчество;

2) пол;

3) дата рождения;

4) дата совершения преступления;

5) род занятий на момент совершения преступления (рабочие, работники сельского хозяйства, служащие, учащиеся; лица без постоянного источника дохода (в том числе безработные));

б) дата вынесения приговора.

Вид преступлений:

1) убийство и покушение на убийство;

2) умышленное причинение тяжкого вреда здоровью;

- 3) изнасилование и покушение на изнасилование;
- 4) грабеж;
- 5) разбой;
- 6) кража;
- 7) преступления, связанные с незаконным оборотом наркотиков;
- 8) хулиганство;
- 9) нарушения правил дорожного движения и эксплуатации транспортных средств.

Меры наказания, назначенные судами:

- 1) лишение свободы;
- 2) исправительные работы без лишения свободы;
- 3) условное осуждение к лишению свободы и иным мерам;
- 4) штраф;
- 5) другие меры наказания.

База данных «Правонарушения» должна состоять из нескольких таблиц, содержащих ключевые поля (поле), по которым можно осуществлять различные связи между таблицами базы данных.

В базе данных должно быть не менее 30 записей (не менее 30 правонарушителей).

## Практическая работа № 28

### Тема: «Создание запросов к базе данных»

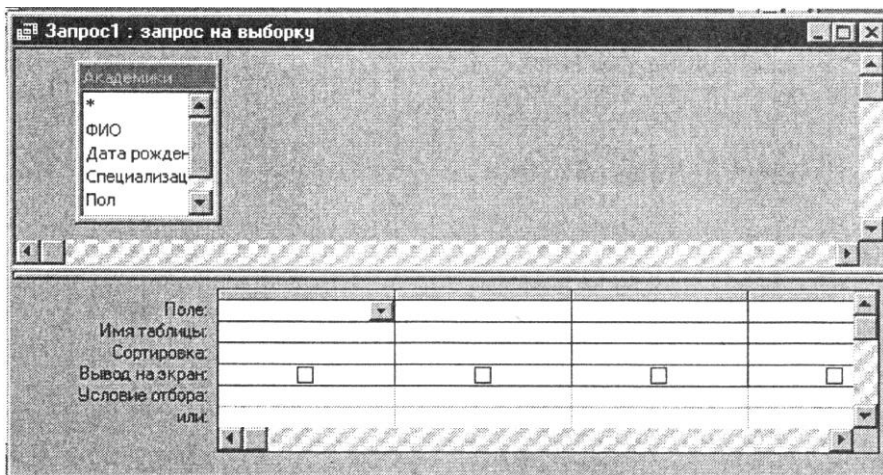
Создать файл базы данных *Akademik.mdb*. С помощью конструктора подготовить таблицу *Члены-корреспонденты Академии наук*. Заполнить таблицу данными, включив следующие поля: *ФИО, Дата рождения, Специализация, Пол, Год присвоения звания*.

ФИО	Дата рождения	Специализация	Пол	Год присвоения звания
Александрович А. И.	22.01.1906	поэт	м	1930
Амбросов А. Л.	16.06.1912	Фитопатологвирусоло	м	1970
Аринчин М. И.	28.02.1914	физиолог	м	1966
Бабосов Е. М.	23.02.1931	философ	м	1977
Бирич Т. В.	10.01.1905	офтальмолог	ж	1972
Бокуть Б. В.	27.10.1926	физик	м	1974
Бондарчик В. К.	01.08.1920	этнограф	м	1972
Будыка С. Х.	17.03.1909	гидролог	м	1972
Гуринович Г. П.	26.04.1933	физик	м	1970
Иванов А. П.	29.12.1929	физик	м	1974
Каменская Н. В.	10.01.1914	историк	ж	1959
Комаров В. С.	29.01.1923	химик	м	1970
Кулаковская Т. Н.	17.02.1919	агрохимик-почвовед	ж	1969
Мацкевич Ю. Ф.	27.07.1911	языковед	ж	1969
Пилипович В. А.	05.01.1931	физик	м	1977
Сикорский В. М.	10.10.1923	историк	м	1972
Старобинец Г. Л.	14.05.1910	химик	м	1969
Судник М. Р.	08.11.1910	языковед	м	1970
Ткачев В. Д.	19.02.1939	физик	м	1974
Хотылева Л. В.	12.03.1928	генетик	ж	1972
Шабуня К. И.	28.10.1912	историк	м	1969
Широканов Д. И.	20.05.1929	философ	м	1974

## Задание 1

Для таблицы *Члены-корреспонденты Академии наук* файла базы данных *Akademik.mdb* с помощью запросов на выборку изменить порядок следования полей, сделать невидимыми указанные поля, отсортировать записи по предложенному критерию, вывести на экран записи, отвечающие данным условиям.

1. Открыть файл базы данных *Akademik.mdb*.



2. В списке **Объекты** выбрать **Запросы** и нажать кнопку **Создать**. В диалоговом окне **Новый запрос** выбрать вариант **Конструктор**.

3. В диалоговом окне **Добавление таблицы** указать на таблицу *Академики* и нажать кнопку **Добавить**, а затем кнопку **Заккрыть**.

4. Выполнить отбор полей. Для этого из списка полей *Академики* двойным щелчком разместить поля на бланке запроса в строке **Поле**.

5. На панели инструментов **Конструктор запросов** нажать кнопку **Вид-Режим таблицы**.

6. Вернуться в режим конструктора.

7. Установить критерии отбора для вывода на экран полей: **ФИО**, **Специализация**, **Год присвоения звания**. Для этого на бланке в строке

8. **Вывод на экран** отключить флажки для полей: **Дата рождения**, **Пол**.

9. Перейти в режим таблицы. Убедиться, что видимыми остались только отобранные поля.

10. Вернуться в режим конструктора.

10. Изменить порядок следования столбцов, переместив столбец **Пол** на второе место. Для этого выделить столбец **Поли** перетащить его в нужное место.

11. Вернуть видимость всем полям таблицы.

12. Отобразить записи, которые содержат информацию об академичках-женщинах.

Для этого в условии отбора в поле *Пол* ввести *ж*.

13. Упорядочить по возрастанию отобранные данные по значению поля *Дата рождения*. Для этого на бланке запроса в строке *Сортировка* в поле *Дата рождения* выбрать способ сортировки *По возрастанию*.

14. Просмотреть отобранные записи. Убедиться, что записи расположены в возрастающем порядке значений поля *Дата рождения*.

15. Создать запросы для вывода на экран записей, отображающих информацию:

- об академиках-физиках;
- об академиках-физиках и химиках;
- об ученых, получивших звание академика до 1970 года;
- об академиках-философах и языковедах;
- обо всех академиках, кроме физиков.

## Задание 2.

1. В личной папке создайте базу данных *Гостиница*, объектом которой является таблица *Данные*:

№	ФИО	Цель поездки	Продолжительность поездки	Тариф в сутки	Фото гостиницы
1	Иванов В.В.	Отпуск	10	80 р.	
2	Судаков А.Н.	Командировка	12	60 р.	
3	Судакова М.П.	Командировка	15	60 р.	
4	Сорокин А.В.	Бизнес	14	120 р.	
5	Ходаков Е.Г.	Бизнес	18	90 р.	
6	Степчин Н.В.	Командировка	21	90 р.	
7	Панченко Т.Ю.	Отпуск	10	70 р.	

1. Создайте серию запросов (в запросы включить все поля таблицы):

1. Выберите фамилии людей, приехавших в командировку более чем на 10 дней.
  2. Выберите фамилии людей, которые платят в сутки от 70 до 100 р.
  3. Выберите фамилии людей, продолжительность пребывания которых 2 или 3 недели.
  4. Выберите фамилии людей, целью приезда которых не является отдых.
  5. Выберите проживающих, фамилии которых начинаются на букву «С».
  6. Создайте запрос с вычисляемым полем – *Сезонные скидки*: [Тариф в сутки]\*0,05.
1. Для первого запроса создайте форму через Мастер форм (внешний вид формы: выровненный).

2. Для таблицы Данные создайте отчет через Мастер отчетов, ориентация – альбомная, макет - ступенчатый).

3. Используя форму, добавьте в таблицу Данные одну новую запись.

4. Скопируйте таблицу Данные в MS Excel, удалите столбец Фото гостиницы.

5. Добавьте столбец Расходы на содержание номера, установите денежный формат, выполните расчеты: если продолжительность поездки превышает 15 дней, то расходы на содержание номера составляют 4% от тарифа, иначе – 3%.

Добавьте столбец Итоговая сумма, установите денежный формат данных, выполните необходимые расчеты (с учетом расходов на содержание номера).

Отформатируйте таблицу: установите перенос по словам, отцентрируйте заголовки, выполните заливку первой строки.

1. Постройте диаграмму по столбцам ФИО проживающего и Итоговая сумма, примените все необходимые настройки, разместите на отдельном листе.

2. Сохраните файл в личную папку под названием Расчеты.

3. Запустите MS Word, установите параметры страницы: размер бумаги - 20×15; все поля по 2 см.

4. Создайте документ по образцу:

***Счет за проживание***

Уважаемый, *ФИО проживающего*,  
за *Продолжительность поездки* дней проживания в нашей гостинице  
Вам необходимо заплатить *Итоговая сумма*.  
Администрация гостиницы.

1. Добавьте нижний колонтитул – *Фамилия, Номер группы, Дата создания*. Отформатируйте его: шрифт – Arial, размер – 13, начертание – курсив, выравнивание – по центру.

2. Сохраните документ в личную папку под названием *Счет1*.

## **Задание 2.**

### **Вариант 37**

1. В личной папке создайте базу данных *Гостиница*, объектом которой является таблица *Данные*:

Номер	Фамилия проживающего	Цель	Гостиница	Количество мест в номере	Продолжительность поездки	Стоимость проживания в сутки
-------	----------------------	------	-----------	--------------------------	---------------------------	------------------------------

1	Иванов	Командировка	Москва	4	10	600 р.
2	Судаков	Командировка	Прибалтийская	2	12	1200 р.
3	Судакова	Командировка	Прибалтийская	2	15	1200 р.
4	Степчин	Отпуск	Москва	3	14	800 р.
5	Панченко	Отпуск	Москва	2	18	1600 р.
6	Панченко	Отпуск	Москва	2	21	1600 р.
7	Круглов	Отпуск	Ереван	1	10	1800 р.

2. Создайте серию запросов (в запросы включить все поля таблицы):

1. Выберите людей, которые приехали в отпуск и остановились в гостинице Москва.
2. Выберите людей, которые платят за проживание в сутки от 1000 до 1500 р.
3. Выберите людей, которые остановились в гостинице Москва или Ереван.
4. Выберите людей, которые остановились не в одноместном номере.
5. Выберите проживающих, чьи фамилии начинаются на букву «П».
6. Создайте запрос с вычисляемым полем – *Стоимость номера с питанием*:

[Стоимость проживания в сутки]+500.

3. Для первого запроса создайте форму через Мастер форм (внешний вид формы: выровненный).

4. Для таблицы *Данные* создайте отчет через Мастер отчетов, ориентация – альбомная, макет - ступенчатый).

5. Используя форму, добавьте в таблицу *Данные* одну новую запись.

6. Скопируйте таблицу *Данные* в MS Excel.

7. Добавьте столбец *Обслуживание номера*, установите денежный формат, выполните расчеты: если цель приезда – командировка, то расходы на обслуживание номера составляют 5% от стоимости проживания, иначе – 3%.

8. Добавьте столбец *Итоговая сумма*, установите денежный формат данных, выполните необходимые расчеты (с учетом обслуживания номера).

9. Отформатируйте таблицу: установите перенос по словам, отцентрируйте заголовки, выполните заливку первой строки.

10. Постройте диаграмму по столбцам *ФИО проживающего* и *Итоговая сумма*, примените все необходимые настройки, разместите на отдельном листе.

11. Сохраните файл в личную папку под названием *Расчеты*.

12. Запустите MS Word, установите параметры страницы: размер бумаги - 20×15; все поля по 2 см.

13. Создайте документ по образцу:

*Счет за проживание*

Уважаемый, *ФИО проживающего*,  
за проживание в нашей гостинице  
Вам необходимо заплатить *Итоговая сумма*.  
Администрация гостиницы «*Гостиница*».

14. Добавьте нижний колонтитул – Фамилия, Номер группы, Дата создания.  
Отформатируйте его: шрифт – Arial, размер – 13, начертание – курсив, выравнивание – по центру.

15. Сохраните документ в личную папку под названием *Счет2*.



## Практическая работа № 29-30

Тема: «Создание растровых и векторных изображений и чертежей в графических редакторах. Создание мультимедийных презентаций (по профилю специальности)»

Цель: изучение базовых возможностей среды Ms PowerPoint 2010-2013 по созданию презентаций.

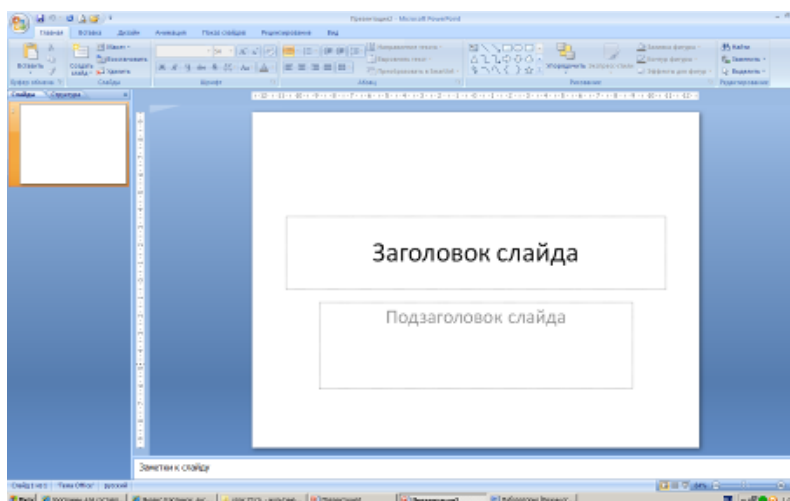
### Настройка анимации

Выполнив работу, вы научитесь:

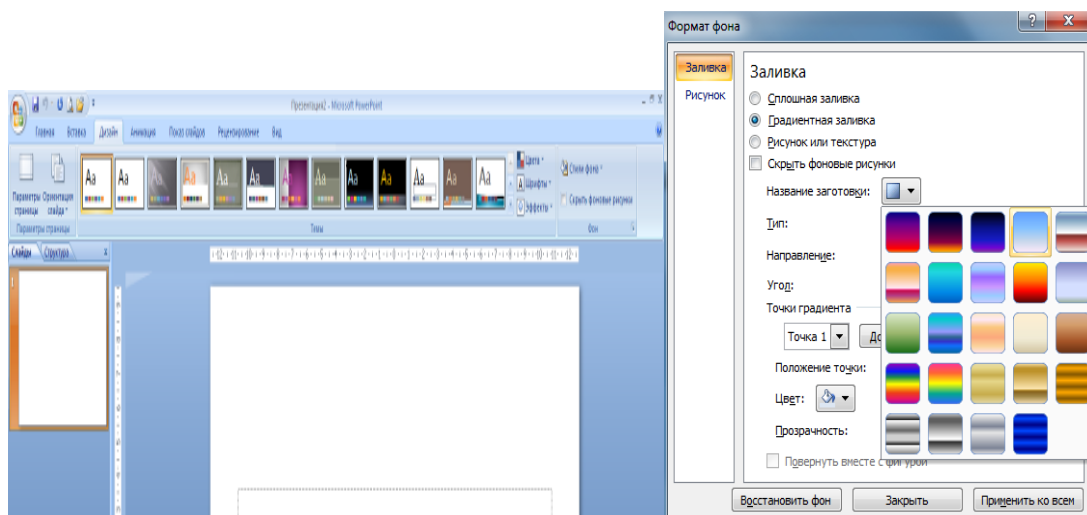
- создавать слайды различной структуры;
- задавать необходимый фон;
- добавлять в презентацию рисунки, объекты WordArt и работать с ними;
- создавать гиперссылки и управляющие кнопки;
- вставлять звук;
- настраивать анимацию.

### СОЗДАНИЕ СЛАЙДОВ. ВСТАВКА КАРТИНОК.

1. Запустите программу PowerPoint. Для этого выполните *Пуск/ Все программы/ Microsoft Office/ PowerPoint*. В открывшемся окне, предназначенном для открытия или выбора презентации, по умолчанию, создается 1 слайд – титульный лист

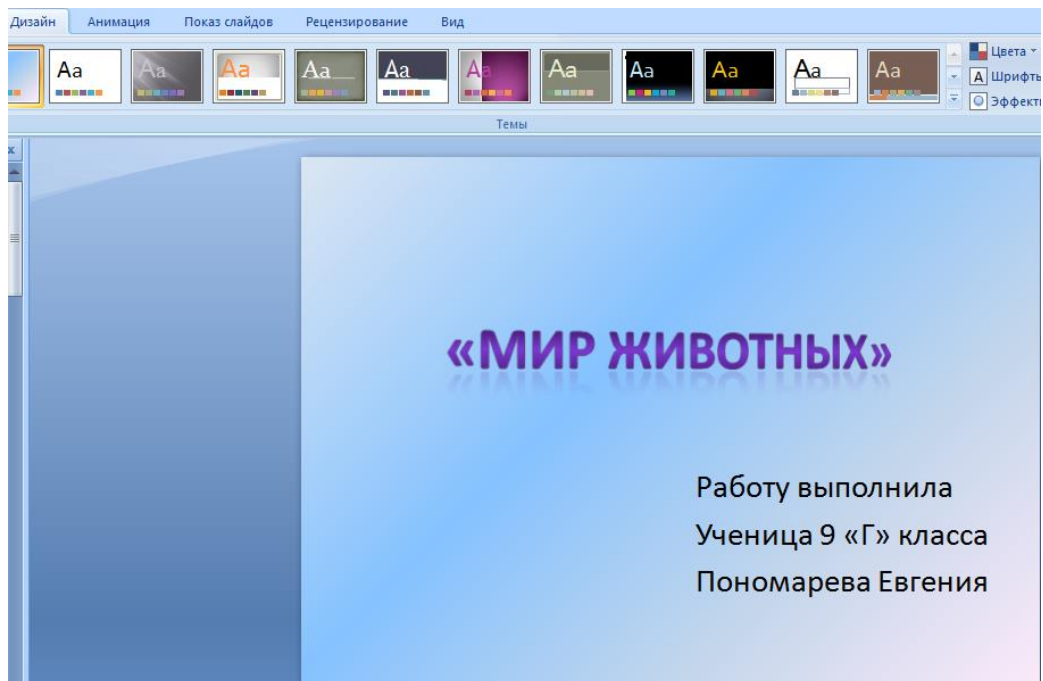


2. Выберите цветовое оформление слайдов. Вкладка *Дизайн/ Стили фона/ Формат фона/ Градиентная заливка/ Расцвет/ Применять ко всем*.

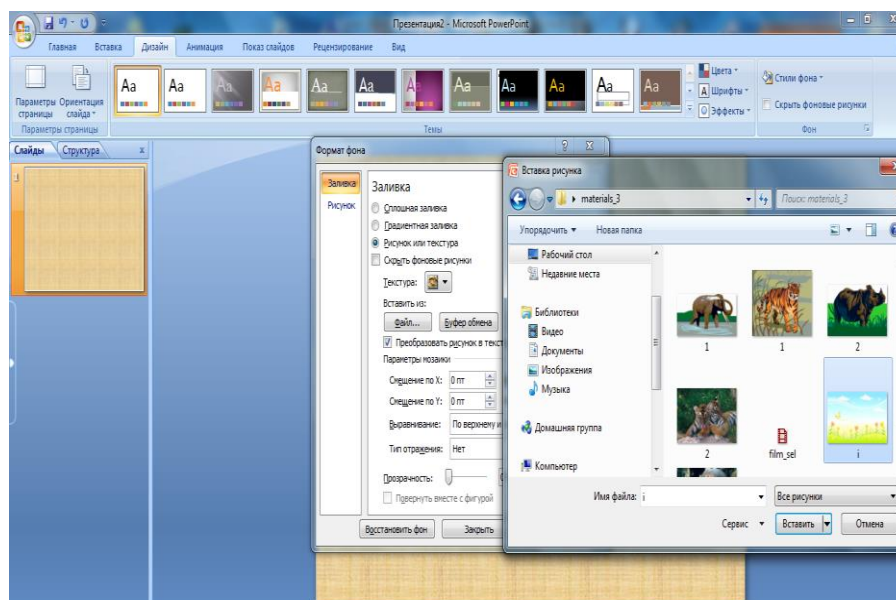


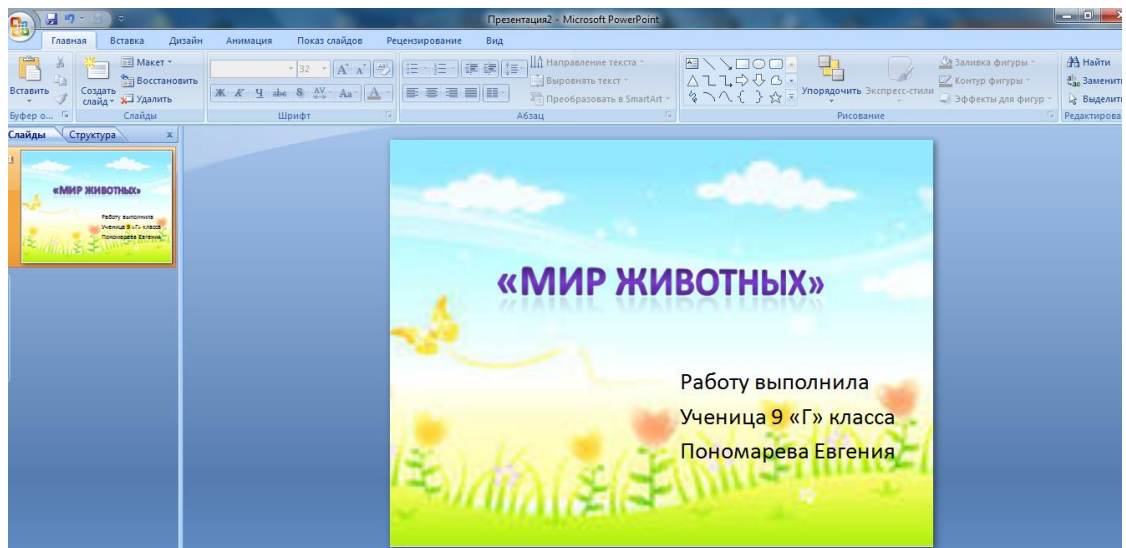
3. Оформление заголовка «Животный мир» - фиолетовым цветом: вкладка **Вставка/WordArt**.

Шрифт Calibri, полужирное начертание, размер шрифта 66. В подзаголовке пишем кто выполнил работу

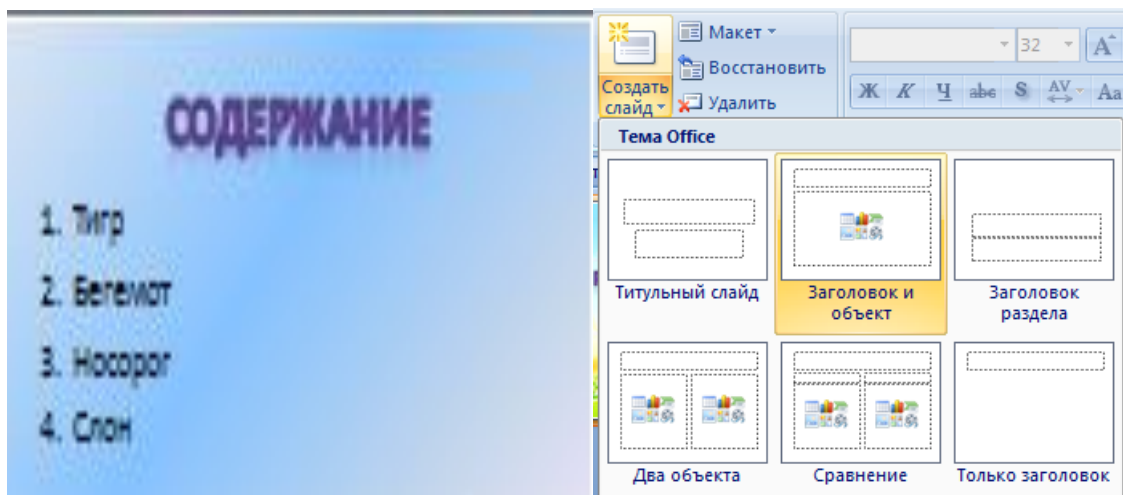


3. Заходим во вкладку **Дизайн/Стили фона/Формат фона/Рисунок или текстура** нажимаем по слову **Файл** выбираем путь hermes/public/Информатика/Группа ()





4. 2 слайд: Главная/ Создать слайд/Заголовок и объект



В заголовке с помощью объекта *WordArt* написать *Содержание*. Размер шрифта 54

3 слайд Главная/ Создать слайд/Два объекта Вставляем картинки, набираем текст



4-6 слайд аналогично 3 слайду

4 слайд

## БЕГЕМОТ



Бегемот — одно из крупнейших современных наземных животных. Вес крупных старых самцов иногда превышает 4 тонны, таким образом, гиппопотам конкурирует с носорогами за второе место по массе среди наземных животных после слонов.

5 слайд

## НОСОРОГ



Носорог — это огромный толстокожий представитель млекопитающих, который имеет один или два рога, служащие не только для обороны, но и для добывания пищи. На Земле сегодня сохранилось всего пять видов носорогов, два из которых обитают в Африке, а три — в Азии. Все они занесены в Красную книгу.

## 6 слайд

# СЛОН

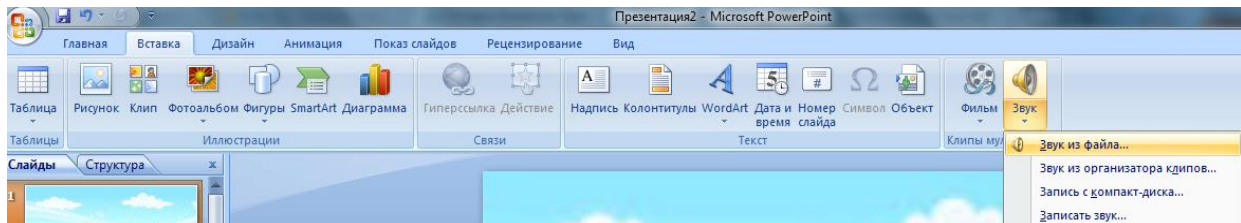


Слоны — удивительные животные. Они очень любят воду, они любят "принимать душ" поливая себя с помощью своего многофункционального хобота. Хобот для слона это длинный нос с различными функциями. Он благодаря ему дышит, нюхает, пьет, захватывает еду, и даже издает звуки. Один только хобот содержит приблизительно 100 000 мускулов.

## НАСТРОЙКА АНИМАЦИЙ И ВСТАВКА ЗВУКА

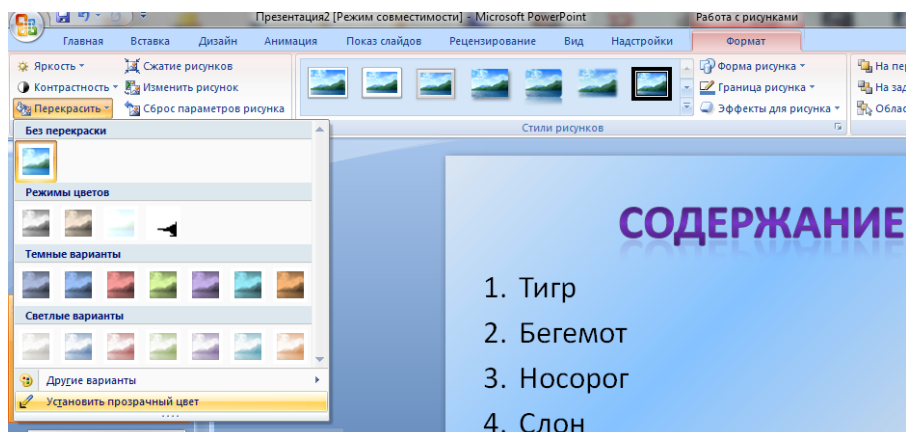
### 1 слайд

Вставка звука. Выбираем вкладку **Звук/Звук из файла** выбираем путь Рабочий стол /9класс/ Презентации /Практическая работа.№2звуковой файл с названием пение птиц



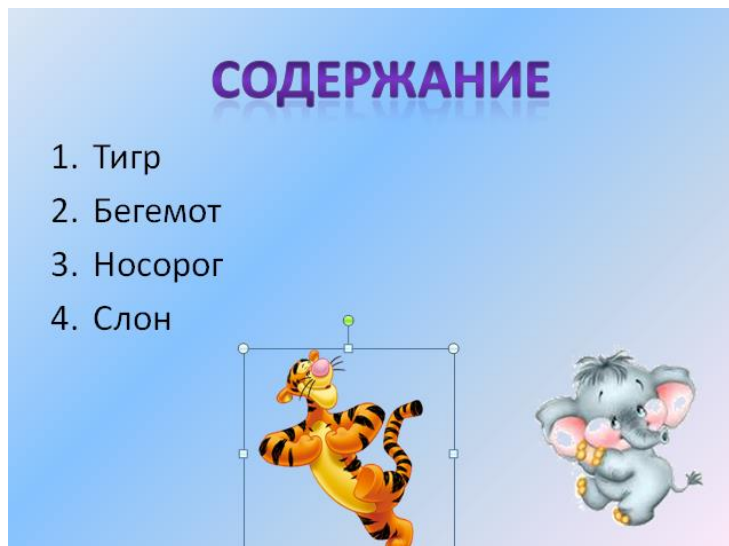
### 2 слайд

Вставляем картинку с тигром и слонem. **Вставка-Рисунок** выбираем путь и сам файл нажимаем ОК. Чтобы убрать белый фон у слона выделяем картинку, выбираем вкладку **Работа с рисунком - Перекрасить - Установить прозрачный цвет**.

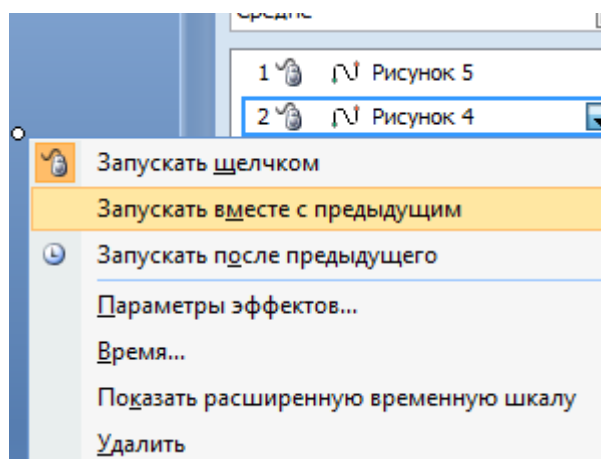
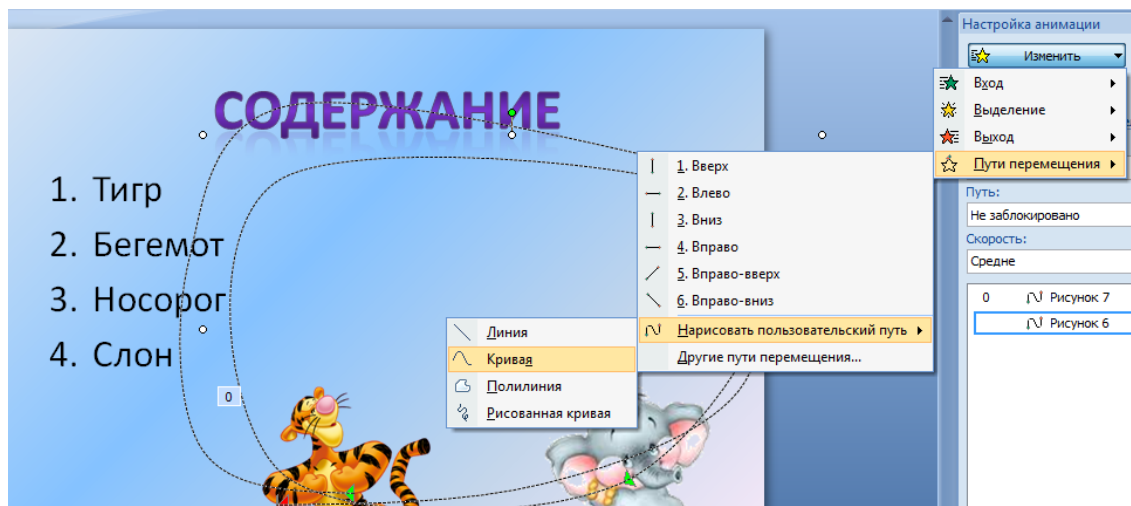


# СОДЕРЖАНИЕ

1. Тигр
2. Бегемот
3. Носорог
4. Слон



Выделяем тигра выбираем вкладку **Анимация/Настройка анимации** справа появляется кнопка **Добавить эффект/Пути перемещения/Нарисовать пользовательский путь/Кривая**. Точно также для слона

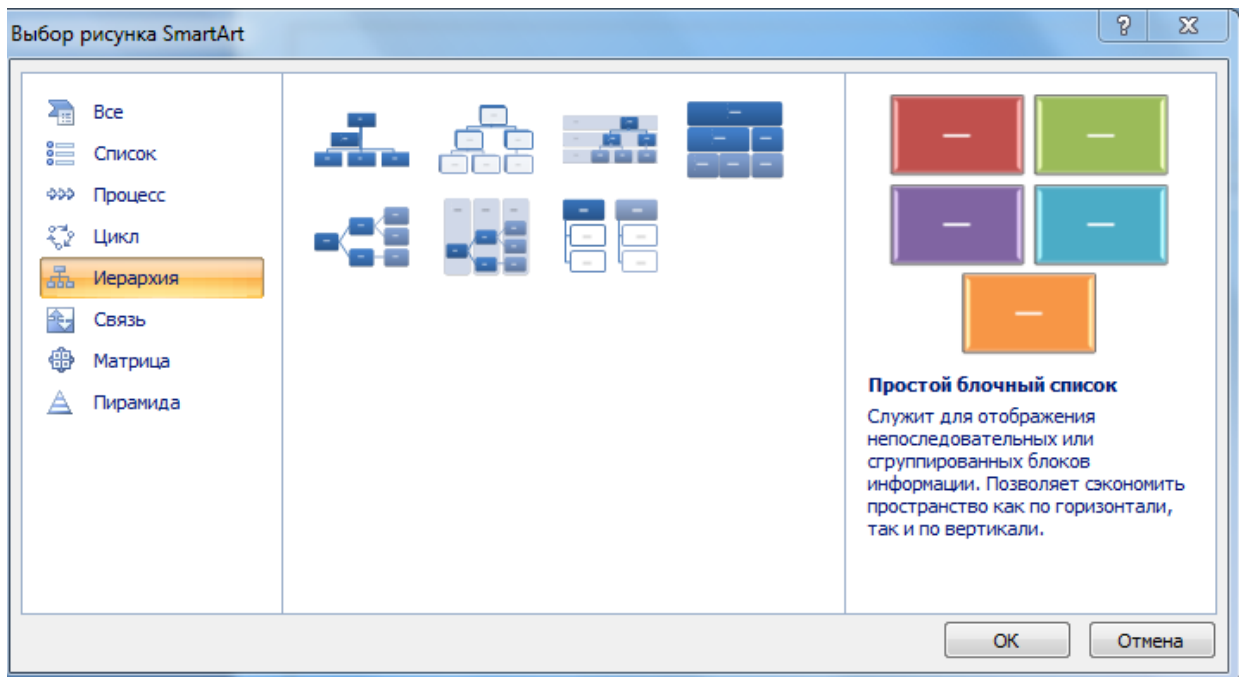


По своему усмотрению настройте Анимацию для картинок на слайдах (**Анимация-Настройка анимации**)

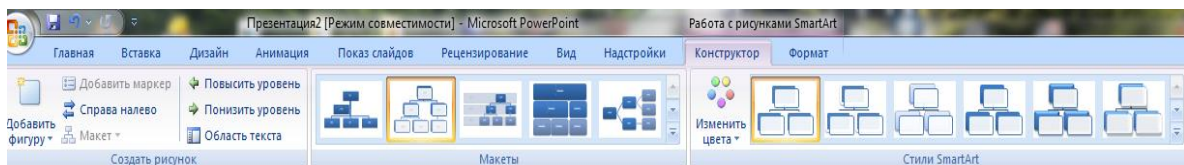
## ВСТАВКА SMARTART

## 7 слайд

Выбираем вкладку **Вставка-SmartArt - Иерархия** выбираем любую схему



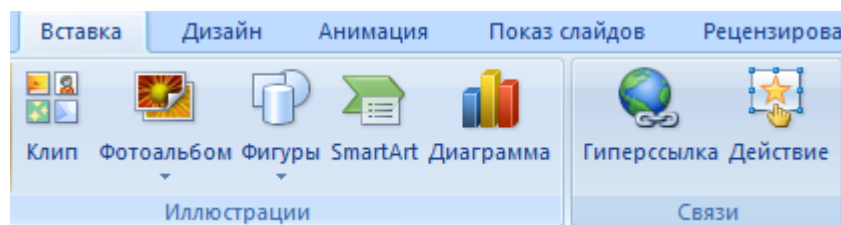
На слайде появляется схема с помощью Конструктора добавляем или удаляем фигуру. Заполняем схему



## ГИПЕРССЫЛКИ. УПРАВЛЯЮЩИЕ КНОПКИ

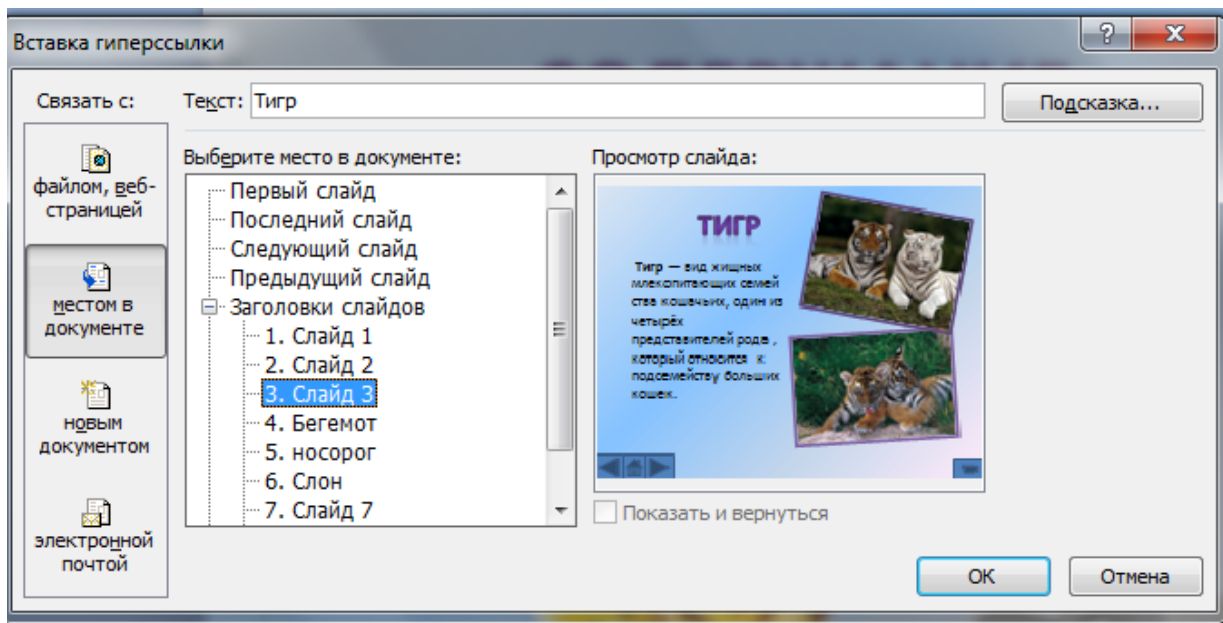
Возвращаемся к второму слайду для создания гиперссылок

1. Выделяем слово ТИГР. Выбираем вкладку **Вставка-Гиперссылка**

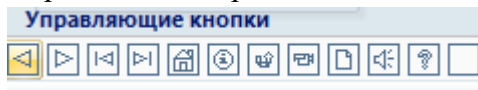


Появляется диалоговое окно выбираем Место в документе и слайд про тигров и нажимаем ОК.

Аналогично для слайдов Бегемот, Носорог, Слон.



Переходим на третий слайд выбираем **Вставка - Фигуры - Управляющие кнопки.**



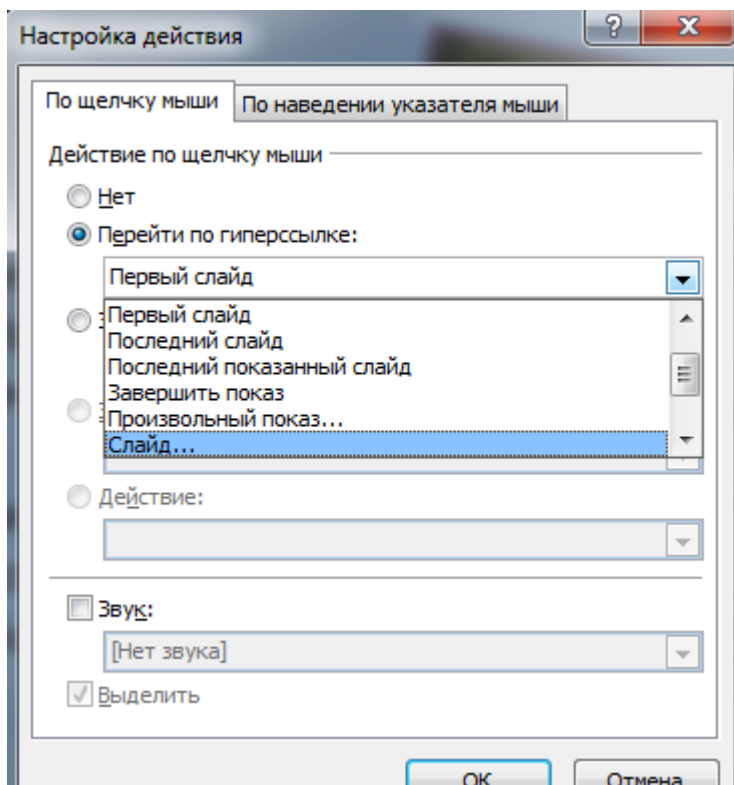
Рисуем в левом нижнем углу слайда три кнопки



Настраиваем гиперссылки. Выделяем кнопку

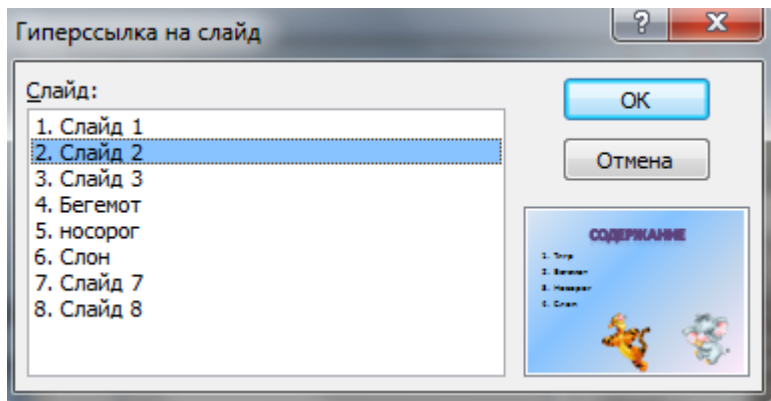


. Выбираем в меню **Вставка-Гиперссылка**



Выбираем **Слайд -Второй слайд - ОК**

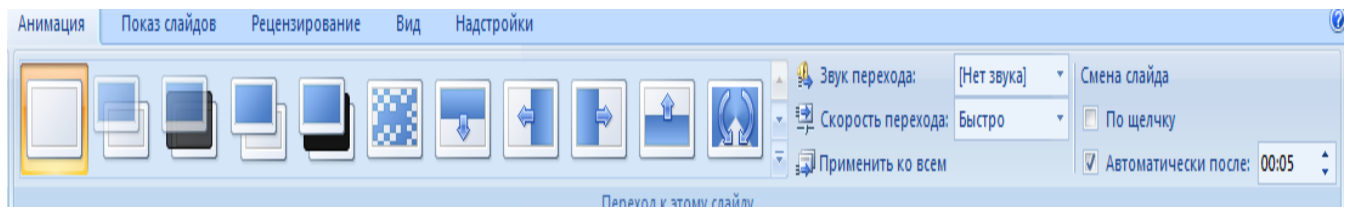




**Аналогично для 4-6 слайда вставка управляющих кнопок и создание гиперссылок**

Создаем последний пустой слайд с помощью объекта WordArt вставляем надпись **Спасибо за внимание!**

Настраиваем презентацию в автоматический режим. Выбираем вкладку **Анимация** ставим галочку **Автоматически после**, устанавливаем количество секунд и нажимаем на кнопку **Применить ко всем**.



***Сохраняем презентацию и работу показываем преподавателю.***