

Міністерство освіти та науки України  
Управління освіти і науки у Донецькій області  
Обласний інститут післядипломної педагогічної освіти  
Маріупольське управління освіти міської ради  
Маріупольський науково-методичний центр  
**230-річчю**  
**м. Маріуполя**  
**присвячується**

**Інформатика**  
**в школі**



С  
П  
Е  
Д  
Ц  
Е  
В  
И  
П  
К  
С  
У  
П  
К  
С  
2

**2008 рік**

## «Спецвипуск»

№ 2 2008 р.

лютий

Маріупольського науково-методичного центру

### Головний редактор

Директор науково-методичного центру Т.І. Кустенко

Редактор О.І. Гринько

### Редколегія:

Б.І. Ерді, І.Е. Дворядкіна,  
І.Г. Смирнова, Н.В. Горбачова,  
О.Б. Панов, Т.Д. Курява,  
О.В. Курявий, Т.Г. Дубовик,  
Н.П. Чередниченко

### Рецензент:

Професор кафедри інформатики, доктор фіз-мат. наук Гранкін В.П.

### Коректор:

Курява Т.Д.

Оформлення, набір, комп'ютерна верстка і друк:

Курява Т.Д.,  
Курявий О.В.  
Кейда С. І.

Адреса редакції:

вул. Євпаторійська, 56

E-mail:

[nmc-kejda@yandex.ru](mailto:nmc-kejda@yandex.ru)

Видавництво здійснює науково-методичний центр з 1994 р. Мова видання – українська, російська. Видається один раз на три місяці.

Точка зору авторів публікацій не завжди поділяється редакцією. За достовірність фактів і цифр, відповідність їх вказаним джерелам відповідальність несе автор.

## *МВ ЗМІСТ*

### Методичні розробки уроків

<i>Вступ</i> .....	3
<i>Ерді Б.І.,</i> Комп'ютерні презентації.....	4
<i>Дворядкіна І.Е.,</i> <i>Смирнова І.Г.,</i> Табличний процесор .....	14
<i>Горбачова Н.В.,</i> Бази даних. Системи управління базами даних .....	37
<i>Панов О.Б.,</i> Глобальна мережа Інтернет.....	63
<i>Курява Т.Д., Курявий О.В.,</i> Комп'ютерне моделювання. Основи алгоритмізації.....	82
<i>Дубовик Т.Г., Чередниченко Н.П.</i> Основи програмування.....	98

## Введение

Методическое пособие разработано преподавателями школ города Мариуполя для проведения занятий по предмету «Информатика» в старших классах средних образовательных школ I-III ступеней, лицеев, гимназий и других учебных заведениях.

Данный сборник по своему содержанию соответствует государственному стандарту базового и полного среднего образования для универсального профиля обучения в старшей школе. Также отдельные темы могут быть использованы для гуманитарного профиля и физико-математического.

Содержание разработанных уроков представлено в виде развернутых поурочных планов, содержащих триединые цели урока, ход урока, теоретическую и практическую части. Также в разработках предложены тестовые задания для определения критериев, оценки знаний учащихся по конкретным темам.

Материал изложен в логической последовательности и доходчивой форме изложения.

Надеемся, что предложенный «Спецвыпуск» поможет учителям информатики сделать уроки информатики результативными и интересными.

**Тема 4. Прикладне програмне забезпечення загального призначення**

**4.3. Комп'ютерні презентації (2 години)**

№ урока	Тема урока	
1	Понятие компьютерной презентации. Назначение системы подготовки презентации. Объекты, которые обрабатываются с помощью системы подготовки презентации.	
2	Создание образовательных компьютерных презентаций. Демонстрация презентаций.	

**Урок 1**

**Тема урока:** Понятие компьютерной презентации. Назначение системы подготовки презентации. Объекты, которые обрабатываются с помощью системы подготовки презентации.

**Цель урока:** *Учащиеся должны знать* предназначение компьютерных презентаций, способы их создания, правила работы с компьютерными презентациями, возможности применения разных типов объектов для создания презентации. *Учащиеся должны уметь* запускать программу для создания презентации, создавать слайды.

**Оборудование:** Программа PowerPoint пакета Microsoft Office.

**Ход урока**

**I.** Проверка домашнего задания.

**II.** Изучение нового материала

1) Знакомство с программой Power Point

1 PowerPoint для Windows – это приложение, предназначенное для создания электронных презентаций с использованием различных видов информации: аудио, видео, графическая, текстовая и их последующей демонстрацией с использованием различных эффектов.

Создание презентации в большинстве случаев включает четыре основных этапа. Сначала необходимо спланировать общий вид презентации. Затем следует отредактировать тексты презентации и поместить рисунки. Кроме того, современные презентации должны позволять задавать специальные эффекты, определяющие тип демонстрации. Появление каждого нового слайда может сопровождаться тем или иным анимационным эффектом.

Существуют три способа создания презентаций:

\* создание новой презентации на основе предлагаемой структуры:

- 1) в меню *Файл* выберите команду *Создать*;
- 2) на вкладке *Презентации* щелкните *Мастер автосодержания*.

\* создание новой презентации с помощью шаблона дизайнера:

- 1) в меню *Файл* выберите команду *Создать*;
- 2) на вкладке *Презентации* щелкните шаблон дизайнера, который вы хотите использовать.

\* создание новой презентации из пустой презентации:

- 1) в меню *Файл* выберите команду *Создать*;
- 2) выберите вкладку *Обилие* и дважды щелкните значок *Новая презентация*.

Новая презентация будет использовать цветовую схему, заголовков и стили текста презентации, установленной по умолчанию.

Хотя существует много способов запуска программы, наиболее простой путь – это использования кнопки *Пуск*.

Окно Power Point имеет стандартные элементы окон Windows.

## 2) Создание презентации

В программе **Power Point** есть умелые помощники – **Мастера**. **Мастера** – это вспомогательные средства, которые помогают выполнить сложный многоступенчатый процесс по созданию презентации. Это ряд диалоговых окон, которые задают вопросы о том, как вы хотите выполнить задачу. После каждого ответа на вопрос мастер будет менять внешний вид презентации. Всегда можно перейти к следующему или вернуть к предыдущему диалоговому окну.

Использование *Мастера автосодержания* – это простейший и самый быстрый способ создания новой презентации. Все, что необходимо будет знать – это как должна выглядеть презентация. **Мастера автосодержания** можно будет запустить прямо со стартового диалогового окна **Power Point**. После выбора темы и шаблонов презентации и ответов на ряд вопросов, касающихся оформления и дальнейшего использования презентации этот мастер подготовит слайды. *Мастеру* понадобится некоторое время, чтобы создать презентацию. Затем мастер откроет презентацию в режиме **Структура**. Там можно увидеть подборку слайдов, готовых для оформления. Предложенные слайды не более чем рабочая версия. Независимо от того, насколько тщательно спланирована презентация, в неё по ходу работы можно вносить любые изменения и коррективы.

Простейший способ добавления нового слайда – это щёлкнуть на кнопке **Создать Слайд** на панели инструментов **Стандартная**. После этого откроется диалоговое окно **Создать Слайд**. В этом диалоговом окне выбрать авторазметку для нового слайда.

Если один слайд не соответствует задуманной теме презентации, его можно удалить. Это делается в режимах **Структура**, **Сортировщик Слайдов** и **Слайды**. В режимах **Структура** или **Сортировщик Слайдов** выберите слайд, который надо удалить, и нажмите <delete>.

**Power Point** берёт на себя всю сложную работу по разметке и оформлению внешнего вида слайдов. Программа содержит много встроенных стилей размещения и оформления. Чтобы применить встроенный стиль авторазметки, нужно назначить один из встроенных шаблонов.

Средство авторазметки автоматически размещает информацию специфическим образом на отдельном выбранном слайде. Можно изменить фон слайдов. Применение шаблонов оформления выполнит это автоматически. В каждом шаблоне определены специфические цветовые и графические элементы оформления слайдов.

Чтобы изменить шаблон оформления, надо щёлкнуть по кнопке **Применить** оформления в режимах **Структура**, **Слайды** или **Сортировщик слайдов**. Выбрать один из доступных шаблонов из левого списка диалогового окна. После щелчка на имени шаблона, появится миниатюра, на которой в точности переданы цвета и размещение элементов оформления.

При выборе шаблона оформления учитывается три критерия: предполагаемая аудитория, тема доклада, форма представления презентации. Кроить презентацию надо исходя из того, кто её будет смотреть (друзья, коллеги, инвесторы). Подобранный шаблон должен подчеркивать серьёзность раскрываемой темы.

И, наконец, то, каким способом планируется представить презентацию, также должно повлиять на выбор шаблона.

### Форматирование слайда

Чтобы изменить формат слайда, вначале необходимо его выделить в режиме сортировщика или сделать текущими. После этого можно воспользоваться одной из команд меню *Формат*.

**Разметка слайда** – данная команда позволяет изменить разметку для выбранного слайда с помощью диалогового окна *Разметка слайда*, аналогичного окну *Создание слайда*. Чтобы выполнить изменения, необходимо выбрать в данном окне требуемую разметку для слайда и нажать кнопку *Применить*.

**Цветовая схема слайда** – это команда используется для установки цветовой схемы для выбранного слайда или для всей презентации. При выполнении данной команды отображается диалоговое окно *Цветовая схема*, в которой имеется две вкладки:

**Стандартная** – позволяет установить для слайдов одну из имеющихся стандартных цветовых схем. Чтобы применить выбранную схему для текущего слайда следует нажать кнопку *Применить*. С другой стороны, чтобы установить выбранную схему для всех слайдов презентации, необходимо воспользоваться кнопкой *Применить ко всем*;

**Специальная** – дает возможность изменить цвет того или иного элемента схемы. Для этого необходимо выбрать требуемый элемент, например, *Фон*, или нажать кнопку *Изменить цвет*. Затем в диалоговом окне *Цвет фона* следует выбрать нужный цвет на вкладке *Обычные* или воспользоваться для определения оттенка вкладкой *Спектр*, после чего нажать кнопку *ОК*.

Измененную цветовую схему нужно применить к выбранному слайду или ко всей презентации (кнопка *Применить ко всем*).

**Фон** – позволяет выбрать заливку, которая будет использоваться для фона слайдов. После выполнения команды отображается диалоговое окно *Фон*, в котором можно с помощью раскрывающегося списка выбрать требуемый цвет или способ заливки.

**Применить шаблон оформления** – дает возможность установить новый шаблон оформления, т.е. установить определенную цветовую палитру для фона и объектов слайда, а также определить расположение элементов на слайдах. Для того следует открыть презентацию в любом режиме и выполнить данную команду. Затем в диалоговом окне *Применить шаблон оформления* следует выбрать новый шаблон для оформления презентации, после чего нажать кнопку *Применить*.

В результате выполнения действий все слайды презентации будут оформлены по новому шаблону.

### 3) Добавление графиков и диаграмм

Модуль *Microsoft Graph*, включенный в комплект *Power Point*, предлагает 18 различных типов диаграмм любых мыслимых форм. В результате процесс формирования диаграмм становится очень простым и почти автоматическим.

Поскольку диаграмма является визуальным представлением числовых данных, она оказывает на аудиторию гораздо более сильное воздействие, чем поток самых красноречивых слов.

**Microsoft Graph** – это мини приложение, установленное на компьютер одновременно с установкой *Power point*. Это самостоятельная программа. Когда идет работа по составлению диаграммы *Microsoft Graph* открывает свое меню и панель инструментов. Прежде чем приступить к созданию диаграммы, надо определить, что необходимо сообщить аудитории с помощью этой диаграммы.

#### **Добавление таблиц**

Если необходимо вывести на экран два текстовых абзаца рядом (например, чтобы привести некоторые соображения за и против какой-

то идеи), следует воспользоваться таблицей в виде строк и столбцов. Таблицы на слайдах легко создаются и модифицируются. Работать с ними проще, чем со столбцами, сформированными табуляцией.

Обычно таблицы на слайдах содержат только текст, однако в них с легкостью могут быть включены и числа. Когда докладчику не нужна вся графическая мощь системы Power Point, таблица послужит хорошей альтернативой для быстрого создания презентации. Кроме того, таблица часто используется как дополнение к графику в тех случаях, когда перед аудиторией необходимы и числовые данные.

Используя для работы с таблицами модуль приложения Word, Power Point тем самым представляет пользователю все преимущества мощного текстового процессора Word с его обширными возможностями.

#### Создание слайда с таблицей

Когда требуется включить в презентацию таблицу, можно либо создать новый слайд с авторазметкой **Таблица**, либо добавить таблицу в уже существующий слайд, на котором могут присутствовать и другие объекты, такие как маркированные списки.

Создать в презентации слайд с таблицей очень просто. Надо выполнить следующие действия:

1. в режиме слайдов перейти к тому слайду, за которым должен следовать табличный.

2. щелкнуть на кнопке **Создать слайд** панели инструментов **Стандартная**.

3. в окне диалога **Создать слайд** дважды щелкнуть на авторазметке **Таблица**.

4. дважды щелкнуть на место заполнения таблицы, чтобы открыть окно диалога, изображенное ниже.

5. ввести в счетчики **Число столбцов** и **Число строк** окна диалога **Вставить таблицу Word** – предполагаемое число строк и столбцов новой таблицы. При этом для изменения указанных величин можно воспользоваться кнопками со стрелками, расположенными в правой части счетчиков. Затем, чтобы создать таблицу, щелкнуть на кнопке ОК.

Если данные были импортированы в таблицу Power Point из приложения Microsoft Excel, имеет смысл создать резервный слайд с таблицей Excel. Когда таблица Excel внедряется в слайд Power Point, она имеет такой же вид, как и в приложении Microsoft Excel, Вдобавок, не выходя из Power Point, можно пользоваться всеми вычислительными средствами Excel. Например, внедренные таблицы Excel могут включать формулы, с помощью которых выполняются сложные расчеты. В этом случае результаты вычислений будут автоматически обновляться при изменении содержащихся в таблице исходных данных.

Можно добавить таблицу в существующий слайд. Это можно сделать двумя способами:

1. щелкнуть на кнопке **Вставить таблицу Microsoft Word** панели инструментов.

2. указать на ячейку, расположенную в левом верхнем углу решетки, нажать кнопку мыши и протащите указатель через нужное количество строк и столбцов.

Второй способ:

- 1.указать пункт **Рисунок** меню **Вставка**.

- 2.выполнить команду **Рисунок - .Таблица Microsoft Word**.

- 3.в появившемся окне диалога **Вставить таблицу Word** ввести требуемое число строк и столбцов и щелкнуть на кнопке ОК.

Если таблица уже создана в программе *Word*, можно просто перетасовать ее в слайд *Power Point*.

#### Ввод текста в таблицу

При создании нового табличного слайда внутри серой рамки на слайде появляется пустая таблица. Теперь можно перейти к заполнению ячеек таблицы. В начале работы с таблицей текстовый курсор мерцает в первой ячейке в левом верхнем углу таблицы. Вводимый текст появляется в месте расположения текстового курсора. Для перемещения точки вставки в другую ячейку нажать клавишу *Tab* или просто щелкнуть на нужной ячейке.

Во время ввода текст переносится внутри ячейки так, чтобы не выходить за ее границы. Другими словами, если текст при вводе упирается в правую границу ячейки, следующее слово будет записано во вторую строку той же ячейки. При вводе текста в ячейку можно закончить один абзац и начать новый, нажав клавишу *Enter*. Достигнув конца последней ячейки в последней строке таблицы, можно, нажав клавишу *Tab*, добавить ещё одну строку внизу таблицы.

#### Форматирование таблицы

После ввода содержимого таблицы и завершения всех необходимых структурных изменений можно заняться совершенствованием общего внешнего вида таблицы. По умолчанию для форматирования таких элементов таблицы, как шрифта текста и цвета, *Power Point* много методов форматирования. Нужно просто выделить текст и воспользоваться кнопками панели инструментов *Форматирование*, командами меню *Формат* или контекстного меню.

Однако кроме текста в таблице есть и другие элементы, допускающие форматирование. Это, к примеру, рамки и цвет заливки ячеек.

Для форматирования ячеек *Power Point* предоставляет пользователю панель инструментов *Таблицы и границы* и команду *Границы и заливка* меню *Формат*. Однако простейшим способом добавления обрамления и заливки ячеек является использование одного из автоформатов таблицы. Каждый автоформат таблицы предлагает свое собственное сочетание рамок ячеек, их заливки, шрифтов и расцветок символов. Кроме того, применяя автоформат, можно автоматически задать размер таблицы таким образом, чтобы ее строки и столбцы по высоте и ширине точно соответствовали содержимому ячеек.

Вместо применения стандартных комбинаций обрамления и заливок можно настроить эти элементы оформления таблицы вручную. Для этого нужно воспользоваться командой *Границы и заливка* меню *Формат* или инструментами панели *Таблицы и границы*, которые кроме всех функций команды *Границы и заливка* предоставляют пользователю некоторые дополнительные возможности.

Если вместо панели *Таблицы и границы* используются команды *Границы и заливка*, то можно выбрать цвет обрамления, а также типы и цвет фонового узора.

На вкладке *Граница* можно указать тип линии обрамления, ее цвет и толщину, а затем настроить требуемое обрамление с помощью кнопок, расположенных по краям поля просмотра *Образец*. Вместо того, чтобы настраивать обрамление вручную, можно выбрать одну из готовых комбинаций линий обрамления с помощью образцов, расположенных в левой части окна диалога *Границы и заливка*. Выбор образца *Рамка* формирует границы по краям выделенной группы ячеек. Образец *Сетка* добавляет границы вокруг всех ячеек выделенного фрагмента, а образец *Все* прорисовывает границы во всей таблице. Чтобы



удалить какую – либо конкретную границу, просто щелкнуть на ней в поле просмотра **Образец**.

Щелкнув на ярлычке **Заливка** окна диалога **Границы и заливка**, можно выбрать цвет в поле **Образец**. При этом заливка может представлять собой однородный цвет, оттенок чистого цвета определенной интенсивности, задаваемой в процентах или узор.

Настроив параметры обрамления и заливки, надо щелкнуть на кнопке ОК, чтобы применить выбранный формат к выделенным ячейкам.

Если структура создаваемой таблицы достаточно сложная, то иногда бывает удобнее перетащить в Power Point таблицу, уже созданную ранее в программе Excel.

#### 4. Добавление и изменение графических объектов (векторных)

ГО – Графический объект. ГО может быть векторного типа (ГОВ) или пиксельного, или по-иному, фотографического (ГОП). Векторные и пиксельные объекты обрабатываются разными методами и обеспечивают разные возможности.

ГОВ включает: элементы: изображение, рамку, поле.


##### Создание ГОВ

ГОВ на слайде создается, одним из следующих способов:

- о составлением изображения из примитивов (линий, прямоугольников, эллипсов и т.д.)
- о включением готового изображения из файла и его модификацией.

Вставка готового изображения из файла и дальнейшая его модификация на слайде делается одним из двух способов:

- меню: **Вставка - Картинка - Из Файла**

- кнопка панели: 

Создав однажды ГО, в любой момент можно изменить его расположение, все его свойства.

##### Масштабирование ГОВ

После выделения объекта на слайде вокруг него появляются восемь меток: серединные и угловые. Масштабировать можно потянув указателем мыши за одну из меток, при этом, верхняя и нижняя серединные метки служат для изменения вертикального размера изображения, левая и правая серединные метки служат для горизонтального изменения размера изображения, а угловые метки дадут пропорциональное изменение размеров.

##### Изменение ГОВ

В любой момент графический объект векторного типа может быть изменен, объединен с другим объектом или его частями. Для изменения изображения следует выполнить следующие действия:

##### Изменение свойств ГОВ

С помощью панели свойств, которая вызывается через контекстное меню "Формат рисунка" или основное меню **Формат-Рисунок**, можно изменить все характеристики ГО. Например, цвет поля и рамки, яркость изображения, размер и местоположение на слайде.

#### 5. Добавление звуковых Файлов

Чтобы добавить звук в слайд открытой презентации, необходимо выполнить следующие действия:

1) Сделать текущим тот слайд, в который добавить звук.

2) Выбрать пункт меню **вставка /фильм и звук**.

3) Если требуется добавить звук из коллекции Clip Gallery, то затем следует вставить установочный компакт – диск в CD-ROM и выбрать после этого в открытом подменю команду **звук из коллекции**.

В появившемся диалоговом окне вставка звука следует открыть требуемую категорию и выбрать нужный звуковой эффект.

С другой стороны, если необходимо добавить в слайд звуковой файл, имеющийся на диске, то в этом случае следует воспользоваться командой *звук из файла*. Затем нужно выбрать среди имеющихся на диске звуковых файлов тот, который требует добавить в слайд, и нажать кнопку *ОК*.

Перед добавлением звука в слайд автоматически появляется окно предупреждения, в котором можно выбрать режим обработки данного эффекта (будет ли звук воспроизводиться автоматически при показе презентации).

Добавленный звук отображается на слайде в виде значка с изображением динамика. Если при вставке звука не был установлен режим автоматического воспроизведения, то для запуска этого звукового файла необходимо выполнить щелчок на его значке.

Можно заставить звук воспроизводиться автоматически, настроив для слайда соответствующие параметры в окне диалога *Настройка анимации*.

Необходимо отметить, что для звуковых файлов существует два основных стандарта: MIDI и WAV. Соответственно, данные файлы имеют расширение \*.mid и \*.wav. Кроме этого есть популярный формат MP3 (расширение \*.mp3)

Чтобы воспроизвести элементы мультимедиа надо выделить нужный элемент в списке *Порядок анимации* и раскрыв вкладку *Параметры воспроизведения* окна диалога *Настройка анимации*, можно изменить способ воспроизведения этого объекта. Имеется возможность приостановить демонстрацию слайд – фильма до окончания воспроизведения мультимедиа – объекта или продолжать демонстрацию на фоне воспроизведения музыки или повествования диктора.

Чтобы значок объекта мультимедиа не появлялся на экране во время демонстрации слайд – фильма, стоит лишь установить флажок *Скрывать, когда не воспроизведения*. При этом значок объекта будет виден в режиме просмотра слайдов, но не появится в процессе показа слайд – фильма.

**III.** Итог урока. Домашнее задание. Подготовить ответы на вопросы:

1. Для чего служит программа PowerPoint.
2. Какие способы создания презентаций существуют.
3. Какие основные этапы форматирования слайда\_

## Урок 2

**Тема урока:** Создание образовательных компьютерных презентаций. Демонстрация презентаций.

**Цель урока:** Учащиеся должны знать предназначение компьютерной презентации, способы их создания, правила работы с компьютерными презентациями, возможности применения разных типов объектов для создания презентации, способы демонстрации презентаций.

Учащиеся должны уметь запускать программу для создания презентации, создавать слайды, создавать образовательные компьютерные презентации из разных школьных дисциплин, демонстрировать созданные презентации, записывать их на дискеты.

**Оборудование:** Программа PowerPoint пакета Microsoft Office.

### Ход урока

I. Проверка домашнего задания.

1. Для чего служит программа PowerPoint.
2. Какие способы создания презентаций существуют.
3. Какие основные этапы форматирования слайда\_

## II. Изучение нового материала

### 1) Анимация слайдов

В ходе демонстрации слайд – фильма абзацы маркированных списков, а также размещенные на слайдах графические объекты могут появляться на экране либо все сразу, либо последовательно. Чтобы маркированный текст постепенно «выстраивался» на слайде, необходимо назначить ему тот или иной эффект анимации.

В режиме сортировщика слайдов можно быстро выбрать и применить готовые, заранее встроенные в программу эффекты анимации текстовых слайдов, так чтобы заголовки и абзацы маркированного списка появлялись последовательно (по заранее продуманной схеме). Для включения такого режима демонстрации надо выделить один или несколько слайдов с маркированным списком, а затем выбрать вариант анимации в раскрывающемся списке *Встроенная анимация текста* панели инструментов *Сортировщик слайдов* или в подменю *Встроенная анимация текста* контекстного меню.

Для расширенного управления анимацией слайдов и для настройки эффектов анимации графических объектов необходимо дважды щелкнуть на слайде, чтобы перейти в режим просмотра слайдов, где и осуществляется дальнейшая настройка.

В режиме слайдов щелкнуть на фоне слайда, чтобы отменить выделение всех объектов. Затем выполнить команду *Настройка анимации* меню *Показ слайдов*. Появится окно диалога *Настройка анимации*.

Щелкнув на кнопке *Эффекты анимации* панели инструментов *Форматирование*, можно вывести на экран панель инструментов *Эффект анимации*. Тот же результат можно получить, указав на подменю *Панели инструментов* меню *Вид* и щелкнув затем на пункте *Эффекты анимации*. Этой панелью инструментов можно пользоваться для анимации отдельных объектов.

На вкладке *Время* окна диалога *Настройка анимации* расположен список объектов, которым ещё не назначен эффект анимации. Можно выделить любой объект и щелкнуть на переключателе включить в разделе *Анимация* вкладки *Время*. По умолчанию для запуска анимации объекта требуется щелкнуть на нем кнопкой мыши, однако можно включить автоматическое инициирование анимации объекта через определенное число секунд после окончания предыдущего события или после появления слайда на экране, если до этого в слайд – фильме не производилась анимация других объектов.

Указав объекты для анимации, можно изменить ее порядок с помощью списка *Порядок анимации* того же окна диалога.

Теперь можно выбрать конкретный эффект анимации для каждого объекта. Нужно щелкнуть в окне диалога *Настройка анимации* на ярлык вкладки *Эффекты*. Для отмеченного объекта можно указать эффект анимации, звука, воспроизводимый при появлении объекта, и цвет, с помощью которого объект будет выделен с момента появления на экране до начала анимации следующего объекта.

Если объект является текстом, можно осуществить настройку параметров раздела *Появление текста* вкладки *Эффекты*. Текст может появляться в режимах *Все вместе*, по словам и по буквам.

### 2) Анимация диаграмм.

Если на слайде размещена диаграмма, можно назначить анимацию некоторым ее элементам. например, имеется возможность настроить столбцы гистограммы так, чтобы они появлялись на экране один за другим. Чтобы назначить эффекты анимации элементам диаграмм, надо отметить ее в списке *Порядок анимации* окна диалога *Настройка*

анимации после чего щелкнуть на ярлыке вкладки *Эффекты в диаграммах*.

На вкладке *Эффекты в диаграммах* надо настроить параметры раздела *Вывод элементов диаграммы* и установить флажок *Анимация сетки и легенды*, если необходимо, чтобы эффекты анимации распространялись также и на вспомогательные элементы диаграммы. Для элементов диаграмм предусмотрено лишь небольшое число различных эффектов анимации, однако у пользователя все же имеется некоторый выбор допустимых вариантов, перечисленных на вкладке *Эффекты в диаграммах*. Кроме того, можно подобрать звук, который будет воспроизводиться при появлении элементов диаграммы. Параметр раздела *После анимации* устанавливает эффект, который будет выполняться после появления на экране целиком все диаграммы.

Чтобы просмотреть эффекты анимации слайдов, надо вывести на экран нужный слайд в режиме просмотра слайдов и выполнить команду *Просмотр анимации* меню *Показ слайдов*. Для демонстрации анимации на экране появится небольшое цветное окно.

Если в режиме слайдов перед выполнением команды *Настройка анимации* щелкнуть на каком-либо объекте, то этот объект в окне диалога *Настройка анимации* также окажется выделенным и можно будет сразу приступить к настройке параметров его анимации на вкладке *Эффекты*.

### 3) Создание слайд - фильма

Апофеоз всей тяжести работы с Power Point наступает в момент демонстрации презентации, когда слайды один за другим выводятся на экран.

Создавая основную последовательность слайдов, тем самым формируется простейший слайд - фильм. Чтобы просмотреть его, надо перейти к первому слайду, а затем щелкнуть на кнопке *Показ слайдов*, расположенной в нижней части окна презентации, либо выполнить команду *Показ слайдов* меню *Вид*. Первый слайд развернется на весь экран и будет оставаться на нем до тех пор, пока не будет дана команда на переход к следующему слайду.

Переходы - это специальные эффекты, которые сопровождают процесс смены слайдов на экране. При смене слайдов в ходе демонстрации слайд - фильма эффект перехода поможет вывести на экран следующий слайд с применением самых разнообразных способов. Можно даже включать звуковое сопровождение процесса смены слайдов.

Эффекты перехода несомненно впечатляют, однако не стоит отвлекать внимание аудитории от содержания презентации перегруженностью эффектов.

Чтобы красиво закончить слайд - фильм, необходимо добавить в конец презентации черный слайд, выполнив команду *Параметры* меню *Сервис*, а затем установив флажок *Завершить черным слайдом* на вкладке *Вид* окна диалога *Параметры*. Нужно обратить внимание на то, что если не включить черный слайд в конец презентации, то после завершающего слайда на экране появится окно Power Point и аудитория неожиданно окажется лицом к лицу с кучей кнопок, меню и панелей инструментов.

### 4) Способы передачи презентаций

После того как выполнены все действия желательно сохранить презентацию на жестком диске или съемном носителе. Сохранение на жестком диске позволяет работать с большими файлами и ускоряет считывание информации с них. Использование съемных носителей (например, дискета) позволяет открыть презентацию на других

компьютерах. Но дискеты ограничивают размеры файла документа (1,44 Мб).

Чтобы сохранить презентацию, проще всего щелкнуть на кнопку Сохранить на панели инструментов Стандартная. Более сложный путь сохранения это Сохранить как... На панели инструментов открыть окно Файл, выбрать Сохранить как ... будет предложено сохранить по умолчанию в папке Мои Документы, но можно самому выбрать папку, или создать новую папку с новым названием, куда поместить сохраняемую презентацию.

После того, как файл сохранился, его теперь можно открывать всякий раз при очередном сеансе работы Power Point. Процедура открытия файла с презентацией ничем не отличается от процедуры открытия файлов и документов других программ.

### III. Итоги урока. Выводы

1. Презентацией называется набор слайдов, содержащих мультимедийные объекты: числа, текст, графику, анимацию, звук и видео, и сопровождаемые необходимыми комментариями.

2. Электронные страницы презентации называются слайдами.

3. Презентация можно создать одним из следующих способов:

- с помощью Мастера создания презентаций;
- используя готовые шаблоны оформления;
- самостоятельное создание презентации.

4. Для работы со слайдами используют несколько режимов:

- команда **Обычный** включает режим редактирования слайда;
- **Сортировщик слайдов** показывает сразу все слайды небольшого размера на экране;
- **Страницы заметок** изображает страницу с рисунком слайда и местом для текста примечания;
- **Демонстрация** запускает демонстрацию презентации.

5. Для оформления презентации в едином стиле используют команду **Вид/Образец/Образец слайдов**.

6. Чтобы применить заготовки фона вызовите команду **Формат/Фон**.

7. Однако чтобы качественно оформить свою презентацию, лучше воспользоваться шаблоном, который будет содержать свою цветовую схему, образцы заголовков и слайдов, а также множество авторазметок, определяющих расположение объектов на слайде. Это осуществляется с помощью команды **Формат/Применить шаблон дизайна**.

8. После завершения оформления слайдов презентации можно выполнить анимацию отдельных объектов на слайдах. Для этого в режиме **Обычный** выбираем команду **Показ слайдов/Настройка анимации** и применяем эффекты к объектам.

9. Для настройки переходов между слайдами необходимо воспользоваться командой **Показ слайдов/Настройка презентации**, либо в режиме сортировщика слайдов панелью инструментов **Эффекты перехода**.

10. Для произвольного управления показа слайдов можно воспользоваться несколькими вариантами: установить настройки командой **Показ слайдов/Настройка действия**, использовать гиперссылки для перехода между слайдами либо установит кнопки управления.

### IV. Домашнее задание.

А. Подготовить ответы на вопросы:

1. Что такое анимация слайдов?
2. К каким объектам можно применять анимацию?
3. Как сохранять презентации?

В. Создать образовательную презентацию по теме.

**Дворядкина Ирина Эрастовна** – учитель информатики ОШ №15,  
**Смирнова Ирина Георгиевна** – учитель информатики ОШ№15

#### 4.4. Табличний процесор. (10 годин)

№ урока	Тема урока
1	Призначення та функції табличного процесора. Електронна таблиця та її основні об'єкти: файл, книга, аркуш, комірка, адреса комірки, діапазон комірок.
2	Введення і редагування даних різного типу. Копіювання, переміщення, вилучення, форматування таблиці.
3	Практична робота «Введення і редагування даних різного типу. Копіювання, переміщення, вилучення, форматування таблиці»
4	Виконання обчислень в середовищі табличного процесора. Використання функцій та операцій для опрацювання даних, поданих у таблиці.
5	Практична робота «Виконання обчислень в середовищі табличного процесора. Використання функцій та операцій для опрацювання даних, поданих у таблиці»
5*	Використання функцій під час опрацювання табличних величин. Логічні функції. (Для фізико-математичного напрямку за рахунок резервного часу)
6	Ділова графіка. Побудова діаграм і графіків на основі табличних даних.
7	Практична робота «Ділова графіка. Побудова діаграм і графіків на основі табличних даних»
8	Упорядкування і пошук потрібних даних в електронній таблиці. Фільтрування даних. Виведення табличних даних і діаграм на друк.
9	Практична робота «Упорядкування і пошук потрібних даних в електронній таблиці. Фільтрування даних. Виведення табличних даних і діаграм на друк»
10	Тематична атестація з теми «Електронні таблиці»

Для забезпечення вивчення теми необхідний програмний засіб: Microsoft Excel.

#### Урок 1

**Тема урока:** Електронные таблицы. Назначение и функции табличного процессора. Электронная таблица и ее основные объекты: файл, книга, лист, ячейка, адрес ячейки, диапазон ячеек.

**Тип урока:** Урок изучения нового учебного материала и первичного закрепления.

#### Цель урока:

- повторить понятие «информационная система (ИС)» и «составляющие ИС»;
- продолжить знакомство с программной составляющей ИС;
- повторить назначение основных типов ППП;
- ввести понятие «электронная таблица»;
- рассмотреть разнообразие программ, работающих с таблицами;
- продолжить знакомство с ППП Microsoft Office;
- формировать навыки работы учащихся с одной из составляющих ППП Microsoft Office – программой Microsoft Excel;
- продолжить формирование информационной культуры учащихся;
- развивать память, внимание;
- воспитывать инициативность, заинтересованность к предмету;

-воспитывать эмоционально позитивную направленность на практическую деятельность.

**Дополнительная подготовка:** наличие Microsoft Excel.

#### **Структура урока.**

- I. Оргмомент (1 мин).
- II. Актуализация опорных знаний (10 мин).
- III. Мотивация учебной деятельности учащихся, сообщение темы, цели урока, его задач (5 мин).
- IV. Изучение нового материала (15 мин).
- V. Практическая часть (10 мин).
- VI. Подведение итогов урока (3 мин).
- VII. Домашнее задание (1мин).

#### **Содержание урока.**

- I. Оргмомент.  
Отметить отсутствующих, проверить наличие учебных принадлежностей, записать дату проведения урока в тетрадях.
- II. Актуализация опорных знаний.
  1. Повторить понятие «информационная система (ИС)» и «составляющие ИС» – восстановить цепочку: ИС – программное обеспечение – ППП – ППП Microsoft Office.
  2. Повторить понятия: приложение, документ.
  3. Повторить назначение основных типов ППП.
- III. Мотивация учебной деятельности учащихся, сообщение темы, цели урока, его задач:
  1. На данный момент учащиеся знакомы с программами, обрабатывающими текст, графику. Цель данного урока: Познакомить учащихся с программами, работающими с массивами чисел.
  2. Сообщение темы урока «Электронные таблицы. Электронные таблицы. Назначение и функции табличного процессора. Электронная таблица и ее основные объекты: файл, книга, лист, ячейка, адрес ячейки, диапазон ячеек». Запись темы в тетрадях.
  3. Рассмотреть разнообразие прикладных задач, решаемых с помощью большого количества вычислений с большими числовыми данными. Области применения электронных таблиц:
    - бухгалтерский и плановый учет;
    - проектно-сметные работы;
    - планирование распределения ресурсов;
    - инженерно-технические расчеты;
    - обработка больших массивов статистической информации.
    - математическое сопровождение задач из физики, астрономии, химии и т.п.
  4. Рассмотреть известные на сегодняшний день программы – «Электронные таблицы».
  5. Тему будем изучать на примере распространенной на сегодняшний день программой Microsoft Excel ППП Microsoft Office.
- IV. Изучение нового материала:
  1. Дать определение: «Электронная таблица (ЭТ) – это программа для обработки данных представленных в виде таблицы».
  2. Рассмотреть назначение и функции ЭТ.  
Основные возможности электронных таблиц:
    - удобство ввода и редактирования данных и формул;
    - проведение вычислений (использование стандартных функций,

автоперерасчет) по формулам, заданным пользователем:

- вставка в таблицу фрагментов из документов, созданных в других программных средствах;
- представление результатов расчетов как в числовом виде, так и с помощью деловой графики (гистограммы, разнообразные диаграммы, графики зависимостей);
- оформление таблиц, отчетов, печать итоговых документов;
- создание и ведение баз данных с возможностью выбора записей по заданному критерию и сортировки по любому параметру;
- коллективная работа с табличными данными в сетях.

3. Ввести понятия: **приложение** – программа **Microsoft Excel**, **документ** – файл с расширением **.xls** — «книга».

4. Рассмотреть способы загрузки программы Microsoft Excel:

- через ярлык на «Рабочем столе»;
- через основное меню «Пуск»-Программы-...
- с помощью поиска через программу «Проводник»;
- через вызов контекстного меню «Создать» ...
- через ранее созданный документ и т.д.

5. Повторить понятие «пользовательский интерфейс».

6. Повторить понятия: «окно приложения», «окно документа», их составляющие (заголовок, кнопки управления, рабочее поле, горизонтальное меню, система ниспадающих меню, панель инструментов, строка состояния, полосы прокрутки и т.д.).

7. Ввести понятия основных объектов ЭТ:

- ячейка; адрес ячейки;
- столбец; имена столбцов; строка; номера строк;
- диапазон; диапазон строк, столбцов, ячеек, их адресация;
- активная ячейка, активный столбец, активная строка...
- лист.

**V.** Практическая часть (учащиеся берут с собой конспекты для записи необходимой дополнительной информации):

1. Загрузить программу Microsoft Excel всевозможными способами, рассмотренными на уроке.

2. Работа с окном приложения Microsoft Excel (работа в синхронном режиме – учитель называет объект, – учащиеся «следят» указателем «мыши», выделяют объект:

- имя приложения. Кнопки управления окнами;
- горизонтальное меню (Файл, Правка, Вид и т.д.);
- панель инструментов;
- строка состояния;
- поле имени;
- строка формул;
- окно документа;
- полосы прокрутки;
- кнопки прокрутки;
- заголовки строк, столбцов; определить максимальное количество строк; определить имя последнего столбца;
- активная (текущая) ячейка: определение адреса;
- выделение ячейки, строки, столбца, их диапазонов;



определение их адресов;

– ярлыки листов;

3. Сохранение документа.

4. Завершение работы с программой Microsoft Excel.

**VI.** Подведение итогов урока:

1. Повторили...

2. Рассмотрели...

3. Научились...

**VII.** Домашнее задание: I.Т.Зарецька, Б.Г.Колодяжний, А.М.Гуржій, О.Ю.Соколов. Информатика 10-11.

Глава 3, п 3.2, стр. 97-99 (отвечать на вопросы 1-4).

Глава 7, §1,2, стр. 379-386 (отвечать на вопросы).

## Урок 2

**Тема урока:** Введение и редактирование данных разного типа. Копирование, перемещение, удаление, форматирование таблицы.

**Тип урока:** Урок формирования умений и навыков.

**Цель урока:**

- продолжить формирование навыков работы учащихся с одной из составляющих ППП Microsoft Office – Microsoft Excel;
- рассмотреть форматы величин, с которыми может работать ЭТ Microsoft Excel (работа в меню «Формат»-«Ячейка»).
- научить учащихся:
  - вводить данные в ячейку ЭТ;
  - редактировать содержимое ячейки;
  - форматировать элементы ЭТ согласно вводимым данным;
- научить учащихся форматировать таблицу:
  - работа с меню «Границы»;
  - работа с меню «Заливка»;
  - изменение размеров строк, столбцов.
- продолжить формирование информационной культуры учащихся;
- развивать навыки применения компьютерных технологий;
- воспитывать трудолюбие, усидчивость.

**Дополнительная подготовка:** наличие ППП Microsoft Excel.

**Дидактические средства:** Ксерокопии страниц географического атласа с данными по материкам.

### Структура урока.

I. Оргмомент (1 мин).

II. Актуализация опорных знаний (10 мин).

III. Мотивация учебной деятельности учащихся, сообщение темы, цели урока, его задач (10 мин).

IV. Изучение нового материала. Практическая работа (20 мин).

V. Подведение итогов урока (3 мин).

VI. Д/з (1мин).

### Содержание урока.

**I.** Оргмомент.

Отметить отсутствующих, проверить наличие учебных принадлежностей, записать дату проведения урока в тетрадях.

**II.** Актуализация опорных знаний.

1. Повторить понятие «электронная таблица» (ЭТ).

2. Повторить об основных объектах ЭТ (ячейка, строка, стол-

бец, диапазон).

3. Привести примеры использования ЭТ. Назначение и функции ЭТ.
4. Способы загрузки ЭТ Microsoft Excel.
5. Сохранение готового документа.
6. Завершение работы с очередным документом.

**III.** Мотивация учебной деятельности учащихся, сообщение темы, цели урока, его задач:

1. Типы величин, с которыми необходимо производить действия: текстовые, числовые, дата, время, финансовые и т.д.

2. Сообщение темы «Введение и редактирование данных разного типа. Копирование, перемещение, удаление, форматирование таблицы.»

**IV.** Изучение нового материала (Практическая часть). (Создается пробная таблица, на которой отрабатываются в синхронном режиме все возможности ЭТ Microsoft Excel):

1. Работа с меню «Границы» панели инструментов.
  - расчет количества строк и столбцов в таблице;
  - выделение диапазона клеток, для оформления таблицы;
  - оформление границ ячеек;
2. Заполнение ячеек данными (любыми – текст, число, дата, ...).
3. Работа с режимом «Формат ячейки» панели инструментов.
  - рассмотреть все вкладки: Число, Выравнивание, Шрифт, Граница, Вид, Защита.
4. Ввести понятие: «многослойная структура ячейки».
5. Изменение размеров строк и столбцов.  
Чтобы изменить ширину столбца нужно:
  - установить указатель мыши на правую границу столбца в строке заголовков столбцов;
  - когда указатель изменит свою форму на разнонаправленные стрелки, нажать на левую кнопку мыши и, не отпуская её, переместить указатель на нужное расстояние вправо (для увеличения ширины столбца) или влево (для уменьшения ширины столбца);
  - отпустить кнопку мыши.

Аналогично изменяется ширина строки (указатель мыши ставится на нижнюю границу строки).

6. Отработать режимы редактирования содержимого ячейки:
  - двойным щелчком мыши по ячейке;
  - через **Строку формул**.
7. Рассмотреть возможность копирования, перемещения, удаления содержимого ячейки:
  - с помощью **Основного (Горизонтального) меню**;
  - с помощью **Панели инструментов**;
  - через **Контекстное меню** (правая кнопка мыши).
8. Ввести понятие «маркер заполнения».
9. Применить «маркер заполнения» для нумерации строк, столбцов таблицы, заполнения ячеек последовательностями (элементами арифметической прогрессии: число, дата, дни недели, ...).
10. Рассмотреть возможности заливки ячеек таблицы различными цветами краски («Заливка»).

**V.** Подведение итогов урока: Повторили..., Узнали..., Научились...

**VI.** Домашнее задание:

Глава 3, п 3.2, стр. 97-99 (отвечать на вопросы 1-4).

Глава 7, § §1-3 (отвечать на вопросы).

### Урок 3

**Тема урока:** Практическая работа: «Введение и редактирование данных разного типа. Копирование, перемещение, удаление, форматирование таблицы».

**Тип урока:** Урок комплексного применения ЗУН учащихся.

- продолжить формирование навыков работы учащихся с программой Microsoft Excel;
- Усвоение умений самостоятельно в комплексе применять:
  - знания по ЭТ;
  - умения работать с элементами ЭТ Microsoft Excel;
  - навыки работы на компьютере;
- отработать:
  - навыки работы с форматами величин, с которыми может работать ЭТ Microsoft Excel (работа в меню «Формат»-«Ячейка»).
  - ввод данных в ячейку ЭТ;
  - редактирование содержимого ячеек;
  - форматировать элементы ЭТ согласно вводимым данным;
- побуждать учащихся к самостоятельной творческой деятельности;
- активизировать процесс познания;
- воспитывать инициативность, ответственность.

**Дополнительная подготовка:** Наличие Microsoft Excel, раздаточного материала.

**Дидактические средства:** Ксерокопии страниц географического атласа с данными по материкам.

#### Структура урока

I. Оргмомент (1 мин).

II. Актуализация ЗУН, необходимых для творческого применения знаний. Мотивация. (15 мин).

III. Применение обобщенных ЗУН. Практическая работа (20 мин).

IV. Контроль (самоконтроль) знаний, умений, навыков (5 мин).

V. Подведение итогов урока (3 мин).

VI. Домашнее задание (1мин).

#### Содержание урока

I. Оргмомент.

Отметить отсутствующих, проверить наличие учебных принадлежностей, записать дату проведения урока в тетрадях.

II. Актуализация ЗУН, необходимых для творческого применения знаний.

1. Повторить понятие «электронная таблица» (ЭТ).
2. Повторить об основных объектах ЭТ (ячейка, строка, столбец, диапазон и т.п.).
3. Привести примеры использования ЭТ.
4. Назначение и функции ЭТ.
5. Способы загрузки ЭТ Microsoft Excel.
6. Сохранение готового документа.
7. Завершение работы с очередным документом.
8. Повторить типы величин, с которыми необходимо производить

действия: текстовые, числовые, дата, время, финансовые и т.д.)

### III. Применение обобщенных ЗУН. Практическая работа (20 мин).

1. Сообщение темы Практической работы: «Введение и редактирование данных разного типа. Копирование, перемещение, удаление, форматирование таблицы».
2. Раздать учащимся ксерокопии страниц географического атласа с данными по частям света (Европа, Азия, Северная Америка, Южная Америка, Африка, ...)
3. Рассмотреть данные, приведенные в таблицах по каждой части света.
4. Рассадить за компьютеры. Напомнить о правилах поведения, ТБ.
5. Предложить создать таблицу с именем *Государства Европы* (Африки, Америки, ...) с графами: название государства; столица государства; площадь; население.
6. Заполнить таблицу на 11-15 государств.
7. Распределить столбцы таблицы так, чтобы она помешалась на

	A	B	C	D	E	F
1	Государства Европы					
2						
3	№ п/п	Название страны	Столица	Площадь, в тыс кв.км.	Население, в тыс.чел	
4	1	Австрия	Вена	84	7513	
5	2	Албания	Тиара	29	2259	
6	3	Бельгия	Брюссель	30	9823	
7	4	Болгария	София	111	8825	
8	5	Великобритания	Лондон	244	55928	
9	6	Венгрия	Будапешт	93	10671	
10	7	Греция	Афины	132	9280	
11	8	Исландия	Рейкьявик	70	220	
12	9	Испания	Мадрид	505	36230	
13	10	Польша	Варшава	313	35000	
14	11	Португалия	Лиссабон	92	9730	
15	12					

видимой части поля документа.

8. Рассмотреть возможности заливки ячеек таблицы различными цветами краски («Заливка»).
9. Сохранить Книгу под именем *Государства Европы* (Африки, Америки, ...) *Иванова* (Петрова, Сидорова, ...).
10. Научить объединять ячейки, помещать содержимое по центру, переносить содержимое по словам, ...
11. Результат практической работы:

### IV. Контроль знаний, умений, навыков:

1. оценить работу каждого ученика (по-возможности);
2. выставить оценки за Практическую работу

### V. Подведение итогов урока: Повторили..., Узнали..., Научились..., Закрепили ..., Отработали ... .

### VI. Домашнее задание:

- Глава 3, п 3.2, стр. 97-99 (отвечать на вопросы 1-4).  
Глава 7, § §1-4 (отвечать на вопросы).

## Урок 4

**Тема урока:** Выполнение вычислений в среде табличного процессора. Использование функций и операций для обработки данных, представленных в таблице.

**Тип урока:** Урок изучения нового учебного материала.

### Цель урока:

- продолжить знакомство с ППП Microsoft Office;
- формировать навыки работы учащихся с Microsoft Excel;
- ознакомить учащихся с возможностями программы Microsoft Excel при выполнении вычислений.

- ознакомить учащихся с перечнем встроенных функций и способами их применения;
- формировать логическое и алгоритмическое мышление;
- развивать познавательный интерес к предмету;
- развивать умения оперировать ранее полученными знаниями;
- продолжить формирование информационной культуры учащихся;
- развивать память, внимание;
- воспитывать инициативность, заинтересованность к предмету;
- воспитывать эмоционально позитивную направленность на практическую деятельность.

**Дополнительная подготовка:** Наличие программы Microsoft Excel.

#### **Структура урока.**

- I. Оргмомент (1 мин).
- II. Мотивация учебной деятельности учащихся, сообщение темы, цели урока, его задач (5 мин).
- III. Изучение нового материала (18 мин).
- IV. Закрепление. Практическая часть (15 мин).
- V. Подведение итогов урока (5 мин).
- VI. Домашнее задание (1 мин).

#### **Содержание урока.**

- I. Оргмомент. Отметить отсутствующих, проверить наличие учебных принадлежностей, записать дату проведения урока в тетрадях.
- II. Мотивация учебной деятельности учащихся, сообщение темы, цели урока, его задач:
  1. программа Microsoft Excel предназначена в основном для вычислений. Наша задача научиться быстро и правильно записывать выражения, используя операции, предусмотренные в ТП;
  2. для оптимального использования возможностей ТП необходимо научиться пользоваться встроенными в программу функциями;
  3. сообщение темы урока, её записи в тетрадях;
  4. постановка целей и задач урока.
- III. Изучение нового материала:
  1. Ввести понятие «формула». Рассмотреть формулу как частный случай арифметического выражения:
    - начинается со знака равенства;
    - используются знаки арифметических операций;
    - напомнить о порядке выполнения арифметических операций и применении скобок;
    - предупредить о возможных ошибочных значениях при вычислениях значений по формуле (см. стр. 391 учебника);
  2. Для облегчения расчетов в табличном процессоре Excel есть встроенные функции:
    - для удобства выбора и обращения к ним, все функции объединены в группы, называемые категориями: математические, статистические, финансовые, функции даты и времени, логические, текстовые и т.д;
    - каждая функция имеет свое неповторимое (уникальное) имя;
    - при обращении к функции после ее имени в круглых скобках указывается список аргументов, разделенных точкой с запятой;
    - ввод функции в ячейку надо начать со знака «=», а затем указать ее имя.
  3. Рассмотреть группы функций (раздаточный материал с таблицами): Математические функции

Название и обозначение функции	Имя функции	Пример записи функции	Примечание
Синус - sin(x)	SIN(...)	SIN(A5)	Содержимое ячеек A5 в радианах
Косинус- cos(x)	COS(...)	COS(B2)	Содержимое ячейки B2 в радианах
Тангенс - tan	TAN(...)	TAN(B5)	Содержимое ячейки B5 в радианах
Квадратный корень - корень	КОРЕНЬ (...)	КОРЕНЬ(D12)	Содержимое ячейки D12>0
Преобразует радианы в градусы - градусы	ГРАДУСЫ (...)	ГРАДУСЫ (C8)	Содержимое ячейки C8 в градусах
Сумма - сумм	СУММ(...)	СУММ(A1;B9)	Сложение двух чисел, содержащихся в ячейках A1 и B9
		СУММ(A1:A20)	Сложение всех чисел, из диапазона от A1 до A20
Число $\pi$ - Пи	ПИ ()	ПИ()	Функция не содержит аргументов

#### Статистические функции

Максимальное значение - макс	МАКС(...)	МАКС(A1:A9)	Поиск максимального среди аргументов
Минимальное значение - мин	МИН(...)	МИН(C1:C23)	Поиск минимального среди аргументов
Среднее значение - срзнач	СРЗНАЧ(..)	СРЗНАЧ(A1:B5)	Находит среднее арифметическое значение среди чисел, диапазона от A1 до B5

#### Текстовые функции

Название и обозначение функции	Имя функции	Пример записи функции	Примечание
Объединяет несколько текстовых элементов в один - сцепить	СЦЕПИТЬ(...)	СЦЕПИТЬ(B9;B14)	Чтобы добавить пробел между сцепленными словами, СЦЕПИТЬ (B9;" ";B14)
Повторяет текст заданное число раз - повтор	ПОВТОР(...)	ПОВТОР(B4;5)	Повторяет текст, содержащийся в ячейке B4 пять раз
Находит крайние левые символы строки - левсимв	ЛЕВСИМВ(...)	ЛЕВСИМВ(A1;1)	Отображает только первую букву текста, содержащегося в ячейке A1.
Делает все буквы в тексте строчными - строчн	СТРОЧН(...)	СТРОЧН(A2:A9)	Все слова, содержащиеся в диапазоне ячеек от A2 до A9 будут написаны строчными (маленькими буквами)

#### IV. Закрепление. Практическая часть.

Раздать карточки с заданием.

#### Задание

D9     =СУММ(D3:D8)

	A	B	C	D
1	<b>Набор первоклассника</b>			
2	<b>Наименование</b>	<b>Цена</b>	<b>Количество</b>	<b>Сумма</b>
3	Тетрадь	6,50р.	12	78,00р.
4	Ручка	3,40р.	3	10,20р.
5	Карандаш	1,20р.	2	2,40р.
6	Линейка	8,55р.	1	8,55р.
7	Альбом	15,10р.	2	30,20р.
8	Обложки	4,80р.	22	105,60р.
9			<b>Итого</b>	<b>234,95р.</b>
10				

1. Введите список предметов из набора первоклассника.
2. Установите денежный формат данных в диапазоне ячеек B3:B8 и введите цену на каждый предмет из набора первоклассника.
3. Введите количество о предметов.
4. Используя формулу (подумайте какую) рассчитайте стоимость всех тетрадей, всех ручек, всех карандашей и т.п.
5. Используя математическую функцию суммы, рассчитайте общую сумму, затраченную на покупку набора для первоклассника.
6. Отформатируйте таблицу по образцу.

V. Подведение итогов урока: Повторили..., Узнали..., Научились..., Закрепили ...

VI. Домашнее задание: Глава 3, п 3.2, стр. 97-99 (отвечать на вопросы 1-4). Глава 7, §3 (отвечать на вопросы).

## Урок 5

**Тема урока:** Практическая работа «Выполнение вычислений в среде табличного процессора. Использование функций и операций для обработки данных, представленных в таблице».

**Тип урока:** Урок комплексного применения ЗУН учащихся.

**Цель урока:**

- продолжить формирование навыков работы учащихся с МExcel;
- усвоение умений самостоятельно в комплексе применять:
  - знания по ЭТ;
  - умения работать с элементами ЭТ Microsoft Excel;
  - навыки работы на компьютере;
- отработать:
  - навыки работы с операциями обработки чисел, использования функций (математических);
  - добавление столбцов;
  - редактирование содержимого ячеек;
  - форматирование элементов ЭТ согласно полученным данным;
- развивать умения планировать свою деятельность;
- воспитать умения самостоятельно мыслить, ответственность за выполняемую работу, аккуратность при выполнении работы.

**Дополнительная подготовка:** Наличие Microsoft Excel.

### Структура урока.

- I. Оргмомент (1 мин).
- II. Актуализация ЗУН, необходимых для творческого применения знаний (10 мин).
- III. Объяснение понятий «относительный адрес», «абсолютный адрес». Мотивация их применения (10 мин).
- IV. Применение обобщенных ЗУН. Практическая работа (20 мин).
- V. Контроль (самоконтроль) знаний, умений, навыков.
- VI. Подведение итогов урока (3 мин).
- VII. Домашнее задание (1 мин).

### Содержание урока.

- I. Оргмомент.  
Отметить отсутствующих, проверить наличие учебных принадлежностей, записать дату проведения урока в тетрадах.
- II. Актуализация ЗУН, необходимых для творческого их применения.
  1. правила записи формул;
  2. математические (арифметические) операции их приоритет;
  3. группы встроенных функций, их применение.
- III. Объяснение понятий «относительный адрес», «абсолютный адрес». Мотивация их применения
  1. Вводятся понятия «относительный адрес», «абсолютный адрес»;

	А	В	С	Строка формул	Е	Ф
1				Реализация книг		
2						
3	Продавец	количество книг	стоимость книги	вырученные деньги	вырученные деньги	%
4	Иванов	20	3,5	70	70	23,8
5	Петров	34		0	119	40,5
6	Смирнов	30		0	105	35,7
7						
8					общая выручка	
9			25	70	294	
10						

2. Сравните представленные таблицы (раздаточный материал). Ответить на вопросы:

- В каком столбце (D или E) верно посчитаны вырученные каждым продавцом деньги?
- Как бы вы заполняли ячейки с формулами в столбцах D и E? (подвести учащихся к выводу о том, что в Excel позволяет скопировать готовую формулу в смежные ячейки, при этом адреса ячеек будут изменены автоматически).
- Чем отличаются формулы в столбцах D и E? Почему?

	Имя	B	C	D	E	F
1	Реализация книг					
2						
3	Продавец	количество книг	стоимость книги	вырученные деньги	вырученные деньги	%
4	Иванов	20	3,5	=B4*C4	=B4*\$C\$4	=E4/\$E\$9*100
5	Петров	34		=B5*C5	=B5*\$C\$4	=E5/\$E\$9*100
6	Смирнов	30		=B6*C6	=B6*\$C\$4	=E6/\$E\$9*100
7						
8				общая выручка		
9				=СУММ(D4:D6)	=СУММ(E4:E6)	
10						

3. Пришли к выводу:

- Использование относительных и абсолютных адресов позволяет оптимизировать работу в Excel.

#### IV. Применение обобщенных ЗУН. Практическая работа.

1. Продолжить работу с документом Государства Европы (Африки, Америки, ...) Иванова (Петрова, Сидорова, ...).
2. Добавить столбец «Плотность населения, в тыс.чел на кв.км». Посчитать для каждого государства.
3. Посчитать общую площадь государств, общую численность населения (применив функцию СУММ).
4. Посчитать среднюю плотность населения.

A1		Государства Европы						
A	B	C	D	E	F	G	H	
1	Государства Европы							
2								
3	№ п/п	Название страны	Столица	Площадь, в тыс кв.км.	% по отношению к общей площади	Население, в тыс.чел	% по отношению к общей численности	Плотность населения, в чел на кв.км.
4	1	Австрия	Вена	84	4,9	7513	4,1	89,4
5	2	Албания	Тиара	29	1,7	2259	1,2	77,9
6	3	Бельгия	Брюссель	30	1,8	9823	5,3	327,4
7	4	Болгария	София	111	6,5	8825	4,8	79,5
8	5	Великобритания	Лондон	244	14,3	55928	30,2	229,2
9	6	Венгрия	Будапешт	93	5,5	10671	5,8	114,7
10	7	Греция	Афины	132	7,8	9280	5,0	70,3
11	8	Исландия	Рейкьявик	70	4,1	220	0,1	3,1
12	9	Испания	Мадрид	505	29,7	36230	19,5	71,7
13	10	Польша	Варшава	313	18,4	35000	18,9	111,8
14	11	Португалия	Лиссабон	92	5,4	9730	5,2	105,8
15	12							
16	15							
17								
18				Общая площадь государств		Общая численность населения		Средняя плотность населения
19				1703		185479		108,9
20								



- V.** Контроль (самоконтроль) знаний, умений, навыков. Отобразить в режиме формул.
- VI.** Подведение итогов урока.
- VII.** Домашнее задание (1мин). Глава 3, п 3.2, стр. 97-99 (отвечать на вопросы 1-4). Глава 7, § §3 (отвечать на вопросы).

### **Урок № 5(\*)**

**Тема урока:** Использование функций для обработки табличных данных.  
Логические функции.

**Тип урока:** Урок объяснения нового материала.

#### **Цели и задачи урока:**

- ознакомить учащихся с применением функций для обработки табличных данных на примере решения задачи о нахождении площади треугольника;
- повторить учебный материал из курса геометрии;
- актуализировать опорные знания по теме;
- формировать навыки работы в EXCEL;
- развивать внимание, память и логическое мышление;
- воспитывать личную ответственность за результаты своей работы и за возможные ошибки;
- воспитывать взаимовыручку, умение работать в парах.

**Дидактический материал:** учебник Погорелова Геометрия 7-9.

#### **Структура урока.**

- I. Актуализация опорных знаний (5 мин.)
- II. Мотивация учебной деятельности учащихся и сообщение темы, целей и задач урока (2 мин.)
- III. Повторение материала из курса геометрии: задача о возможности построения треугольника по трём заданным сторонам (2 мин.)
- IV. Изучение нового материала (20 мин.)
- V. Практическое задание. Выполняет пара учеников за одним компьютером (15 мин.)
- VI. Подведение итогов урока и домашнее задание (1мин.)

#### **Содержание урока.**

- I. Фронтальный опрос.
- Каковы назначение и функции табличного процессора?
  - Что такое формула в Excel? Каковы её компоненты?
  - Какие из функций нам уже известны?
  - Как ввести функцию в формулу?
  - Как ввести формулу в ячейку?

**II.** Итак, мы уже знаем, что в «Excel» есть целый ряд встроенных функций. На предыдущих уроках мы познакомились с некоторыми из них. Но этот перечень был бы неполным без такой категории, как логические функции. Именно благодаря им, перечень задач, решаемых с помощью «Excel» может быть существенно расширен.

Тема урока: Использование функций для обработки табличных данных. Логические функции.

**III.** Применение логических функций мы рассмотрим с вами на примере решения задачи о нахождении площади треугольника по трём сторонам.

Формулировка задачи. Определить, существует ли треугольник, длины сторон которого совпадают с заданными числами. В случае утвердительного ответа вычислить площадь этого треугольника.

В этом примере строится таблица, исходными данными для которой являются три числа **a, b и c**. Таблица дает ответ на вопрос, существует ли треугольник, длины сторон которого совпадают с этими числами. Если такой треугольник существует, то в таблице вычисляются его полупериметр и площадь. Для существования треугольника необходимо и достаточно, чтобы одновременно выполнялись (т.е. были истинными) три неравенства, называемых неравенствами треугольника: **a < b+c; b < a+c; c < a+b**. Для проверки выполнения или невыполнения условий используются логические выражения и логические функции.

**IV.** Логическая функция **И(arg1,arg2,...)** возвращает истину, если значения всех её аргументов истинны.

Для выбора одного из двух вариантов действий в зависимости от выполнения или невыполнения какого-либо условия используется логическая функция **ЕСЛИ**.

У этой функции три аргумента **ЕСЛИ(arg1, arg2,arg3)**. Первый аргумент – это логическое выражение. Она возвращает значение второго аргумента, если значение первого – истина, и значение третьего – в противном случае.

Вычисление площади треугольника осуществляется по формуле Герона:

$$P = \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)}, \text{ где } p - \text{ полупериметр, т.е. } p = (a+b+c)/2$$

Для вычисления корня квадратного – функция **КОРЕНЬ**.

Эта функция возвращает значение квадратного корня из выражения, записанного в скобках.

Занесем данные в таблицу.

Таблица может выглядеть следующим образом:

<u>Вычисление площади треугольника</u>			
	<b>A</b>	<b>B</b>	<b>C</b>
<u>Стороны</u>	1	2	3
	<b>a &lt; b+c</b>	<b>b &lt; a+c</b>	<b>c &lt; a+b</b>
<u>Проверки</u>	ИСТИНА	ИСТИНА	ЛОЖЬ

Можно ли построить треугольник?

**НЕЛЬЗЯ**

Полупериметр

**Треугольник построить нельзя**

Площадь

**Треугольник построить нельзя**

Или таким образом, если треугольник существует:

Вычисление площади треугольника

	А	В	С
<u>Стороны</u>	5	6	7
	$a < b+c$	$b < a+c$	$c < a+b$
<u>Проверки</u>	ИСТИНА	ИСТИНА	ИСТИНА

Можно ли построить треугольник?

МОЖНО

Полупериметр

9

Площадь

14,69693846

Можно перейти в режим отображения формул. Для этого в пункте меню **СЕРВИС/ПАРАМЕТРЫ** в диалоговом окне нужно поставить флажок возле пункта **ФОРМУЛЫ**. В режиме отображения формул можно увидеть, что в ячейке D9 записана формула:  
 =ЕСЛИ(И(В7;С7;D7);"МОЖНО";"НЕЛЬЗЯ")

В ячейке C11 записана формула:

=ЕСЛИ(D9="МОЖНО";(В4+С4+D4)/2;"Треугольник построить нельзя")

В ячейке C13 записана формула:

=ЕСЛИ(D9="МОЖНО";КОРЕНЬ(С11\*(С11-В4)\*(С11-С4)\*(С11- D4)); "Треугольник построить нельзя")

Таким образом, при заданной каждый раз тройке значений **а, в** и **с** подготовленная таблица будет выдавать сообщение о том, можно ли построить треугольник, и, если можно, то рассчитывать его полупериметр и площадь по формуле Герона.

Теперь вы, работая в парах, выполните практическое задание по подготовке такой таблицы.

**V.** Запустите Excel и заполните таблицу следующим образом:

Адрес ячейки	Какие данные заносим в ячейку		
A4	Стороны		
A7	Проверки		
B1	Вычисление площади треугольника		
B3	А		
C3	В		
D3	С		
B4	5	или	1
C4	6		2
D4	7		3
B6	$a < b+c$		
C6	$b < a+c$		
D6	$c < a+b$		
B7	$=b5 < c5+d5$		
C7	$=c5 < b5+d5$		
D7	$=d5 < b5+c5$		
A9	Можно ли построить треугольник?		
A11	Полупериметр		
A13	Площадь		
D9	=ЕСЛИ (И (В9 ; С9 ; D9) ; "МОЖНО" ; "НЕЛЬЗЯ")		
C11	=ЕСЛИ (D11="МОЖНО" ; (В5+С5+D5) / 2 ; "Треугольник построить нельзя")		

C13	=ЕСЛИ(D11="МОЖНО";КОРЕНЬ(C13*(C13-B5)*(C13-C5)*(C13-D5));"Треугольник построить нельзя")
-----	--

Проверьте работу таблицы на разных значениях входных данных.

Покажите работу учителю.

**VI.** Домашнее задание:

Повторить главу 7, § 3. Выучить конспект. Составить алгоритм решения квадратного уравнения с использованием логических функций.

## Урок 6

**Тема урока:** Деловая графика. Построение диаграмм и графиков на основе табличных данных.

**Тип урока:** Урок изучения нового учебного материала и первичного закрепления.

**Цель урока:**

- продолжить формирование навыков работы с Microsoft Excel;
- познакомить учащихся с разнообразием деловой графики (схемы, чертежи, графики, диаграммы, ...);
- познакомить учащихся с возможностью программы Excel создавать графики, диаграммы на основе исходных и производных данных;
- формирование ключевых компетенций в области представления и обработки информации средствами вычислительной техники;
- осуществлять развитие аналитического мышления;
- воспитание аккуратности, добросовестности.

**Дополнительная подготовка:** Наличие Microsoft Excel.

### Структура урока.

- I. Оргмомент (1 мин).
- II. Мотивация учебной деятельности учащихся, сообщение темы, цели урока, его задач (10 мин).
- III. Изучение нового материала (15 мин).
- IV. Практическая часть (15 мин).
- V. Подведение итогов урока (3 мин).
- VI. Домашнее задание (1 мин).

### Содержание урока.

**I.** Оргмомент.

Отметить отсутствующих, проверить наличие учебных принадлежностей, записать дату проведения урока в тетрадях.

**II.** Мотивация учебной деятельности учащихся:

1. общеизвестно, что люди лучше, легче воспринимают информацию, представленную в графическом виде (графики, диаграммы, таблицы, схемы, ...), чем текстовую с большим количеством числовых данных.
2. программа Excel позволяет создавать графики, диаграммы на основе исходных и производных данных;

Сообщить тему урока.

1. Поставить перед учащимися цели и задачи урока.
2. познакомиться с возможностью табличного процессора отображать числовую информацию в графическом виде;
3. рассмотреть виды графики, используемые в Excel;
4. научиться выбирать для решения задачи оптимальный тип диаграммы;
5. ввести понятия: масштаб, оси, ряд, категории, легенда, ...;
6. научиться строить различные типы диаграмм с помощью Мастера построения диаграмм;

### III. Изучение нового материала.

Excel дает возможность отображать данные не только в виде таблицы, но и в виде **диаграммы**, которая отображает данные нагляднее, облегчает восприятие и помогает при анализе сравнения данных.

**Диаграмма** – это графическое представление данных рабочего листа.

Для наглядного представления данных необходимо подобрать **Тип и Вид** диаграммы.

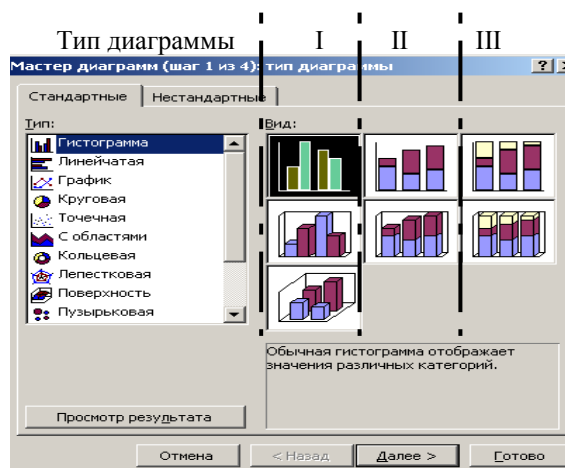
Excel предлагает следующие виды и типы диаграмм:

**Гистограмма** – столбчатая диаграмма.

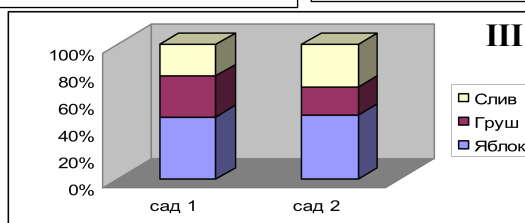
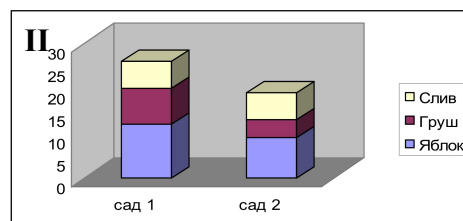
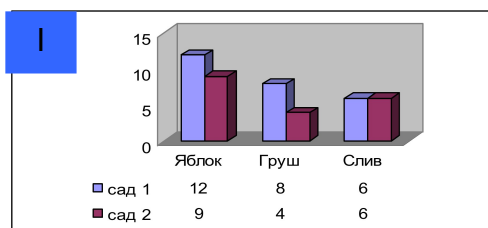
I столбец. Высота прямоугольника характеризует величину показателя. Диаграмма позволяет сравнить показатели величин друг с другом.

II столбец. Показывает численную долю отдельного показателя в суммарном объеме. Позволяет проследить динамику суммарного объема.

III столбец. Позволяет сравнивать процентное соотношение долей каждого показателя в общем объеме.



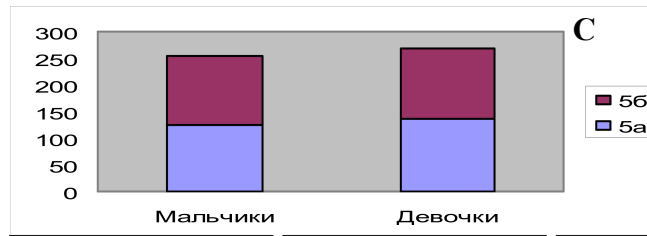
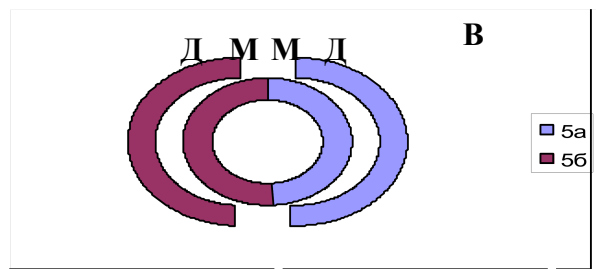
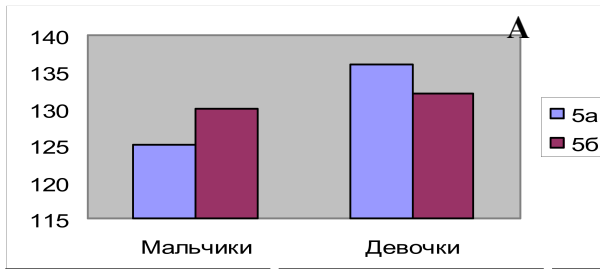
**Пример: Наш сад.** Запишите вид каждой из диаграмм и расскажите (устно) какой вид диаграммы позволяет а) сравнить количество



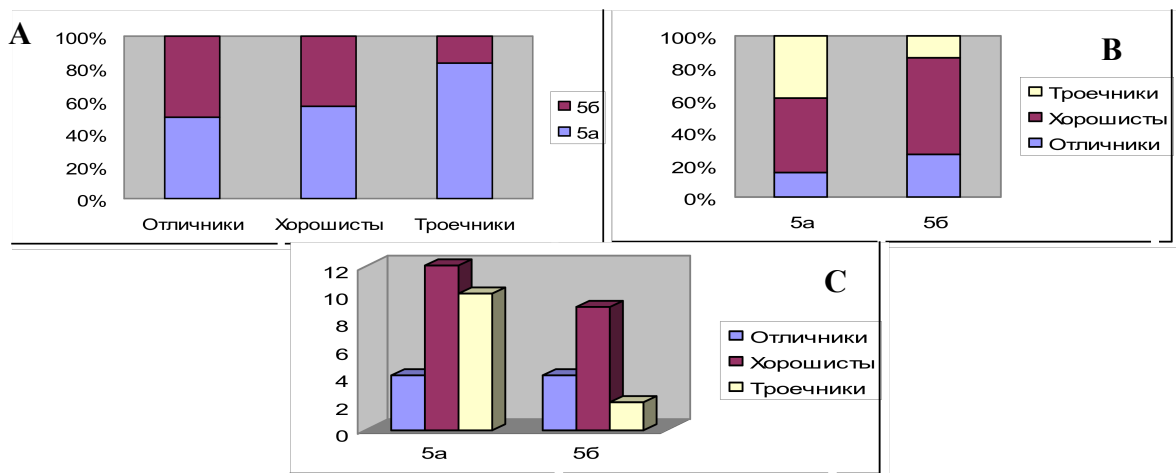
### Вопросы.

Определите, какой тип диаграммы наилучшим образом наглядно представляет данные:

1. Сравнить средний рост мальчиков и девочек в 5а и 5б классах.



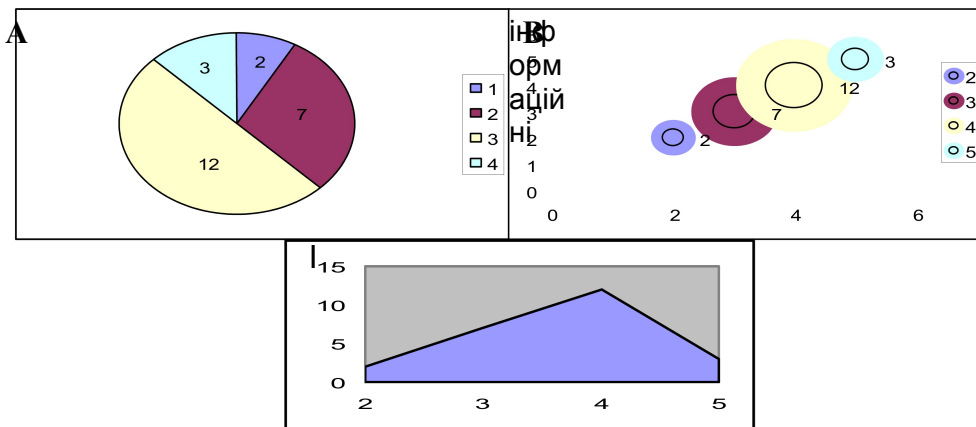
2. Сравнить процентное соотношение «Отличников», «хорошистов» и «троечников» в 5а и 5б классе относительно всех учащихся в этом классе.



3. Проследить динамику сдачи макулатуры по полугодиям, отобразить доли классов в общей сумме сданной макулатуры.



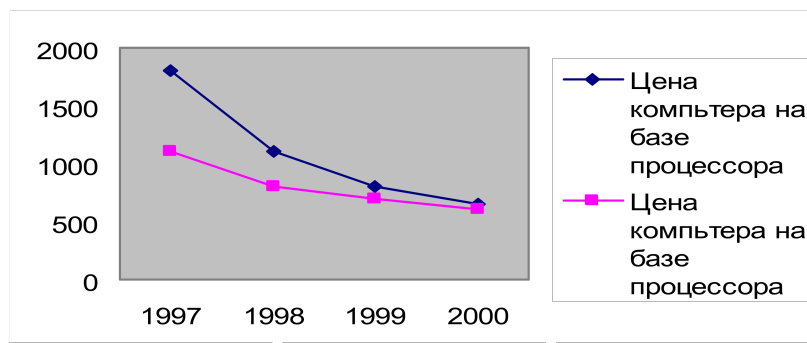
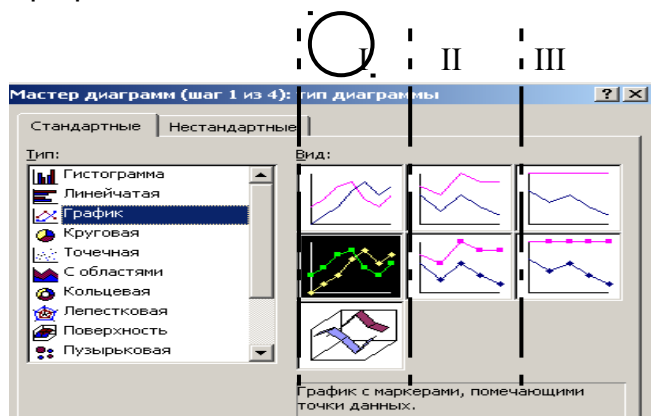
4. Какой тип диаграммы наиболее наглядно отобразит результаты контрольной работы в классе?



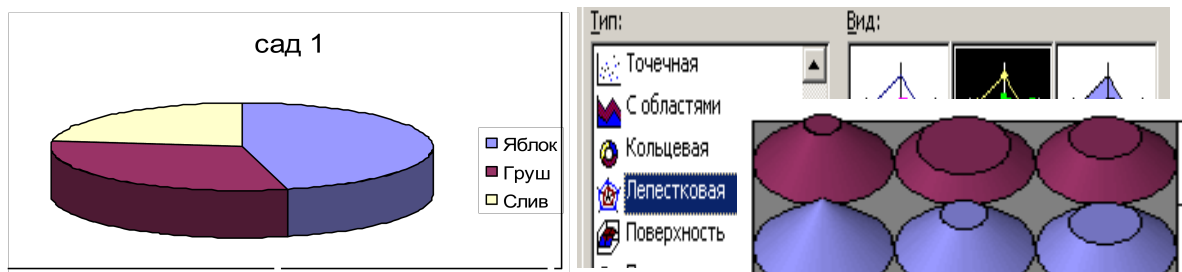
Графики

График показывает зависимость одной величины от другой. Часто используется для наглядного представления изменения параметра с течением времени. Аналогично диаграммам Excel может предложить построить три вида графиков. Чаще используются графики в первом столбце.

Пример: Цена компьютера на базе процессора Pentium и Atlon.



Круговая диаграмма строится по одному ряду данных. Показывает долю каждого параметра в целом.




Лепестковая диаграмма отображает распределение значений относительно начала координат.

Коническая, цилиндрическая, пирамидальная, нестандартные диаграммы помогут разнообразить вид представления данных.

IV. Практическая часть. Закрепление:

1. открыть файл с именем «Реализация книг».
2. Раздать ПАМЯТКУ по построению диаграмм «Создание диаграммы» ПАМЯТКА.

Для создания диаграммы используется **Мастер диаграмм**. Его можно запустить кнопкой . Диаграмма при этом создается в диалоговом режиме.

**Кнопка** Мастера диаграмм находится на панели инструментов Стандартная или в Меню **Вставка**. После выбора типа диаграммы, Далее, выбираем диапазон данных, **оформляем подписи** во вкладке **Ряд**. Далее можно уточнить **Параметры диаграммы**: заголовки, легенду, оси, линии сетки, таблицу данных. Далее мастер диаграмм предложит расположение диаграммы на **отдельном листе или имеющемся**. Изменить цвет элементов диаграммы можно при помощи контекстного меню на объекте.

**Выбор данных** можно произвести до запуска мастера диаграмм. Excel различает в диапазоне **текстовые** и **числовые** данные, поэтому если в диапазоне выделить подписи к рядам и столбцам, то программа автоматически предложит **легенду** и **подписи категорий** (надписи по оси Ох).

Если ряд аргументов представлен **числовыми данными**, то при построении графика функции необходимо выделить значения функции, выбрать тип графика, далее во вкладке **Ряд** в подписях по оси Ох выделить диапазон аргументов.

3. построить 1-2 типа диаграмм по столбцу реализация книг с полным оформлением (подписи к осям, ряды данных, легенда, ...);

4. сохранить результаты работы под именем «Реализация книг Иванова (Петрова, Сидорова)».

V. Подведение итога урока.

VI. Домашнее задание:

Выучить по конспекту. Уметь строить диаграммы.

## Урок 7

**Тема урока:** Практическая работа «Деловая графика. Построение диаграмм и графиков на основе табличных данных».

**Тип урока:** Урок комплексного применения ЗУН учащихся.

**Цель урока:**

- обучающие:
  - повторение и закрепление знаний учащихся правил построения диаграмм в таблицах Excel;
  - повторение алгоритма построения диаграмм и графиков;
- развивающие:
  - формирование ключевых компетенций в области представления и обработки информации средствами вычислительной техники;
  - развивать умения планировать свою деятельность;
- воспитывающие:
  - воспитание аккуратности, добросовестности.
  - воспитание умения самостоятельно мыслить, ответственности за выполняемую работу, аккуратности при выполнении работы.

**Дополнительная подготовка:** Наличие Microsoft Excel.

### Структура урока.

I. Оргмомент (1 мин).

II. Актуализация ЗУН, необходимых для творческого применения знаний (10 мин).

III. Применение обобщенных ЗУН. Практическая работа (25 мин).

IV. Контроль (самоконтроль) знаний, умений, навыков (5 мин).



V. Подведение итогов урока (3 мин).

VI. Домашнее задание (1мин).

### Содержание урока.

#### I. Оргмомент.

Отметить отсутствующих, проверить наличие учебных принадлежностей, записать дату проведения урока в тетрадях.

Как вы знаете, в нашей жизни часто встречаются различные задачи анализа данных и представления результатов в виде диаграмм и графиков. Наша цель: на уроке отработать умения построения различных типов диаграмм с учетом характера исходных данных.

#### II. Актуализация ЗУН, необходимых для творческого применения знаний:

- вспомнить: что называется диаграммой в Excel? (Диаграмма – графическое представление данных рабочего листа.);
- какие типы диаграмм можно строить в программе Excel?
- напомнить алгоритм работы с диалоговым окном построения диаграммы «Мастером диаграмм»;
- вспомнить основные понятия: ряд, легенда, оси, ...

#### III. Применение обобщенных ЗУН. Практическая работа:

1. Продолжить работу с документом Государства Европы (Африки, Америки, ...) Иванова (Петрова, Сидорова, ...). Открыть его.
2. Следуя подсказкам «Мастера диаграмм» построить две различные диаграммы (тип выбрать самим, исходя из анализа исходных данных).
3. За исходные данные принять данные столбцов «Площадь государств» и «% по отношению к общей численности».
4. Диаграммы озаглавить, прописать оси, указать ряды данных, отобразить легенду.
5. Для отображения в диаграмме только необходимой информации воспользоваться Меню: «Формат»-«Столбец»-«Скрыть».
6. Готовую работу сдать учителю.

#### IV. Контроль (самоконтроль) знаний, умений, навыков:

Оценить работу учащихся. Выставить оценки.

#### V. Подведение итогов урока

- графический способ представления информации наглядно отображает её характер;
- в программе Excel имеется возможность графически отобразить информацию в различном виде;

#### VI. Домашнее задание:

Уметь выбирать типы диаграмм для отображения конкретной числовой информации. Уметь в диалоговом режиме с «Мастером диаграмм» строить диаграммы.

## Урок 8

**Тема урока:** Упорядочение и поиск нужных данных в электронной таблице. Фильтрация данных. Вывод табличных данных и диаграмм на печать.

**Тип урока:** Урок усвоения материала и первичного его закрепления.

#### Цели и задачи урока:

- продолжить формирование у учащихся навыков работы в «Excel»;
- познакомить учащихся со способами упорядочения данных (сортировки), фильтрации;
- научить осуществлять поиск данных, их сортировку;
- показать на примерах использование простого и расширенного

- фильтров;
- вырабатывать умение пользоваться учебником;
- закрепить знания, применяя их на практике
- воспитывать чувство ответственности за результаты работы.
- расширять кругозор, используя межпредметные связи (география).

**Дополнительная подготовка:** Проверить наличие на партах учебника «Информатика 10-11» авт. И.Т. Зарецкая и др.

#### **Структура урока.**

- I. Оргмомент (1 мин).
- II. Проверка домашнего задания. Актуализация опорных знаний. (5 мин).
- III. Мотивация учебной деятельности учащихся, сообщение темы, целей и задач урока (5 мин).
- IV. Изучение нового материала в процессе выполнения практического задания по учебнику (25 мин).
- V. Подведение итогов урока, выставление оценок (7 мин).
- VI. Домашнее задание (2 мин).

#### **Содержание урока.**

- I. Отметить отсутствующих, записать дату проведения урока в тетрадях.
- II. Устный опрос по вопросам в конце параграфов учебника.
- III. На прошлых уроках мы выяснили, что в Excel в табличном виде могут храниться достаточно большие объёмы информации. Для того, чтобы быстрее осуществить поиск, нужно данные в таблице сначала упорядочить.

Сегодня на уроке мы с вами познакомимся со способами упорядочения данных в таблице и научимся осуществлять поиск нужных данных, а также их сортировку и фильтрацию. Запишите в тетрадях тему урока.

**IV.** Откройте учебник на странице 417. Сейчас мы с вами рассмотрим ПРИМЕР 2: «Сведения о реках Украины», детально его разберем, а затем вы самостоятельно выполните задание, пользуясь учебником.

Прочитаем формулировку задачи и рассмотрим таблицу, приведенную в книге (рис46):

Учащиеся выполняют работу самостоятельно, пользуясь учебником, но могут обращаться с вопросами к учителю, который помогает, подсказывает, контролирует выполнение задания, разрешает вывести на печать построенные диаграммы.

**V.** Итак, сегодня мы: Узнали..., Научились..., Выполнили..., Оценки...

**VI.** Домашнее задание: глава 7 повторить, уметь отвечать на вопросы; примеры из §6-го разобрать, задача № 2 на стр.424.

### **Урок 9**

**Тема урока:** Практическая работа «Упорядочение и поиск нужных данных в ЭТ. Фильтрация данных. Вывод табличных данных и диаграмм на печать».

**Тип урока:** Урок комплексного применения учащимися ЗУН

#### **Цели и задачи урока:**

Выполнение учениками практической работы, в ходе которой они демонстрируют:

- навыки работы в среде табличного процессора «Excel»;
- умение находить нужные данные, сортировать и фильтровать их;
- строить диаграммы и выводить их на печать;
- применять полученные знания на практике.

#### **Структура урока.**

- I. Оргмомент (1 мин).

- II. Проверка д/з. Актуализация опорных знаний. (5 мин).
- III. Мотивация учебной деятельности учащихся, сообщение темы, целей и задач урока (5 мин).
- IV. Закрепление материала в процессе выполнения практического задания (25 мин).
- V. Подведение итогов урока, выставление оценок (7 мин).
- VI. Д/з (2 мин).

#### **Содержание урока.**

**I.** Отметить отсутствующих, записать дату проведения урока в тетрадях и журнале.

**II.** Устный опрос по вопросам в конце параграфов главы 7 учебника.

**III.** Сегодня на уроке вы выполняете практическую работу по поиску, сортировке и фильтрации данных в среде табличного процессора «Excel». Вам также необходимо продемонстрировать своё умение выбирать тип диаграммы, наиболее подходящий для решения вашей задачи, построить эту диаграмму, при необходимости вывести её на печать.

**IV.** *Ход практической работы:*

1. Каждому учащемуся открыть свой файл «Государства Европы».
2. Осуществить сортировку данных в таблице
  - а) По названию государств в алфавитном порядке;
  - б) По названию столиц в алфавитном порядке;
  - в) По площади;
  - г) По численности населения;
  - д) По плотности населения;
3. Сохранить результаты работы для проверки.
4. С помощью фильтров выберите государства с наибольшими (наименьшими):
  - а) Площадью;
  - б) Численностью населения;
  - в) Плотностью населения
5. Выбрать соответствующие типы диаграмм и построить их.
6. Сохранить результаты работы для проверки.
7. Показать работу учителю.
8. Вывести на печать указанные учителем диаграммы.

**V.** Итак сегодня нам удалось...Оценки...

**VI.** Домашнее задание: повторить всю главу 7, подготовиться к контрольной работе и тематической аттестации.

## **Урок 10**

**Тема урока:** Тематическая аттестация по теме «Электронные таблицы»

**Тип урока:** Урок контроля и оценивания ЗУН

**Цели и задачи урока:**

- Выполнение учениками контрольного тестирования, в ходе которого они демонстрируют теоретические знания по теме «Электронные таблицы»;
- Выполнение учениками практической работы, в ходе которой они демонстрируют:
  - навыки работы в среде табличного процессора «Excel»;
  - умение применять полученные знания на практике;
  - Комплексное применение ЗУН при решении задач.

**Структура урока.**

- I. Организация работ (2мин)
- II. Тестовые задания (теория) (20 мин)—выполняет I подгруппа (одна половина учащихся);
- III. Решение задачи (практика) (20 мин)—выполняет II подгруппа (вторая половина учащихся);
- IV. Группы меняются местами (2мин)
- V. Домашнее задание (1мин)

#### **Содержание урока.**

- I. Раздать тетради для контрольных работ и тестовые задания одной группе учащихся;
- II. I группа пишет контрольное тестирование;
- III. II группа выполняет практически домашнюю задачу (1, 2 или 3 на стр. 424 учебника);
- IV. Группы меняются местами;
- V. Домашнее задание: повторить весь материал по теме.

*Пример тестовых заданий:*

*Тестирование по теме Электронная таблица EXCEL*

1. Пересечение строки и столбца в таблице EXCEL называется
  - а) Таблица б) Диапазон в) Ячейка г) Поле
2. Адрес ячейки ( A1, C5 ...) указывает ...
  - а) Тип данных, записанных в ячейке
  - б) На пересечении какого столбца и строки она находится ячейка
  - в) Ничего, задается пользователем
3. Что нужно сделать, если данные не помещаются в видимой части ячейки, а сокращать их нельзя?
  - а) Уменьшить информацию, чтобы она помещалась по ширине
  - б) Увеличить ширину ячейки
  - в) Установить флажок «переносить по словам » для данной ячейки
  - г) Уменьшить масштаб выведения на экран
4. Как отображается изменение данных исходной таблицы Excel на соответствующей диаграмме?
  - а) Диаграмма не изменяется б) Диаграмма изменяется автоматически
  - в) По желанию пользователя г) Зависит от задачи
5. Что отображает строка формул?
  - а) что набирается в текущей ячейке или что в ней уже есть
  - б) Только математические формулы в) Имя текущей ячейки
  - г) Только текст
6. Укажите правильный способ изменения размера диаграмм
  - а) С помощью команды Формат б) С помощью команды Меню
  - в) Передвигая мышью точки выделения диаграммы
  - г) Размер изменять нельзя
7. Укажите неправильный вариант или варианты записи функции суммирования
  - а) СУММ(B2;B7) б) СУММ(B2;B7,B20)
  - в) СУММ(B2+B7) г) СУММ(B2),B7)
8. Укажите правильный вариант или варианты записи функции MIN
  - а) MIN(B1;M1;C1) б) MIN(B1:M1)
  - в) MIN(B1:C1;E1:M1) г) MIN(BI-M1)
9. Какой формат данных можно одновременно установить в ячейке?
  - а) Денежный и общий б) Числовой и дату в) Дату и общий
  - г) В ячейке может быть только один формат данных
10. Можно ли установить в разных ячейках одной строки разные форматы данных (например текстовый и денежный)
  - а) Да б) Нет в) Да, но в столбце должен быть тот же формат

11. Можно ли установить в разных ячейках одного столбца разные форматы данных (например текстовый и денежный)
  - а) Да б) Нет в) Да, но в строке должен быть тот же формат
12. Что означает запись адреса ячейки \$C2 в формуле?
  - а) Адрес ячейки не изменяется во время копирования формулы
  - б) Денежный формат ячейки
  - в) Во время копирования формулы не изменяется номер строки
  - г) Во время копирования формулы не изменяется имя столбца
13. Что означает запись адреса ячейки C\$2 в формуле?
  - а) Адрес ячейки не изменяется во время копирования формулы
  - б) Денежный формат ячейки
  - в) Во время копирования формулы не изменяется номер строки
  - г) Во время копирования формулы не изменяется имя столбца
14. Что отображается после ввода в ячейке формулы и нажатия Enter?
  - а) Имя ячейки б) Результат вычислений по этой формуле
  - в) Значок =, а потом результат вычислений г) Адрес ячейки
15. Что происходит с формулой при копировании ее в другую ячейку?
  - а) Изменяется структура формулы
  - б) Изменяются ссылки на ячейки, где нет знака \$
  - в) Изменений нет г) Копировать формулу нельзя
16. Как в Excel можно записать формулу суммирования диапазона ячеек от A3 до A15?
  - а) =СУММ(A3;A15) б) =СУММ(A3-A15)
  - с) =СУММ(A3,A15) г) =СУММ(A3:A15)
17. Можно ли создать разные таблицы в одной рабочей книге?
  - а) Да в) Да, если таблицы одинаковой тематики
  - б) Нет г) Да, если таблицы одинаковой структуры
18. Предложение, которое начинается со знака равенства и содержит операторы, ссылки на ячейки, функции, константы называется...
  - а) Функцией б) Формулой в) Программой г) Макросом

Список использованной литературы.

1. Информатика. Программы для общеобразовательных учебных заведений.— Запорожье: Премьер, 2003.—304с. на укр. яз.
2. Информатика: Учебн. пособие для 10-11 кл. средн. общеобразоват.-школ/ И.Т.Зарецкая, В.Г. Колодяжний. А.М. Гуржий, О.Ю.Соколов.— К.: Учебная книга, 2002.—496с.:ил.
3. Информатика: 7-11кл. Учеб. пособие — 2-е изд., доп.— К.: Издательство А.С.К., 2003.— 536с.:ил.

#### 4.5. Базы данных. Системы управления базами данных. (8 годин)

№ урока	Тема урока	
1	Понятие о базах данных и их видах: фактографические и документальные. Информационно-поисковые системы и системы управления базами данных (СУБД), их назначения и функции.	
2	Основные понятия базы данных. Типы данных. Проектирование базы данных и создания структуры базы данных.	
3	Практическая работа № 1. Проектирование базы данных и создание структуры базы данных.	
4	Практическая работа №2. Введение и редактирование данных. Работа с таблицами.	
5	Практическая работа №3. Поиск, сортировка и фильтрация данных.	
6	Практическая работа №4. Использование запросов.	
7	Практическая работа №5. Создание форм и отчетов.	
8	Тематическое оценивание из темы «Базы данных. Системы управления базами данных».	

#### Урок 1

**Тема урока:** Понятие о базах данных и их видах: фактографические и документальные. Информационно-поисковые системы и системы управления базами данных (СУБД), их назначения и функции

**Тип урока:** изучение и первичное закрепление знаний.

**Цели урока:**

- ✓ ознакомить учащихся с общими представлениями о базах данных, об информационно-поисковых системах, о системах управления базами данных, показать разнообразие способов организации данных, дать понятия о видах баз данных, показать значимость материала, его применение в практической деятельности;
- ✓ развивать активную познавательную деятельность и аналитическое мышление учеников, память;
- ✓ воспитывать у учеников позитивное отношение к обучению путем раскрытия практического значения темы, воспитывать информационную культуру.

**Оборудование:** схемы моделей баз данных, компьютерная презентация на данную тему, реляционная база данных «Гимназия»

#### Структура урока

I. Организационный момент	2 мин
II. Актуализация опорных знаний. Мотивация учебной деятельности	5 мин
III. Усвоение новых знаний	25 мин
IV. Первичная проверка усвоения материала	5 мин
V. Итоги урока	5 мин
VI. Домашнее задание	3 мин

#### Ход урока

**I.** Организационный момент.

**II.** Актуализация опорных знаний. Мотивация учебной деятельности.

До появления вычислительной техники вся информация сохранялась в письменном или печатном виде. Однако, чем больше были объемы информации, которыми человек должен был оперировать, тем острее

становился вопрос хранения информации и ее обработки. *Информация предназначена для обработки, называется данными.* Для автоматизации обработки используют компьютер. Данные организуют специальным образом – в виде баз данных.

Пользуетесь ли вы базами данных? Можно ли назвать записную книжку базой данных? Дневник? Газету? Компакт-диск? Поисковые службы Internet? Различные архивы? Бухгалтерские документы? Расписание движения поездов? Заказы покупателей в магазине? Наличие товара на складе?

*III. Усвоение новых знаний.*

**Базы данных** – это структурированная совокупность взаимосвязанных данных определенной предметной области, предназначенная для длительного сохранения и постоянного применения.

Приведите примеры, где имеют дело с такой организацией данных?

*Прогнозируемые ответы учеников.* Библиотека, архив, классный журнал, телефонный справочник и тому подобное.

За видами БД можно разделить на документальные и фактографические базы данных.

**Документальная БД** – база данных, запись которой отражает конкретный документ, вмещает его библиографическое описание и, возможно, другую информацию о нем. (Пример – каталог библиотеки, поисковые службы Internet).

**Фактографическая БД** – база данных, что вмещает информацию, которая непосредственно относится к предметной отрасли. Например, электронные выпуски газет, расписание движения транспорта, адресное бюро, номера телефонов, рецепты приготовления блюд.

Чтоб обеспечить скорость и качество поиска данных в базе, этот процесс должен быть автоматизирован. Для поиска информации используют информационно-поисковую систему.

Ее функции:

- сохранение больших объемов информации;
- быстрый поиск необходимой информации;
- добавление, изъятие и изменение информации;
- вывод информации в удобной для человека форме.

Различают автоматизированные, библиографические, диалоговые документальные и фактографические.

Можно ли Rambler назвать информационно-поисковой системой? Почему?

Какие еще информационно-поисковые системы вы знаете?

Компьютерную базу данных можно создать посредством специальных программных средств, которые имеют название **Системы Управления Базами Данных (СУБД)**.

**СУБД** – комплекс программных средств для создания БД, сохранения и поиска в них необходимой информации.

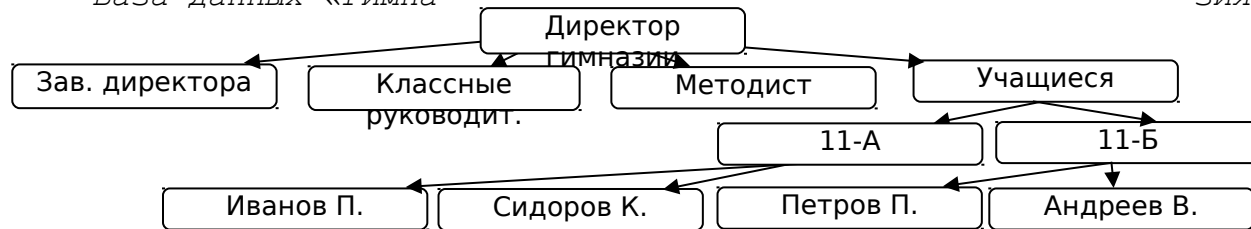
Современные СУБД должны удовлетворять следующие требования:

- возможность манипуляции данными (введение, выбор, вставка, удаление и т.п.);
- возможность поиска и формирования запросов;
- обеспечение целостности (согласованности) данных;
- обеспечение защиты и секретности;
- оформление документации.

За структурой организации информации в БД различают такие модели.

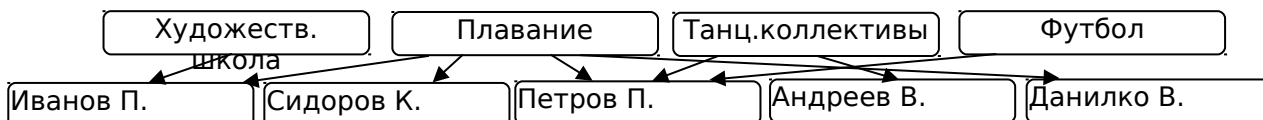
Иерархическая модель БД - представляет собой совокупность объектов разного уровня, причем объекты нижнего уровня подчинены объектам верхнего уровня.

База данных «Гимназия»



Сетевая модель БД - это совокупность объектов разного уровня, где каждый объект может быть связан с другим.

База данных «Кружок»



Реляционная модель БД - модель построена на взаимоотношениях между составляющими общей структуры базы данных. Представляет собой одну таблицу или совокупность взаимосвязанных таблиц.

Примеры реляционных СУБД: dBASE, FoxBase, FoxPro, Clipper, Access, MySQL.

Демонстрация реляционной базы «Гимназия»

**IV.** Первичная проверка усвоения материала..

Превратить к табличной форме модели баз данных Гимназия и Кружок.

**V.** Итоги урока

Обсуждение вопросов.

1. Что называют базой данных?
2. Перечислите основные свойства базы данных.
3. Что такое информационно-поисковая система? Какие ее функции?
4. Что такое СУБД? Каким требованиям она должна удовлетворять?
5. Какие модели БД вам знакомы?
6. Что такое реляционная БД? Приведите примеры?
7. Что такое иерархическая модель БД?
8. Чем отличается сетевая модель данных от иерархической?

**VI.** Домашнее задание

- Отвечать на вопросы по учебнику.
- Выучить определение: база данных, ИПС, СУБД.
- Знать модели баз данных, уметь изобразить их вид в блок-схеме.
- Придумать и нарисовать в тетради блок-схему конкретной сетевой и иерархической базы данных. Нарисовать придуманные блок - схемы на листе формата А4 (создание дидактического материала)
- Построить их реляционные модели.

## Урок 2

**Тема урока:** Основные понятия базы данных. Типы данных. Проектирование базы данных. Создание структуры базы данных

**Тип урока:** комбинированный (повторение, изучение и первичное закрепление знаний)

**Цели урока:**



- ✓ дать основные понятия базы данных: структура БД, таблица, запись, поле, ключевое поле;  
дать представление о типах данных: числовой, текстовый, дата, логический, OLE;  
научить учеников определять цель создания базы данных;  
показать важность проектирования базы данных;
- ✓ развивать логико-алгоритмическое мышление, умение сравнивать, устанавливать связи между понятиями.
- ✓ воспитывать у учащихся культуру работы с информацией, воспитывать навыки работы в группах, формировать понимание значения данной темы в будущей учёбе и профессиональной деятельности.

**Оборудование:** таблицы с фрагментами реляционной базы данных, инструкционные листы с заданиями, компьютерная презентация темы.

#### **Структура урока**

I. Организационный момент	2 мин
II. Проверка домашнего задания	5 мин
III. Актуализация опорных знаний. Мотивация учебной деятельности	5 мин
IV. Сообщение темы урока. Формулировка цели урока	3 мин
V. Усвоение новых знаний	10 мин
VI. Первичная проверка усвоения материала	10 мин
VII. Итоги урока	5 мин
VIII. Домашнее задание	5 мин

#### **Ход урока.**

- I.** Организационный момент.
- II.** Проверка д/з. Проверка выполнения задания в тетради.  
*Фронтальный опрос:*
- Что называют базами данных? Какие виды баз данных вы знаете?
  - Чем отличаются документальные БД от фактографических?
  - Что называют информационно-поисковыми системами? Приведите примеры ИПС.
  - Какие модели баз данных вы знаете?
  - Если информация организована в виде таблицы, к какой модели БД ее можно отнести?
  - Приведите примеры реляционных БД.
- III.** Мотивация учебной деятельности.  
Вы пришли в библиотеку. Вам необходимо быстро найти подходящую для Вас книгу. Сколько обычно занимает времени поиск и регистрация нужного материала. Как сделать более эффективной работу библиотекаря? (*Прогнозируемый ответ:* Создать базу данных.)  
А из чего нужно начинать создание базы данных?
- IV.** Сообщение темы урока. Формирование цели урока.
- V.** Усвоение новых знаний.  
Основным объектом реляционной базы данных является **таблица**.  
Создание базы данных.
- Проектирование БД (без компьютера)
  - Определение цели создания БД (обсуждение)
  - Определение таблиц, что входят в состав БД (обсуждение)

- Определение структуры таблиц, первичного ключа каждой таблицы. (обсуждение)
- Создание структур (за компьютером) На этом этапе посредством конкретной СУБД описывается структура таблиц, что входят к БД
- Введение записей.

**Строка таблицы** – это запись, что содержит информацию об отдельном объекте таблицы. Структура записей одинакова; совокупность элементов данных, из которых состоит запись, называется **полем**. Информация записи находится в полях. **Поле таблицы** – это колонка таблицы.

Поле 1	Поле 2	Поле 3	Поле 4	Поле 5

**Первичный ключ БД (ключевое поле)** – это поле или группа полей, посредством которых можно однозначно идентифицировать запись. Значение первичного ключа не должно повторяться в разных записях. С каждым полем связано еще одно свойство – **тип поля**. Тип определяет множество значений, которые может принимать поле в разных записях.

В реляционных базах данных используют типы полей:

- счётчик
- числовой
- дата
- логический.
- текстовый
- OLE
- Денежный
- MEMO и др.

**Значение полей** – это некоторые величины определенных типов. От типа величины зависят те действия, которые можно с ней проводить.

**Описать структуру БД** – это значит указать все поля таблицы и их характеристики.

Например:

Название поля	Тип
Номер	Счётчик
Код идентификационный	Числовой
Фамилия	Текстовый
Имя	Текстовый, Мастер подстановок
Отчество	Текстовый, Мастер подстановок
Год рождения	Дата
Адрес	Текстовый
Паспорт	Текстовый
Телефон	Текстовый
Медицинская страховка	Логический
Должность	Текстовый
Стаж	Числовой
Фото	OLE
Ставка	Денежный

Предоставлен фрагмент реляционной БД.

Номер поезда	Дата	Вагон	Номер места	Цена билета	Полка	Наличие билетов
87	23.10.07	плацкарт	32	200,50	Нижняя	да

Указать главный ключ, описать характеристики полей.

Решение вместе с учениками (ключ: номер поезда + дата)

Мы рассмотрели два примера. Скажите, которому из этапов создания баз данных отвечали наши действия?

Некоторые советы по проектированию БД

- определить структуру будущей базы данных, определить источник данных, продумать задачи, которые следует решить с помощью создаваемой БД;
- определить поля для каждой таблицы
- определив данные и задачи, разделить их на группы – они станут таблицами;
- выделить ключевые поля (общие для всех таблиц);
- определить условия выбора для запросов.

Тип данных определяет, какого вида данные допускается вводить в поле.

**VI.** Первичная проверка усвоения материала. Решение задач. Работа в группах. Спроектировать следующие базы данных.

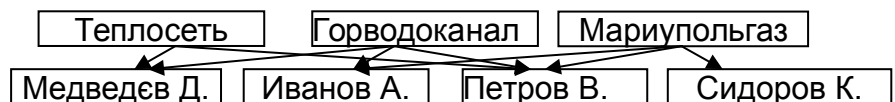
1. Превратить нижеприведенную информацию к табличному виду, определить имя таблицы и название каждого поля:  
*Иванова, Сидоров, Петрова, Романов, 10, 9, 8, физика, химия, биология, литература, 7*
2. Превратить нижеприведенную информацию к табличному виду, определить имя таблицы, название каждого поля и первичный ключ:  
*+20, Мариуполь, северный, Киев, дождь, дождь, южный +20 +15, Донецк, южный, без осадков, без осадков, Львов +17, восточный.*
3. Таблица базы данных «Товар» вмещает поля: код товара, наименование, цена, дата изготовления, срок годности, сертификат качества, фото. Определить тип каждого поля.

**VI.** Итоги урока. Обсуждение вопросов:

1. Из каких этапов состоит проектирование баз данных?
2. Зачем перед созданием БД нужно определять структуру таблиц?
3. Что такое запись? Поле?
4. Что такое первичный ключ? Какая его роль в таблице?
5. Что определяет тип поля?
6. Какие типы полей бывают?

**VII.** Домашнее задание.

- Выучить этапы создания базы данных.
- Сравнить БД, которые созданы в табличном процессоре с БД, созданными в СУБД.
- Придумать и описать структуру таблицы базы данных, которая вмещает четыре поля разных типов: числового, текстового, дата, логического.
- Данная структура БД «Жилищно-коммунальные услуги».



- Определить модель базы данных;
- Превратить данную структуру к табличному виду;
- Описать структуру созданной таблицы.

### Урок 3

**Тема урока:** Практическая работа № 1 «Проектирование базы данных и создания структуры базы данных»

**Тип урока:** закрепление знаний, изучение нового материала (комбинированный)

**Цели урока:**

- ✓ ознакомить учеников с интерфейсом СУБД Microsoft Access; отработать умения и навыки создания структуры базы данных в режиме Конструктора.

- ✓ развивать алгоритмическое и эвристическое мышление (прогнозирование ожидаемого результата) умение выделять главное;
- ✓ воспитывать самостоятельность, настойчивость в достижении цели.

**Оборудование:** электронная презентация для объяснения нового материала, инструкционные листы для практической работы, домашнего задания.

#### Структура урока

I. Организационный момент	2 мин
II. Проверка домашнего задания	5 мин
III. Мотивация учебной деятельности	3 мин
IV. Подготовка к основному этапу (мотивация учебной деятельности, объяснение нового материала)	7 мин
V. Закрепление знаний, формирование навыков	20 мин
VI. Итоги урока	5 мин
VII. Домашнее задание	3 мин

#### Ход урока

**I.** Организационный момент.

**II.** Проверка домашнего задания с целью коррекции.

Обсуждение вопросов:

- Этапы создания базы данных.
- Создание структуры базы данных.
- Что такое поле? Запись?
- Типы данных полей.
- Какой этап создания базы данных вы прошли выполняя домашнее задание?

**III.** Мотивация учебной деятельности.

Второй этап создания базы данных – создание структуры в конкретной СУБД. Для создания структуры БД на компьютере мы будем использовать СУБД Microsoft Access. Эта программа входит в офисный пакет фирмы Microsoft.

**IV.** Подготовка к основному этапу.

Сообщение темы урока, целей.

Основным объектом работы Microsoft Access являются **таблицы**. Сведения из каждого источника сохраняются в отдельной таблице. При работе с данными из нескольких таблиц устанавливаются связи между таблицами.

##### Создание таблиц.

Создание таблиц выполняется в окне Access во вкладке «Таблица» щелчком по кнопке Создать. При этом возможны следующие способы:

**Режим таблицы.** Выводит бланк (форму) абстрактной таблицы, который потом может принимать конкретные форму и содержание. В режиме таблицы добавляются, редактируются или просматриваются табличные данные. Также можно проверить орфографию и напечатать табличные данные, отфильтровать и отсортировать записи, изменить внешний вид таблицы или изменить структуру таблицы, добавив или удалив столбцы.

**Конструктор.** В режиме конструктора можно непосредственно указать параметры всех элементов структуры таблицы.

**Мастер таблиц.** Позволяет выбрать поля, включаемые в таблицу, из широкого списка образцов полей разных типов (деловые контакты, ведение домашнего хозяйства и т. д.).

**Импорт таблиц.** Импорт данных в таблицу из других файлов.

**Связь с таблицами.** Установление связей с данными, хранящимися в таблицах других БД (не обязательно Access), позволяющих изменять эти данные.

После создания таблицы на экран выводится ее окно и можно приступить к вводу и редактированию данных в полях.

Редактирование данных в поле и работа с фрагментами таблиц (копирование, перемещение, удаление) выполняются по общим правилам.

#### Создание таблиц в Режиме Конструктора

Создание таблиц в окне *Конструктора* предоставляет широкие возможности по определению параметров создаваемой таблицы. Для этого необходимо в окне базы данных нажать кнопку *Создать* и выбрать из списка возможных вариантов в окне диалога *Новая таблица режим Конструктор* и нажать кнопку *ОК*. Откроется окно *Конструктора таблицы*.

В верхней части окна диалога находится таблица, которая содержит следующие атрибуты создаваемой таблицы: *Наименование поля*, *Тип данных* и *Описание*. Кроме этих основных атрибутов, каждое поле таблицы обладает дополнительными свойствами, отображаемыми в нижней части *Конструктора*. Строятся таблицы в окне *Конструктора* по следующим действиям:

#### Наименование поля

Наименование поля вводится в столбце **Имя поля**. При задании наименований полей необходимо помнить правила:

1. Наименование поля может содержать до 64 символов.
2. Наименование поля может содержать буквы, цифры, пробелы и специальные символы, за исключением точки (.), восклицательного знака (!), надстрочного символа (^), прямых скобок ([]) и управляющих символов с кодами ASCII 0-31.
3. Наименование поля не может начинаться с пробела.

Два поля в одной таблице не могут иметь одинаковых наименований.

#### Типы данных

Наименование поля используется для ссылки на данные таблицы. Для определения типа хранимых данных используется тип данных. Тип данных поля вводится в поле столбца **Тип данных**. В MS Access допустимыми являются данные следующих типов:

При работе с базой данных Access допустимы следующие типы данных:

<b>Текстовый</b> – одна строка текста (до 256 символов).	<b>Поле MEMO</b> – текст, состоящий из нескольких строк, которые затем можно будет просмотреть при помощи полос прокрутки (до 65 535 символов).
<b>Числовой</b> – число любого типа (целое, вещественное и т.д.).	<b>Дата/время</b> – поле, содержащее дату или время.
<b>денежный</b> – поле, значение которого выражено в денежных единицах.	<b>Счетчик</b> – поле, которое вводится автоматически с вводом каждой записи.
<b>Логический</b> – содержит одно из значений TRUE (истина) или FALSE (ложно) и применяется в логических операциях.	<b>Поле объекта OLE</b> – содержит рисунки, звуковые файлы, таблицы Excel, т.д.





#### Определение связей между таблицами

В MS Access можно установить постоянные связи между таблицами, которые будут поддерживаться при создании форм, отчетов и запросов. Устанавливая связи между двумя таблицами, выбираются поля, которые содержат одну и ту же информацию. Чаще всего будут связи-

ваться первичный ключ одной таблицы с совпадающими полями другой таблицы.

Создание связей между таблицами в MS Access осуществляется в окне диалога Схема данных. для определения связей между таблицами необходимо выполнить следующие действия:

В большинстве случаев связывают ключевое поле (представленное в списке полей полужирным шрифтом) одной таблицы с соответствующим ему полем (часто имеющим то же имя), которое называют полем внешнего ключа во второй таблице. Связанные поля не обязательно должны иметь одинаковые имена, но они должны иметь одинаковые типы данных (из этого правила существует исключения) и иметь содержимое одного типа. Кроме того, связываемые поля числового типа должны иметь одинаковые значения свойства Размер поля.

Переключение между режимом конструктора и режимом таблицы.	Кнопка  на панели инструментов в режиме конструктора.
	Кнопка  панели инструментов в режиме таблицы.
Создание или изменение ключевого поля.	Кнопка  панели инструментов.
	Меню «Правка», команда «Ключевое поле»
Создание связей между таблицами	<ol style="list-style-type: none"> <li>1. Закрыть все открытые таблицы.</li> <li>2. Кнопка  на панели инструментов.</li> <li>3. Выбрать поле в одной таблице и переместить его с помощью мыши на соответствующее поле во второй таблице.</li> <li>4. Создать связь нажатием кнопки «Создать».</li> </ol>

Дополнительная информация.

**Маска ввода** – формат, состоящий из постоянных символов (таких как скобки, точки или дефисы) и специальных символов маски, указывающих, в какие позиции, в каком количестве и какого типа данные могут быть введены.

*Инструктаж по технике безопасности.*

*Инструктаж к выполнению практической работы*

**V.** Закрепление знаний, формирование навыков работы с программой.

Индивидуальная самостоятельная работа за компьютерами.

**Практическая работа № 1 «Создание базы данных «Автосалон»**

1. Создайте базу данных «Автосалон»

2. Создайте таблицу **Владельцы** в режиме **Конструктора**. В ней должны быть следующие поля:

Имя поля	Тип данных	Маска ввода	Индексированное поле?	Подстановка
Код идентификационный	Числовой	00000000	Да, совпадения не допускаются	–
Фамилия	Текстовый	–	–	–
Имя	Текстовый (Мастер подстановок)	–	–	Фиксированный набор значений
Отчество	Текстовый (Мастер подстановок)	–	–	Фиксированный набор значений
Адрес	Текстовый	–	–	–
Паспорт	Текстовый	>LL00	–	–
Телефон	Числовой	00\–00\–	–	–

		00		
--	--	----	--	--

3. Выберите поле *Ключевое* – **Код идентификационный**, сохраните таблицу под именем **Владельцы**

4. Создайте таблицу **Машины**.

Имя поля	Тип данных	Маска ввода	Индексированное поле?	Подстановка
Номер порядковый	Счётчик	-	Да (не допускаются совпадения)	-
Марка	Текстовый (Мастер подстановок)	-	-	Фиксированный набор значений
Номер машины	Текстовый	-	-	-
Год выпуска	Числовой	0000	-	-
Цена	Числовой		-	
Код идентификационный владельца	Текстовый	00000000	-	-
Фото авто	Поле объекта OLE	-	-	-
Код поломки	Числовой	000	-	-
Дата сдачи в ремонт	Дата/время	00.00.0000;0;.	-	-
Дата возврата	Дата/время	00.00.0000;0;_	-	-

5. Выберите *Ключевое* поле – **Номер порядковый**, сохраните таблицу под именем **Машины**.

6. Создайте таблицу **Персонал**.

Имя поля	Тип данных	Маска ввода	Индексированное поле?	Подстановка
Код сотрудника	Числовой	0000	Да (Совпадения не допускаются)	-т
Фамилия	Текстовый	-	-	-
Имя	Текстовый (Мастер подстановок)	-	-	Фиксированный набор значений
Отчество	Текстовый (Мастер подстановок)	-	-	Фиксированный набор значений
Год рождения	Числовой	00\00\00	-	-
Адрес	Текстовый	-	-	-
Телефон	Текстовый	00\00\00	-	-
Фото работника	Поле объекта OLE		-	
Должность	Текстовый	-	-	-
Номер бригады	Текстовый	0	-	-
Оклад	Числовой	-	-	-
Медицинская страховка	Логический	-	-	Флажок

7. Выберите *Ключевое* поле – **Код сотрудника**, сохраните таблицу под именем **Персонал**

8. Создайте таблицу **Поломки**.

Имя поля	Тип данных	Маска ввода	Индексированное поле?	Подстановка
Код поломки	Числовой	000	Да, совпадения не допускаются	-
Поломка	Текстовый	-	-	-
Номер бригады	Текстовый	0	-	-
Стоимость детали	Числовой	-	-	-
Стоимость работы	Числовой	-	-	-

9. Выберите *Ключевое поле* – **Код поломки**, сохраните таблицу под именем **Поломки**

**VI.** Итоги урока.

1. Какие этапы создания базы данных мы отработали на уроке?

*Прогнозируемые ответы:* проектирование, создание структуры посредством конкретной СУБД.

2. Какую именно СУБД мы использовали?

3. В каком режиме мы создавали таблицы?

*Прогнозируемые ответы:* создание базы данных в режиме конструктора.

**VII.** Домашнее задание.

- Повторить этапы создания базы данных.

- Выполнить задание в тетради:

Таблицы базы данных имеют следующие поля: *шифр товара, наименование товара, единица измерения, цена, количество проданного товара за день, дата продажи товара, фамилия продавца; фамилия продавца, оклад, стаж работы.* Создать структуру таблиц. Определить первичный ключ.

## Урок 4

**Тема урока:** Практическая работа «Введение и редактирование данных. Работа с таблицами. Связывание таблиц»

**Тип урока:** комбинированный

**Цели урока:**

- ✓ отработать умения и навыки создания структуры базы данных; продолжить знакомство учащихся с интерфейсом СУБД Access; ознакомить с видами связей, научить создавать связи между таблицами;
- ✓ развивать алгоритмическое мышление учащихся, умение сравнивать, анализировать;
- ✓ воспитывать умение оценить результаты своей работы, воспитывать умение работы в паре, информационную культуру.

**Оборудование:** программа СУБД Access, инструкционные листы для практической работы, материал для домашнего задания.

**Структура урока**

I. Организационный момент	2 мин
II. Проверка домашнего задания	5 мин
III. Подготовка к основному этапу (мотивация учебной деятельности, объяснение нового материала)	10 мин
IV. Закрепление знаний, формирование навыков	20 мин
V. Итоги урока	5 мин
VI. Домашнее задание	3 мин

**Ход урока**

**I.** Организационный момент.




II. Проверка домашнего задания.

III. Подготовка к основному этапу.

В MS Access можно установить постоянные связи между таблицами, которые поддерживаются при создании форм, отчетов и запросов.

Устанавливая связи между двумя таблицами, выбираются поля, которые содержат одну и ту же информацию. Чаще всего будут связываться первичный ключ одной таблицы с совпадающими полями другой.

Создание связей между таблицами в MS Access осуществляется в окне диалога Схема данных. для определения связей между таблицами необходимо выполнить следующие действия:

1. Закрывать все открытые таблицы.
2. Кнопка  на панели инструментов.
3. Выбрать поле в одной таблице и переместить его с помощью мыши на соответствующее поле во второй таблице.
4. Создать связь нажатием кнопки «Создать»

#### Типы связей

##### Связь «Один к одному»

Пример 1. В учебном заведении только один преподаватель может быть директором, причём, директором он может быть, только в одном учебном заведении. Такая связь называется «Один к одному»



Пример 2. Связь «Один к одному».

Таблица «Сотрудники»

Номер	ФИО	Должность
1	Иванов В.С.	Инженер
2	Петров В.С.	Бухгалтер
3	СидоровС.Т.	Сторож

Таблица «Информация о сотрудниках»

Номер	Год род	Кол-во
1	1960	2
2	1958	3
3	1970	1

Первичный ключ установлен по полю – Номер.

##### Связь «Один ко многим»

Пример 1. В одной гимназии может работать несколько преподавателей. Связь «Один ко многим»



Пример 2. Связь «Один ко многим»

Таблица «Товар»

Наименование	Един измер	Цена
Сахар	кг	7
Макаронны	пачка	6
Куры	кг	20

Таблица «Отпуск товаров»

Товар	Дата	Кол-во
Сахар	10.12.06	100
Сахар	02.02.06	100
Сахар	03.05.07	200
Макаронны	01.05.08	250
Макаронны	01.02.08	100

Первичный ключ установлен по полю Наименование в таблице Товар.

##### Связь «Многие ко многим».

Пример 1. Рассмотрим связь "преподаватель читает предмет". Один преподаватель может читать несколько предметов и один предмет могут читать несколько преподавателей. Связь «Многие ко многим».



Таблица «Учебные группы и дисциплины»    Таблица «Преподаватели»

Группа	Предмет	Номер преп		Номер преп.	ФИО
ПС-1	Высшая мат.	10	←	10	Краснов В.О.
ТИ-1	Программиров.	10	←	12	Володин К.С.
РТ-2	Программиров.	12	←	62	Булгаков П.О.
ПС-1	Мат. анализ	12	←	30	Миронова С.В.

Связь осуществляется по полю *Номер преп* таблицы «Учебные группы и дисциплины» и таблицы «Преподаватели»

*Инструктаж по технике безопасности.*

*Инструктаж к выполнению практической работы.*

#### IV. Закрепление знаний, формирование навыков

#### Практическая работа «Ввод и редактирование данных. Связывание таблиц»

1. Откройте базу данных **Автосалон**, поочередно заполните данными все таблицы, учитывая, что записи полей **Код идентификационный** (Таблица **Владельцы**), **Номер порядковый** (Таблица **Машины**), **Код сотрудника** (Таблица **Персонал**), **Код поломки** (Таблица **Поломки**) не должны иметь совпадения.

Предлагаемые данные (следующие практические работы предполагают наличие именно этих данных!):

Таблица «Владельцы»

Код идент	Фамилия	Имя	Отчество	Адрес	Паспорт	Телефон
10000000	Иванов	Иван	Иванович	ул. Зелинского 108/6	ВВ45	21-32-65
20000000	Сидоров	Сергей	Петрович	ул. Зелинского 108/9	ВВ56	21-36-45
30000000	Фёдоров	Андрей	Иванович	пр. Ленина 12/45	КК25	41-44-77
40000000	Москалёв	Фёдор	Иванович	пр. Ленина 12/45	РР26	41-45-45
50000000	Трофимов	Андрей	Иванович	пр. Ленина 60/78	ВВ45	24-45-12

Таблица «Машины»

порядковый номер	Марка	Год выпуска	Цена	Код идент владельца	Фото авто	Код поломки	Номер машины	Дата сдачи	Дата возврата
1	Москвич	1945	70000	10000000		100	123135	12.02.2006	20.03.2006
2	Spider	1937	180000	20000000		100	124561	10.07.2006	20.08.2006
3	Omega	1940	150000	40000000		200	321231	13.01.2006	20.03.2006
4	SLK230	1960	200000	40000000		200	102123	10.09.2007	20.09.2007
5	Ferrari	1970	1230000	20000000		200	112360	10.02.2007	20.09.2007
6	BMW	1975	130000	60000000		400	113654	06.02.2006	20.03.2006
7	BMW	2000	000000	30000000		300	545343	06.02.2006	20.03.2006
8	BMW	2001	122330	10000000		400	233244	06.02.2006	20.03.2006

Таблица «Персонал»

Код сотрудника	Фамилия	Имя	Отчество	Год рождения	Адрес	Должность	бригадыНомер	Оклад (грн)	страховкаМедицинская
0001	Конюхов	Васи- лий	Фёдорович	1963	ул. Ватути- на, 45/12	меха- ник	1	2000	Да
0002	Болихин	Васи- лий	Иванович	1946	пр. Строи- телей, 6/12	ма- стер	1	2500	Да
0003	Вноеев	Иван	Федорович	1985	пр. Строи- телей 12/45	меха- ник	1	3000	Да
0004	Зиновьев	Андрей	Андреевич	1980	пр. Победы 56/89	сто- рож	2	800	Да
0005	Крестник	Андрей	Сидорович	1956	пр. Строи- телей 98/45	мене- джер	2	2000	Нет

Таблица «Поломки»

Код по- ломки	Поломки	Номер бригады	Стоимость детали	Стоимость ра- боты
100	кузов	1	200	100
200	крыло	1	300	200
300	дверь	2	300	124
400	покраска	2	600	400
500	заднее окно	3	123	100

2. Закройте все таблицы с данными. Создайте связи между таблицами.

**Владельцы – Машины** (Код идентификационный владельца) связь «один ко многим» (связь с обеспечением целостности данных)

**Поломки – Машины** (Код поломки) связь «один ко многим» (связь с обеспечением целостности данных)

**Поломки - персонал** (Номер бригады) связь «многие ко многим»

**V. Итоги работы.**

- Какие этапы создания базы данных вы отработали на двух практических работах?
- Прогнозируемые ответы: проектирование, создание структуры посредством конкретной СУБД, введение записей.
- Какую именно СУБД мы использовали?

**VI. Домашнее задание.**

1. Создать свою базу данных в режиме Конструктора (можно использовать структуру таблиц прошлого домашнего задания и структуры таблиц для контрольной работы (см. **Урок № 8**)
2. Создайте структуры баз данных в тетради:

- БД учащихся Вашего класса с использованием отношения «один к одному»
- БД выдачи книг в библиотеке с использованием отношения «один ко многим»
- БД планирования спортивных мероприятий на неделю с использованием отношений «многие ко многим»

## Урок 5

**Тема урока:** Практическая работа «Поиск, сортировка и фильтрация данных»

**Тип урока:** комбинированный

**Цели урока:**

- ✓ познакомить учащихся с возможностями обработки информации в базе данных: поиском, фильтрацией, сортировкой;
- ✓ формировать навыки выполнения поиска данных, сортировки, фильтров;
- ✓ развивать умение анализировать информацию с БД для формирования фильтра, развивать логическое мышление учеников, умение сравнивать, выделять главное;
- ✓ развивать навыки самоанализа собственной деятельности, смекалку, интерес к предмету.

**Оборудование:** электронная презентация для объяснения нового материала, база данных «Автосалон», инструкционные листы для выполнения практической работы.

### Структура урока

I. Организационный момент	2 мин
II. Проверка домашнего задания	5 мин
III. Подготовка к основному этапу (мотивация учебной деятельности, объяснение нового материала)	10 мин
IV. Закрепление знаний, формирование навыков	20 мин
V. Итоги урока	5 мин
VI. Домашнее задание	3 мин

### Ход урока

**I.** Организационный момент.

**II.** Проверка домашнего задания.

Обсуждение вопросов:

1. Что такое база данных? Цель создания баз данных?
2. Какие этапы создания базы данных?
3. Что является основным элементом баз данных?
4. Как можно пересмотреть содержание таблицы?
5. Можем ли пересмотреть выборочные поля?

**III.** Подготовка к основному этапу. Объяснение материала:

- демонстрация презентации по теме;

- демонстрация выполнения фильтрации, сортировки в БД «Автосалон»

**Фильтры** используются для вывода данных, которые соответствуют некоторым условиям.

Фильтры бывают: **Фильтр по выделенному**, **Фильтр для** (условие вводится с клавиатуры), **Расширенный фильтр**

### Сортировка и фильтрация данных.

Действие	Способ получения.
Сортировка данных в поле.	Контекстное меню \ команда «Сортировка по..»
Наложение фильтра на записи согласно выделенному фрагменту.	Контекстное меню \ команда «Фильтр по выделенному»

Изменение критериев фильтрации.	Контекстное меню \ команда «Изменить фильтр»
Применение фильтра.	Контекстное меню \ команда «Применить фильтр»
Удаление фильтра (восстановление содержания таблицы)	Контекстное меню \ команда «Удалить фильтр»








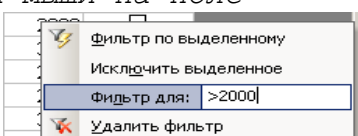
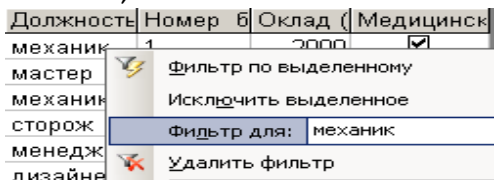


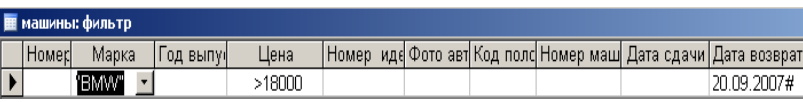
*Инструктаж по технике безопасности.*

*Инструктаж к выполнению практической работы.*

#### IV. Закрепление знаний и формирование навыков

##### Практическая работа «Поиск. Сортировка. Фильтрация данных»

1. Откройте базу данных **Автосалон**.
2. Выполните задания. В тетрадь записать основные команды выполнения поиска, сортировки, фильтров.

№	Задание	Алгоритм выполнения
1.	Найти в таблице Владельцы фамилию _____	Поставьте курсор а поле Фамилия, выполните  Правка, Найти или нажать кнопку  , введите фамилию, например, фамилию Дубров. Проследите куда переместился курсор.
2.	Выполнить поиск владельцев авто, фамилии которых начинаются с буквы «С» заканчиваются буквой «в»	Правка, Найти или нажать кнопку  Ввести С*в. Введите свои параметры поиска.
3.	Выполнить <b>фильтр по выделенному</b> : вывести в таблице «Владельцы» только фамилии Сидоров	Выделить фамилию Сидоров, нажать кнопку  , затем удалить фильтр, нажав кнопку 
4.	Сортировать все записи таблицы «Владельцы» по полю Фамилия по возрастанию	Выделить поле Фамилия, нажать кнопку 
5.	Сортировать все записи таблицы «Персонал» по полю Фамилия по убыванию	Выделить поле Фамилия, нажать кнопку 
6.	Вывести записи таблицы «Персонал», сотрудников, у которых оклад больше 2000грн	<i>Нажать правой кнопкой мыши на поле</i>  <i>«Оклад», ввести:</i>
7.	Вывести записи таблицы «Персонал», сотрудников, у которых оклад больше 2000грн и имеющих должность механика	<i>Нажать правой кнопкой мыши на поле «Должность», ввести «Механик»</i> 
8.	Удалить фильтр	<i>Правой кнопкой мыши, выбрать</i>  Удалить фильтр
9.	В таблице «Машины» Вывести машины BMW или Ferrari, которые стоят более 18000, дата возврата 20.09.2007	<i>Нажать на кнопку изменить фильтр</i>  , ввести 

10. В таблице «Персонал» вывести записи: год рождения сотрудников >1970, должность – механик, оклад – >1000	Выполнить Записи, Фильтр, Расширенный фильтр:			
	Поле:	Год рождения	Должность	Оклад (грн)
	Сортировка:			
	Условие отбора: или:	>1970	"механик"	>1000

V. Итоги урока.

Оценивание результатов работы учащихся.

Вопросы:

- Какие задачи ставили перед собой в начале урока? Достигли ли их?
- С какими понятиями мы познакомились сегодня на уроке?
- Для чего используют фильтры?
- Изменяют ли рассмотренные фильтры содержание таблицы?

VI. Домашнее задание. Повторить основные понятия базы данных. Выполнить различные фильтры по созданной Вами базе данных (домашнее задание прошлого урока)

## Урок 6

**Тема урока:** Практическая работа «Использование запросов»

**Тип урока:** закрепление знаний, изучение нового материала, формирование практических навыков (комбинированный)

**Цели урока:**

- ✓ ознакомить учащихся с новым объектом базы данных – запросами; научить учащихся использовать команды СУБД для добавления, удаления, выборочного пересмотра данных своей БД; показать актуальность создания разных видов запросов в работе с данными базы данных, научить создавать запросы «на выборку»; научить формировать критерии для запросов со сложными условиями для получения необходимой информации;
- ✓ развивать мыслительные способности: сравнение, обобщение, анализ, синтез.
- ✓ воспитывать навыки самостоятельной работы, настойчивость, эмоциональную направленность на практическую деятельность

**Оборудование:** электронная презентация для объяснения нового материала, база данных «Автосалон», инструкционные листы для выполнения практической работы.

### Структура урока

I. Организационный момент	2 мин
II. Проверка домашнего задания	5 мин
III. Подготовка к основному этапу (мотивация учебной деятельности, объяснение нового материала)	10 мин
IV. Закрепление знаний, формирование навыков	20 мин
V. Итоги урока	5 мин
VI. Домашнее задание	3 мин

### Ход урока

I. Организационный момент.

II. Проверка домашнего задания.

Фронтальный опрос.

- Для каких целей используют поиск, фильтрацию, сортировку данных?
- Яки образом можно упорядочить записи в таблице?
- Алгоритм выполнения расширенного фильтра.

III. Подготовка к основному этапу.

Демонстрация компьютерной презентации по теме, работа с базой данных «Гимназия», «Автосалон»

Под **Запросом** понимается операция отбора данных из таблицы, осуществление вычислений, выполнение групповых операций, объединение полей из нескольких таблиц или запросов.

#### Простой запрос

Открыть вкладку Запросы в окне базы данных, нажать кнопку Создать.

В диалоговом окне Новый запрос выбрать Простой запрос и нажать кнопку ОК. Первый шаг *Мастера*, позволяет выбрать таблицы, к которым будет адресован Запрос, и нужные поля из этих таблиц. Нажатие на кнопку > отправляет выбранное поле в новую таблицу-запрос. Нажатие на кнопку >> отправляет все поля в таблицу-запрос. Аналогично кнопки < и << отправляют поля из таблицы запроса в исходные таблицы. На втором шаге *Мастера* имеется возможность определить вид отчета: подробный или итоговый.

После выбора варианта «подробный» нажать кнопку Далее. На следующем шаге можно изменить имя нового Запроса и нажать кнопку Готово.

#### Конструктор запросов

Перейти на вкладку Запросы и нажать кнопку Создать. В диалоговом окне Новый запрос выбрать *Конструктор* и нажать ОК. Откроется окно диалога Добавление таблицы, в котором выбрать нужные таблицы для построения Запроса и нажать кнопку Добавить. После выбора нужных таблиц закрыть окно диалога. Окно, которое появится вслед за этим, это основное средство работы с запросами. Оно позволяет сформировать новый Запрос, а также редактировать Запросы. В верхней половине окна показаны выбранные таблицы и связи между ними. Можно редактировать эту схему, добавляя в нее новые объекты: таблицы или запросы, соответствующими командами из контекстного меню.

Нижняя часть окна - *бланк запроса* - содержит описание Запроса в табличной форме. Каждая колонка в нем отвечает одному полю. Строки Поле и Имя Таблицы заполняются двойным щелчком по выбранному полю в верхней части окна. Таблица расширяется вправо автоматически, когда пользователь добавляет в Запрос новые поля. Контекстное меню, связанное с бланком запроса, позволяет скрывать или, наоборот, показывать строки Имя таблицы и Групповая операция.

Строка Условие отбора позволяет вводить критерии для выборки данных из базы данных. Эти критерии аналогичны критериям используемых в Построителе выражений.

Строка Групповая операция содержит список функций и во многом определяет вид данных Запроса. Под *групповой операцией* понимается обработка и обобщение значений данного поля при помощи определенной функции. В результате в таблице Запроса меняется расположение или количество записей. Групповая операция позволяет легко создать вычисляемое поле. Допустим, необходимо подсчитать в Запросе, сколько раз встречается каждое значение некоторого поля. Для этого надо создать в бланке еще одну колонку для этого поля и определить для нее групповую операцию Count (подсчет). При применении групповой операции программа автоматически создаст новые имена полей, комбинируя названия функции для группировки и имена исходных полей Запроса.

Строка Сортировка на бланке Запроса дает возможность отсортировать полученные в результате выполнения Запроса данные.

Еще один способ «конструирования» Запроса - изменение его формулировки на языке SQL. Когда пользователь конструирует Запрос,

программа переводит последний на язык SQL. Текст Запроса можно увидеть, если выбрать команду Режим SQL в списке Вид или ту же команду из меню Вид. Этот тек можно отредактировать и, следовательно, изменить Запрос.

### Построитель выражений

При построении Запросов используются сложные условия отбора записей из таблиц. Для того чтобы составить его, используют Построитель выражений. Он запускается через контекстное меню, связанное со строкой Условие отбора, по команде Построить.

При составлении выражения надо помнить несколько простых правил:

1. Выражение всегда имеет логический тип. То есть в любом случае его значение должно быть Да или Нет.

2. Существуют определенные требования к синтаксису выражения. Например, имена объектов и полей заключаются в квадратные скобки, символьные строки - в кавычки, даты - в знаки #. имя объекта базы данных отделяется от имени поля восклицательным знаком.

Окно построителя имеет четыре области со своими полосами прокрутки. В верхней области располагается создаваемое выражение. Три нижние используются для выбора элементов. Они заполняются по иерархическому принципу. Левая область содержит список всех источников данных для Запроса. Ими могут быть объекты базы данных Access: таблицы, запросы, формы и т. д. Для облегчения поиска нужной информации список представлен в таком же виде, как и дерево каталогов в Windows. Кроме этого, в левый список включены папки, содержащие функции, константы и операторы, которые могут быть использованы в выражении.

Кнопки с изображенными на них знаками математических операций позволяют быстро вводить в выражение соответствующие символы.

Кроме традиционных знаков математических действий существует еще несколько операторов:

**Between And** - Заменяет знаки «больше или равно» и «меньше или равно». Например, условие Between 1981 And 1994 эквивалентно условию 
$$\geq 1981$$
 And 
$$\leq 1994$$
.

Оператор **In** позволяет использовать *списки значений*. Например, выражение In («Петров»; «Сидоров») позволяет отобрать записи с указанными фамилиями.

Выражение **Not Null (Is Not Null)** используется для отбора *непустых* записей в указанном поле.

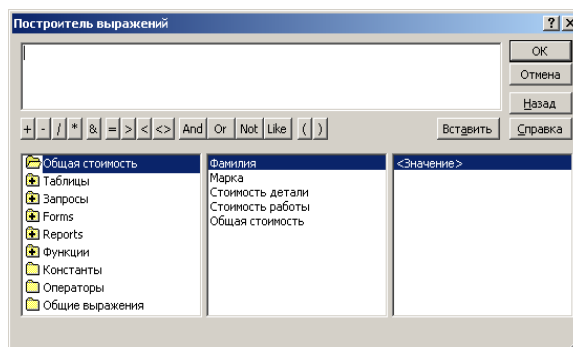
Выражение **Is Null** используется для отбора *пустых* записей в указанном поле.

$\backslash$  - Обратная косая черта обозначает деление целых частей делимого и делителя. Результат округляется до целого.

$\uparrow$  - Обозначает операцию возведения в степень.

Мод - Операция получения остатка от деления целых частей аргументов.

Знак **&** используется для «сложения» данных символьного типа. Для соединения строк можно использовать и более привычный знак «плюс». Например, следующие три выражения эквивалентны: "Петров"





"Петр" & "ов" "Петр" + "ов"

Оператор **Like** используется для создания масок при определении строк с неизвестными символами и требует дополнительных специальных символов и правил синтаксиса:

? - Знак вопроса обозначает любой одиночный символ. Например, если неизвестно, как написать: Carl или Karl, то условие можно задать как Like "?arl!",

\* - Звездочка обозначает любую последовательность символов.

# - Обозначает любую неизвестную цифру.

[] - Квадратные скобки подразумевают, что перед выполнением запроса должно быть активизировано окно диалога с приглашением ввести параметр, являющийся критерием отбора. Выражение в скобках выводится в этом окне в качестве сообщения-подсказки.

Логические операторы > (больше), >= (не менее), < (меньше), <= (не более) и операторы сравнения Or, And также используются для построения условий отбора. Они могут применяться к двум или нескольким выражениям и используются со скобками для изменения порядка выполнения выражения. Кроме них Access используется оператор логического отрицания Not. Условие Not «Иванов» позволяет выбрать все строки, которые не содержат указанный литерал в соответствующем поле. Результат выполнения логической операции Imp (импликация) зависит от порядка следования операторов.

Операторы «меньше» и «больше» могут использоваться не только с числовыми данными, но и с датами или текстом. Применительно к текстовым строкам они позволяют отбирать записи, предшествующие указанной строке или следующие за ней. В этих случаях учитывается алфавитный порядок значений поля. Например, выражение [Город]>«Москва» подразумевает, что в поле Город надо отобразить все названия, которые стоят в алфавитном списке после Москвы.

#### Перекрестный запрос

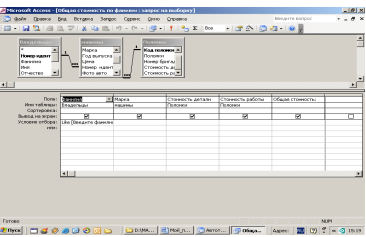
*Перекрестный запрос* - это операция построения таблицы для вычисления итоговых значений на основе существующей таблицы или запроса. Перекрестный запрос создается в предположении, что исходная таблица содержит необходимые данные для формирования заголовков строк и столбцов новой таблицы.

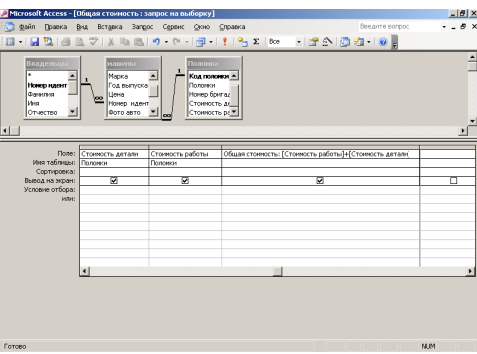
*Инструктаж из техники безопасности*

*Инструктаж к выполнению практической работы*

#### IV. Закрепление знаний, формирование навыков.

##### **Практическая работа «Создание запросов»**

№	Задание	Алгоритм выполнения
1.	Выполните запрос по таблице <b>Машины</b> .	Нажмите кнопку Создать, Запрос, добавить таблицу <b>Машины</b> , выполните запрос: <i>Марка, Цена, Год выпуска</i> . Сохраните запрос под именем Запрос1.
2.	Выполните запрос по таблицам <b>Владельцы</b> и <b>Машины</b> .	 Нажмите кнопку Создать, Запрос, добавить таблицу <b>Машины</b> , выбрать поля <i>Марка</i> , добавить таблицу <b>Владельцы</b> , выбрать поля: <i>Фамилия, Имя</i> . Сохранить Запрос2.
3.	Выполните запрос по таблицам <b>Владельцы</b> и <b>Машины</b> .	Выполните Создать запрос, Конструктор. Добавить таблицы: <b>Машины, Владельцы</b> . Перенести поля <i>Фамилия, Имя, Идентификационный номер</i> с таблицы <b>Владельцы</b> , условие отбора: Like[Введите фамилию]

		Перенести поле <i>Марка</i> с таблицы <b>Машины</b> . Сохраните запрос под именем Запрос3.
4.	Выполните запрос по таблицам <b>Машины, Поломки</b> .	Создать запрос, Конструктор. Добавить таблицы <b>Машины</b> : перенести поле <i>Марка</i> , таблица <b>Поломки</b> : перенести поля <i>Код поломки, Поломка</i> , условие отбора <i>Код поломки=100 или 200</i> Сохранить под именем Запрос4.
5.	Выполните запрос по таблице <b>Персонал</b> .	 Создать запрос, Конструктор. Добавить таблицу <b>Персонал</b> , перенести поля <i>Фамилия, Имя, Отчество, Мед_страховка</i> , условие отбора <i>Истина</i> по полю <i>Мед_Страховка</i> Сохранить под именем Запрос5.
6.	Выполните запрос по таблицам <b>Поломки, Владельцы, Машины</b> .	Создать, Конструктор. Добавить таблицы <b>Поломки, Владельцы, Машины</b> . Перенести поля: <i>Фамилия, Номер идентификационный, Марка, Стоимость деталей, Стоимость работы</i> В последнем поле ввести: <i>Общая стоимость: [Стоимость работы]+[Стоимость детали]</i> Сохранить запрос под именем Запрос6.
7.	Дополните предыдущий запрос: условие отбора – Введите фамилию	Скопировать запрос <i>Общая стоимость</i> . Сохранить Запрос7. В условии отбора по полю <i>Фамилия</i> введите <i>Like [Введите фамилию] &amp; "*" (запрос по первой любой букве)</i>
8.	Выполните запрос «Срок выполнения работ» по таблицам <b>Машины, Владельцы</b> .	Создать запрос, Конструктор. Добавить таблицы <b>Машины, Владельцы</b> . Перенести поля: <i>Марка, Фамилия, Номер машины, Дата сдачи, Дата возврата</i> . В последнем поле ввести: <i>Срок: [Дата возврата]-[Дата сдачи]</i> Сохранить под именем Запрос8.
9.	Самостоятельно выполните запрос	<i>Дата сдачи в ремонт &lt;20.08.2006, поломка – крыло, марка машины, фамилия владельца</i> . Сохраните под именем Запрос9.
10.	Самостоятельно выполните запрос	Выполнить запрос сотрудников, работающих в 1 бригаде, имеющих медицинскую страховку. Сохраните под именем Запрос.10
11.	Самостоятельно выполните запрос	Выполните запрос сотрудников, имеющих 6 символов в фамилии («?????») и работающими механиками. Сохраните под именем Запрос11
12.	Самостоятельно выполните запрос	Выполните запрос о наличии машин (марка, цена, год выпуска) каждого владельца по его идентификационному номеру или фамилии. Отсортируйте записи в порядке убывания в поле <i>Цена</i> . Сохраните под именем Запрос12
13.	Скопируйте <b>Запрос 6</b> , переименовав его в <b>Запрос13</b>	Выполните перекрестный запрос, выбрав вместо таблицы <b>Запрос13</b> . Выберите поле <i>Фамилия</i> в качестве заголовков

		<p>строк, поле Поломки в качестве заголовков столбцов. В качестве вычисления – Сумма.</p> <table border="1"> <thead> <tr> <th>Фамилия</th> <th>Итоговое значение</th> <th>Общая сумма</th> <th>дверь</th> <th>крыло</th> <th>кузов</th> <th>покрас</th> </tr> </thead> <tbody> <tr> <td>▶ Буров</td> <td></td> <td>300</td> <td></td> <td></td> <td>300</td> <td></td> </tr> <tr> <td>Григорьев</td> <td></td> <td>500</td> <td></td> <td>500</td> <td></td> <td></td> </tr> <tr> <td>Дубров</td> <td></td> <td>300</td> <td></td> <td></td> <td>300</td> <td></td> </tr> <tr> <td>Иванов</td> <td></td> <td>2100</td> <td></td> <td>500</td> <td>600</td> <td>1000</td> </tr> <tr> <td>Москалёв</td> <td></td> <td>2100</td> <td></td> <td>1500</td> <td>600</td> <td></td> </tr> <tr> <td>Сергеев</td> <td></td> <td>1500</td> <td></td> <td>500</td> <td></td> <td>1000</td> </tr> <tr> <td>Сидоров</td> <td></td> <td>1100</td> <td></td> <td>500</td> <td>600</td> <td></td> </tr> <tr> <td>Трофимов</td> <td></td> <td>300</td> <td></td> <td></td> <td>300</td> <td></td> </tr> <tr> <td>Фёдоров</td> <td></td> <td>424</td> <td>424</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>						Фамилия	Итоговое значение	Общая сумма	дверь	крыло	кузов	покрас	▶ Буров		300			300		Григорьев		500		500			Дубров		300			300		Иванов		2100		500	600	1000	Москалёв		2100		1500	600		Сергеев		1500		500		1000	Сидоров		1100		500	600		Трофимов		300			300		Фёдоров		424	424			
Фамилия	Итоговое значение	Общая сумма	дверь	крыло	кузов	покрас																																																																							
▶ Буров		300			300																																																																								
Григорьев		500		500																																																																									
Дубров		300			300																																																																								
Иванов		2100		500	600	1000																																																																							
Москалёв		2100		1500	600																																																																								
Сергеев		1500		500		1000																																																																							
Сидоров		1100		500	600																																																																								
Трофимов		300			300																																																																								
Фёдоров		424	424																																																																										
14.	Выполните пере- крёстный запрос по таблице <b>Персонал</b> .	<table border="1"> <thead> <tr> <th>Фамилия</th> <th>1</th> <th>2</th> <th>3</th> </tr> </thead> <tbody> <tr> <td>▶ Андреев</td> <td></td> <td>механик</td> <td></td> </tr> <tr> <td>Болихин</td> <td>мастер</td> <td></td> <td></td> </tr> <tr> <td>Вноеев</td> <td>механик</td> <td></td> <td></td> </tr> <tr> <td>Дивясов</td> <td></td> <td>механик</td> <td></td> </tr> <tr> <td>Земная</td> <td></td> <td></td> <td>дизайнер</td> </tr> <tr> <td>Зиновьев</td> <td>сторож</td> <td></td> <td></td> </tr> <tr> <td>Косюхин</td> <td>механик</td> <td></td> <td></td> </tr> <tr> <td>Крепостник</td> <td></td> <td>менеджер</td> <td></td> </tr> <tr> <td>Приходько</td> <td>зам.дир.</td> <td></td> <td></td> </tr> <tr> <td>Роднин</td> <td></td> <td></td> <td>механик</td> </tr> <tr> <td>Чеснов</td> <td></td> <td></td> <td>механик</td> </tr> </tbody> </table>						Фамилия	1	2	3	▶ Андреев		механик		Болихин	мастер			Вноеев	механик			Дивясов		механик		Земная			дизайнер	Зиновьев	сторож			Косюхин	механик			Крепостник		менеджер		Приходько	зам.дир.			Роднин			механик	Чеснов			механик																						
Фамилия	1	2	3																																																																										
▶ Андреев		механик																																																																											
Болихин	мастер																																																																												
Вноеев	механик																																																																												
Дивясов		механик																																																																											
Земная			дизайнер																																																																										
Зиновьев	сторож																																																																												
Косюхин	механик																																																																												
Крепостник		менеджер																																																																											
Приходько	зам.дир.																																																																												
Роднин			механик																																																																										
Чеснов			механик																																																																										

#### VI. Подведение итогов.

Вопросы:

- Какие задачи мы ставили перед собой в начале урока? Достигли ли мы их?
- Работу с какими объектами СУБД мы отрабатывали?
- Какие операции с данными можно выполнять посредством запросов?
- Изменяется ли информация в базе данных?
- Создается ли новая таблица при выполнении запросов?
- Можно ли назвать запрос фильтром? Почему?

#### VII. Домашнее задание.

- Повторить этапы создания базы данных.
- Какие операции в табличном процессоре похожие с работой запросов в базах данных?
- Создать 6 запросов по своей базе данных.

### Урок 7

**Тема урока:** Практическая работа «Формы. Отчеты»

**Тип урока:** повторение, изучение нового материала, практическая работа (комбинированный)

#### Цели урока:

- ✓ ознакомить учащихся с объектами баз данных: формами и отчетами, их назначением; дать представление о форме как элементе интерфейса пользователя;
- ✓ научить учеников пользоваться формами для введения, редактирования, изъятия данных
- ✓ развивать логико-алгоритмическое мышление, внимание, зрительную память;
- ✓ развивать навыки самоанализа собственной деятельности, воспитывать у учащихся навыки работы в паре, настойчивость, эмоциональную направленность на практическую деятельность, эстетический вкус при форматировании форм, отчётов.

#### Структура урока

I. Организационный момент	2 мин
II. Проверка домашнего задания	5 мин
III. Подготовка к основному этапу (мотивация учебной деятельности, объяснение нового материала)	10 мин
IV. Закрепление знаний, формирование навыков	20 мин

- V. Итоги урока
- VI. Домашнее задание

5 мин  
3 мин

### Ход урока

**I.** Организационный момент.

**II.** Проверка домашнего задания.

Обсуждение вопросов:

- В каком виде мы чаще всего организуем данные?

*Прогнозируемые ответы:* в виде таблицы.

- Какое программное обеспечение для этого вы предложите?

*Прогнозируемые ответы:* Можно использовать текстовый процессор, табличный процессор, систему управления базами данных.

- Сравните возможности перечисленных программных продуктов относительно возможностей обработки информации.

*Прогнозируемые ответы.* В текстовом процессоре можно оформлять таблицы, но возможности ограничиваются форматированием и печатью таблиц. Табличный процессор дает возможность обрабатывать информацию: упорядочивать, изменять, выполнять математические вычисления, фильтровать, упорядочивать данные. СУБД – программы которые предназначены для работы с большим количеством информации организованной специальным образом. СУБД включают у себя некоторые возможности как табличного процессора так и текстового редактора.

- Какие СУБД вам известны?
- Какие объекты баз данных вы знаете?
- Для чего используют таблицы?
- Какая функция запросов?
- Для чего в запросах используют \*, &, # ?
- Для чего используется выражение Like?

**III.** Подготовка к основному этапу.

Для более комфортной работы в разных СУБД существуют специальные объекты – **формы**. Вообще можно не знать, что такое база данных, но с помощью форм с ней работать (например: оплата коммунальных услуг, кассиры в железнодорожных и авиа кассах и т.д. )

**Формы** – это объекты базы данных для обеспечения интерфейса пользователя. Посредством форм можно пересматривать и редактировать базы данных. Формы создаются на основе таблиц. Использование форм значительно упрощает работу с БД. **Отчеты** – это формы для печати. Они создаются на основе таблиц или запросов. Самый простой способ создать отчет – поручить сделать это программе Access позволяет строить **автоотчеты** (то есть отчеты, создаваемые автоматически) двух типов: «в столбец» и «ленточные»

*Инструктаж из техники безопасности*


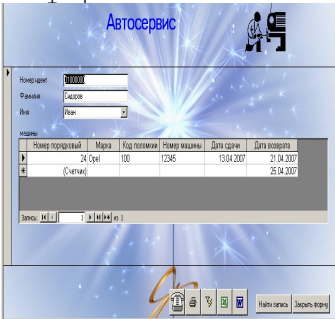








*Инструктаж к выполнению практической работы*

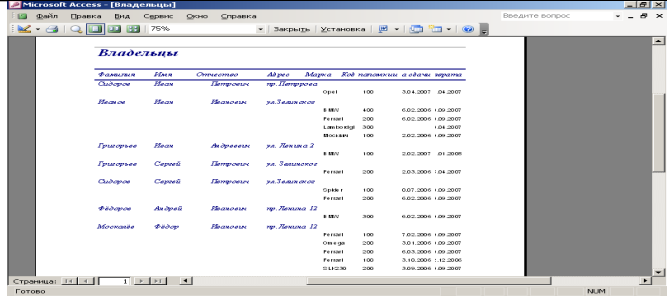
**IV.** Закрепление знаний и формирование навыков

#### Практическая работа «Создание форм, отчетов»

1. Открыть базу данных «Автосалон»
2. Выполнить следующие задания.

№	Задание	Алгоритм выполнения
1.	Создание формы на основе одной таблицы с помощью Мастера форм.	В окне базы данных щёлкните по кнопке <i>Формы</i> . Щелкните по кнопке <i>Создать</i> , выберите <i>Мастер форм</i> , таблицу <b>Владельцы</b> , включите в форму все поля. Выберите переключатель в <i>Один столбец</i> , щелкните <i>Далее</i> . Выберите стиль <i>Рисовая бумага</i> . Сохраните форму под

		именем Форма1 Введите в форму 2 записи.
2.	Создание подчиненной формы на базе нескольких таблиц с помощью Мастера форм.	В окне базы данных щёлкните по кнопке <b>Формы</b> . Щелкните по кнопке <i>Создать</i> , выберите <i>Мастер форм</i> , таблицу <b>Владельцы</b> , включите в форму поля: <i>Код идентификационный, Фамилия, Имя</i> . Выберите таблицу <b>Машины</b> , в ней поля: <i>Номер порядковый, Марка, Код поломки, Номер машины, Дата сдачи, Дата возврата</i> . Выберите вид представления данных - подчинённые формы. Выберите переключатель <i>Табличный</i> , щелкните <i>Далее</i> . Выберите стиль <i>Официальный</i> . Сохраните форму под именем <b>Форма2</b> . Введите в подчиненную форму на имя Иванова Ивана (номер иденти. -10000000) ещё одну автомашину.
3.	Настройка макета формы.	Откройте <b>Форму1</b> в режиме <i>Конструктора</i> . Измените стиль формы нажав кнопку  на <i>Рисовую бумагу</i>
4.	Настройка макета формы. 	Откройте <b>Форму2</b> в режиме <i>Конструктора</i> . Добавьте фоновый рисунок: Нажмите на кнопку <i>Свойства</i>  , выберите <i>Форма</i> , выберите рисунок, добавьте рисунок. Настройте макет формы. Добавьте на форму логотип заведения (подходящий имеющийся рисунок - кнопка  или созданный самостоятельно при помощи кнопок    )
5.	Настройка макета формы.	Откройте <i>Форму1</i> в режиме <i>Конструктора</i> . Добавьте кнопку <i>Закреть форму</i> . Нажмите на кнопку <i>Панель элементов</i> -  . Появится панель на которой выберите кнопку, создайте её на форме. В открывшемся диалоговом окне выберите функции кнопки, рисунок, текст: <i>Категории - Работа с формой, Действия -Закреть форму</i> . Аналогично создайте кнопки <i>Найти запись, Запуск Excel, запуск Word, запуск любого приложения (например любимой игры), Применение фильтра по выделенному, Печать формы, Автонабор номера</i> . Все кнопки выполните одного размера при помощи <i>Свойств</i>  , ширину и высоту кнопки выбрать самостоятельно. 

6.	Создайте самостоятельно два отчёта по таблицам <b>Владельцы</b> и <b>Персонал</b> с помощью <b>Мастера отчётов</b> .	
----	--	---

## V. Итоги урока

Оценивание результатов. Работы.

Обсуждение вопросов.

- Какую цель мы ставили в начале урока?
- Достигли ли мы поставленной цели?
- С какими объектами БД мы познакомились на сегодняшнем уроке?

## VI. Домашнее задание.

Подготовиться к контрольной работе:

- повторить: основные определения;
- составить логическую схему знаний по всей теме на листе А4;
- создать две формы и два отчёта по собственной базе данных.

## Урок 8

**Тема урока:** Тематическое оценивание по теме: «Базы данных. Системы управления базами данных»

**Тип урока:** контрольная работа.

**Цели урока:**

- ✓ оценить уровень достижений учеников по теме «БД. СУБД»
- ✓ развивать мыслительные способности: анализ, синтез, сравнение, обобщение.
- ✓ воспитывать самостоятельность, умение рационально использовать время, умение реально оценить свои знания при выборе разноуровневой карточки-задания.

**Оборудование:** разноуровневые карточки-задания к контрольной работе, работающие базы данных.

### Структура урока

I. Организационный момент	3 мин
II. Основной этап	40 мин
III. Домашнее задание	2 мин

### Ход урока

**I.** Организационный момент.

**II.** Основной этап.

*Инструктаж по выполнению контрольной работы.*

Задание выполняется в тетради для контрольной работы, содержит теоретическую и практическую часть (2-3 теоретических вопроса, практические задания создаются по работающей базе данных 5-6 заданий.)

#### Пример карточки-задания:

1. Какие объекты предназначены для ввода и просмотра данных? **(16)**
2. Какая основная характеристика полей? **(16)**
3. Какими бывают модели хранения данных? **(16)**
4. В БД **ПЕРЕГОВОРЫ**, Структура таблиц:  
 АБОНЕНТ: Ф.И.О. абонента, адрес, номер телефона;  
 ПЕРЕГОВОРЫ: номер телефона, дата разговора, код города, количество минут переговоров, стоимость переговоров;  
 ЦЕНА: код города, стоимость одной минуты.

**Применить** фильтры и найти первую запись абонента с заданным номером, звонившего в огород с указанным кодом. (1б)

5. **Составить** запросы, в которых необходимо:

– определить количество звонков абонента с заданной фамилией (1б);

– общую стоимость переговоров абонента с заданной фамилией (фамилию задать самим); (1б)

– Ф.И.О., номер телефона, дата переговоров, стоимость одной минуты переговоров. (2б)

6. Составить запрос на редактирование любой таблицы. (1б)

7. Создать формы, которые содержат такие поля: номер телефона, Ф.И.О. абонента, дата переговоров, стоимость переговоров (1б)

8. Создать любой отчёт (1б)

### III. Домашнее задание

Выполнить в тетради задания контрольной работы, пользуясь конспектом (для тех, кто не справился на уроке)

*Дополнительно* **Примеры заданий для создания базы данных.**

#### **Вариант №1.**

1. Создать базу данных согласно структуре таблиц данных:

**АБОНЕНТ:** Ф.И.О. абонента, адрес, номер телефона;

**ПЕРЕГОВОРЫ:** номер телефона, дата разговора, код города, количество минут переговоров, стоимость переговоров;

**ЦЕНА:** код города, стоимость одной минуты.

2. Применить фильтры и найти первую запись абонента с заданным номером, звонившего в город с указанным кодом.

3. Составить запросы и:

– определить количество звонков и общую стоимость переговоров абонента с заданной фамилией (фамилию задать самим);

– определить общую стоимость всех переговоров.

4. Составить запрос на выборку таких данных:

Ф.И.О., номер телефона, дата переговоров, стоимость одной минуты переговоров.

5. Составить запрос на редактирование любой таблицы.

6. Создать формы, которые должны содержать такие поля: номер телефона, Ф.И.О. абонента, дата переговоров, стоимость переговоров; номер телефона, дата переговоров, код города, стоимость 1 минуты.

#### **Вариант 2.**

1. Создать базу данных согласно структуре таблиц данных:

**ТОВАР:** шифр товара, наименование товара, единица измерения, цена;

**КОЛИЧЕСТВО:** шифр товара, количество проданного товара за день, дата продажи товара, фамилия продавца, номер отдела;

**ПРОДАВЕЦ:** фамилия продавца, оклад, стаж работы.

2. Применить фильтры и найти записи за 20 июня 1996 г. для 6-го отдела.

4. Составить запросы и:

– определить стоимость товара с заданным шифром;

– подсчитать количество товара с одним конкретным шифром, проданного одним конкретным продавцом за один месяц;

– определить среднюю цену всех товаров.

5. Составить запрос на выборку таких данных: шифр товара, фамилия продавца, стаж работы.

6. Составить запрос на редактирование любой таблицы.

7. Создать формы, которые должны содержать такие поля: наименование товара, дата продажи, количество; наименование товара, дата продажи, номер отдела, фамилия продавца и стаж его работы.



## Тема 5. Глобальная сеть Интернет ( 6 годин)

№ урока	Тема урока	
1	Компьютерные сети. Глобальная сеть Интернет.	
2	Основные услуги Интернет. Служба WWW.	
3	Основы языка гипертекстовой разметки HTML.	
4	Информационно-поисковые системы.	
5	Электронная почта в глобальной сети Интернет.	
6	Форумы и их назначение.	

### Урок 1

**Тема урока:** Компьютерные сети. Глобальная сеть Интернет.

#### Цель урока:

- Познакомиться с видами и характеристиками компьютерных сетей, принципами работы глобальной сети Интернет.
- Отработать навыки подключения к сети.

#### Ход урока

#### I. Мотивация и актуализация

1. Необходимость обмена информацией на больших расстояниях.
2. Время, необходимое для получения желаемой информации.

#### II. Представление темы, целей и ожидаемых результатов

Результатом данной работы должно стать следующее:

- изучение и закрепление базовых понятий, касающихся компьютерных сетей;
- самостоятельная регистрация и подключение к сети Интернет;

#### III. Подача необходимой информации

1. Виды компьютерных сетей, топология, основные характеристики. Сервер. Рабочая станция.
2. Информационная связь в сети Интернет. Аппаратные, программные, информационные ресурсы современных сетей.
3. Протокол. Идентификация компьютеров в сети. Адресация в Интернет. Провайдер.

**Компьютерная сеть** – совокупность компьютеров, соединенных с помощью каналов связи в единую систему

По **территориальному признаку** сети делятся на: Локальные, Региональные, Глобальные

**Локальная сеть** – сеть, объединяющая компьютеры, установленные в одном помещении.

**Региональные сети** – сеть, объединяющая компьютеры в пределах одного региона (города, страны)

**Глобальная сеть** (Интернет) – сеть, объединяющая многие локальные, региональные, корпоративные сети и включающая десятки миллионов компьютеров.

**ЛВС (локальные вычислительные сети)** можно разделить на: одноранговые и сети типа "клиент-сервер".

#### Одноранговая сеть

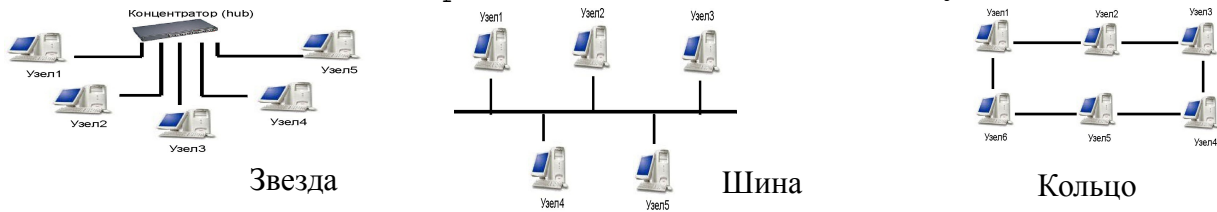
В данном типе сетей нет единого центра управления взаимодействием рабочих станций и нет единого устройства для хранения данных. Пользователю сети могут быть доступны все устройства, подключенные к другим станциям (диски, принтеры).

#### Сети типа "клиент-сервер"

Основные компоненты сетей этого типа:

- **СЕРВЕР** – это компьютер, который предоставляет в сеть свои ресурсы или обеспечивает выполнение определенных услуг, управляет работой сети.
- **РАБОЧАЯ СТАНЦИЯ** (клиент) – это компьютер, который пользуется услугами сервера.

**Топология сетей** – геометрическая схема соединений узлов сети



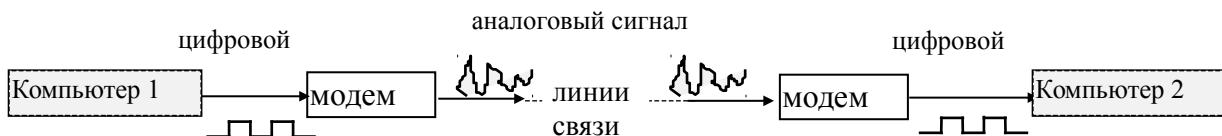
(a) Аппаратная составляющая

- компьютер
- телефонная линия
- модем (преобразует цифровые сигналы терминала или компьютера в аналоговые сигналы, которые могут передаваться по телефонному оборудованию).

Характеристики модема

- **Скорость передачи данных.** Ее можно определить числом модуляций сигнала в секунду и измерять в **бодах**.
- **Протокол.** Протокол – набор правил обмена информацией между компьютерами, установленными по взаимному соглашению. Тип протокола определяет и скорость передачи. Протокол v.34 – позволяет принимать данные со скоростью 33600 бит в секунду (bps). v.92 – принятый в 2000 году, скорость передачи 57600 bps.

Схема передачи сигнала



(b) Программная составляющая

- *операционная система, поддерживающая работу локальных сетей*
- *браузер*

(c) Информационная составляющая

- **провайдер** – организация, которая предоставляет доступ в Интернет.

**Идентификация компьютеров в сети**

Все компьютеры, подключенные к Интернету, находят друг друга в автоматическом режиме. Это возможно благодаря тому, что каждый компьютер имеет свой адрес, называемый **IP-адресом**.

**IP-адрес** представляет собой 32-разрядный (или 4-байтный) код и состоит из двух частей. Первая часть IP-адреса обозначает адрес подсети Интернета, к которой подключен узел, а вторая часть – адрес локального узла внутри подсети.

Для записи адресов в Интернете используется более наглядная система, основанная на **доменных именах** серверов, или сокращенно **DNS** (Domain Name Server – доменное имя сервера). Так же, как и IP-адрес, доменное имя однозначно определяет положение сервера в Интернете. Доменное имя строится по иерархическому принципу.

1. На самом верхнем уровне (домен верхнего уровня) обычно находится название страны, например, uk (Великобритания), ru (Россия) или ua (Украина). Хотя могут быть и исключения, как в случае США. Вместо названия страны США могут быть сокращения, отвечающие

типу организации, которой принадлежит домен: com (коммерческий домен), gov (правительственный), mil (военный), edu (образовательный), net (сетевой), org (прочих организаций). (см.. таблицу)

2. Слева от домена верхнего уровня через точку дописывается обозначение города, штата или организации. Однако эта часть имени может отсутствовать.
3. Левее от обозначения города (организации) через точку следует обозначение сервера, которое таким образом занимает крайнюю левую позицию в доменном имени.

В итоге доменное имя сервера (проще говоря, домен, или Hostname) может иметь следующий вид: **www.microsoft.com**

#### IV. Интерактивное упражнение

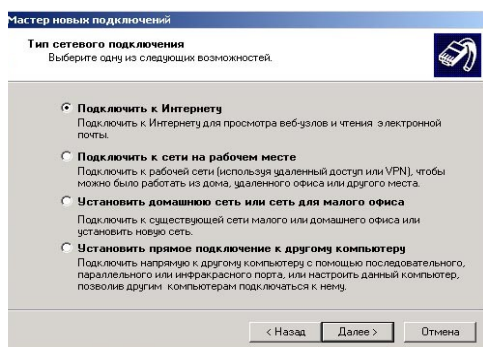
**Задача:** создать собственное подключение к Интернет.

Исходные данные (даются учителем): телефон провайдера, имя пользователя (логин), пароль.

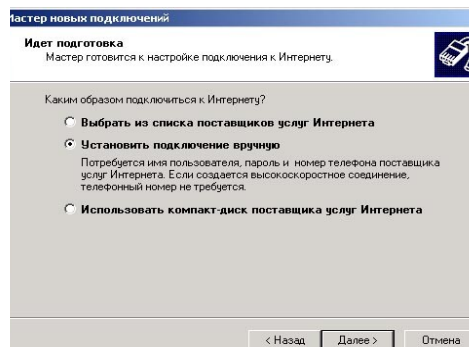
Порядок выполнения:

1. Открыть: *Панель управления-Свойства обозревателя.*
2. Выбрать вкладку *Подключения.*
3. Нажать кнопку *Добавить.*
4. Следуя указаниям мастера подключения к Интернет, выполнить действия:

- Выбрать тип сетевого подключения



- Способ подключения

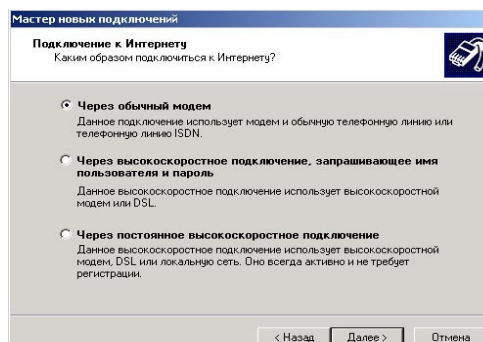


- Вид подключения

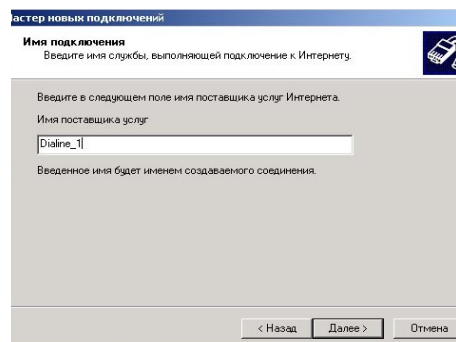
- Ввести имя подключения

(например Dialine\_<номер компьютера>)

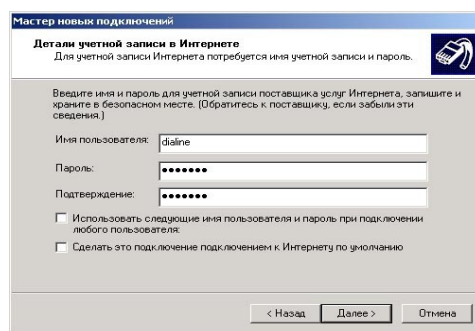
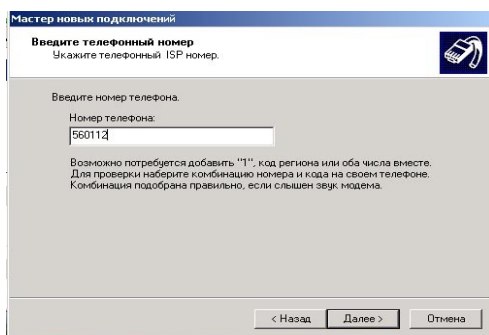
- Указать номер телефона провайдера



- Ввести имя пользователя и пароль



- Завершить работу мастера подключения



- Запустить созданное подключение через ярлык на Рабочем столе
- V. Подведение итогов, оценивание результатов урока**

Вопросы для защиты:

1. Какие существуют виды компьютерных сетей? Их назначение.
2. Каким образом передаются данные в компьютерных сетях?
3. Какие составляющие необходимы для организации компьютерной сети?
4. Описать метод идентификации компьютеров в сети Интернет. Понятие протокола.

## Урок 2

**Тема урока:** Основные услуги Интернет. Служба WWW.

**Цель урока:**

- Познакомиться с основными услугами сети Интернет. Рассмотреть возможности и изучить основные понятия службы WWW.
- Изучить структуру программы-обозревателя. Научиться открывать сайты с помощью программы методом указания URL-адреса.

### Ход урока

**I. Мотивация и актуализация**

1. Всегда ли есть нужная книга в библиотеке? Как получить последние новости? Высокая стоимость международной телефонной связи. Невозможность участвовать в научных конференциях.
2. Технические возможности Интернет.

**II. Представление темы, целей и ожидаемых результатов**

Результатом данной работы должно стать следующее:

- ознакомление с видами служб сети Интернет, в частности службы WWW;
- изучение и приобретение навыков самостоятельной работы с программой - обозревателем;
- выполнение практического задания.

**III. Подача необходимой информации**

1. Основные службы Интернет. Технология клиент-сервер.
2. Служба WWW. Web-страницы и сайты. Принципы работы с гипертекстом. Гипертекстовые связи.
3. Программы-обозреватели. Программа Internet Explorer. URL-адрес.

Службы (сервисы) - это виды услуг, которые оказываются серверами сети Internet.

В истории Интернет существовали разные виды сервисов, одни из которых в настоящее время уже не используются, другие постепенно теряют свою популярность, в то время как третьи переживают свой расцвет. Перечислим те из сервисов, которые не потеряли своей актуальности на данный момент:

- *World Wide Web* - всемирная паутина - служба поиска и просмотра гипертекстовых документов, включающих графику, звук и видео.
- *E-mail* - электронная почта-служба передачи электронных сообщений.
- *Usenet, News* - телеконференции, группы новостей - разновидность сетевой газеты или доски объявлений.
- *FTP* - служба передачи файлов.
- *ICQ* - служба для общения в реальном времени с помощью клавиатуры.
- *Telnet* - служба удаленного доступа к компьютерам.
- *Gopher* - служба доступа к информации с помощью иерархических каталогов.

Примечание: последние две службы угасающие.

**Клиент-сервер** – это технология взаимодействия компьютеров, программное обеспечение которых состоит из двух частей – **серверной**, установленной на одном из компьютеров, и **клиентской**, установленной на остальных компьютерах.

Установленные на компьютере программы «наделяют» его теми или иными функциями. Из них основной является возможность стать **клиентом или сервером**.

Разместить на компьютере файлы можно в том случае, если на нем установлена специальная программа для обслуживания (англ. to serve – обслуживать, соответствующее существительное – server).

**Программа-сервер** – программа, предоставляющая пользователю информационные ресурсы компьютера и отвечающая на запрос программы-клиента.

**Программа-клиент** – программа, запрашивающая определенные информационные услуги.

**WWW** (World Wide Web, англ. Всемирная паутина) – это служба поиска и просмотра гипертекстовых документов. Эти документы называются Web-страницы, а совокупность близких по смыслу или тематике и хранящихся вместе Web-страниц называется – Web-сайт или Web-узел.

Web-страницы могут включать в себя текст, рисунки, анимацию, звук, видео, а также активные элементы – небольшие программы, оживляющие страницу, делающие ее интерактивной.

Идея гипертекста проста: гипертекст – это текст, содержащий ссылку на другой документ, который может быть аналогичной Web-страницей. Гипертекст представлен в виде гиперссылок, выделенных на странице обычно подчеркиванием, цветом, по которым достаточно щелкнуть мышью, и будет осуществлен переход к другой Web-странице или загружен нужный файл. Именно потому, что страницы с помощью гиперссылок переплетены между собой, эту службу называют «паутина». Информация во всемирной хранится в памяти компьютеров – серверов и имеет адрес вида: **<http://www.narod.ru>**

#### **E-Mail (Electronic mail – электронная почта)**

Позволяет обмениваться сообщениями с помощью компьютера. Электронная почта похожа на обычную почту, но письма нужно не опускать в почтовый ящик, а отправлять и принимать с помощью специальных программ, установленных на компьютере. Письмо представляет собой обычный текстовый файл (можно вместе с письмом отправить любой двоичный файл) с заголовком в начале. E-mail адрес в Internet, в общем случае, состоит из имени пользователя и имени машины (например, dima@elvis.msk.ru).

#### **News (Usenet, Телеконференции)**

Usenet сейчас – это тысячи News-групп, каждая из которых содержит статьи с определенной темой. Статья представляет собой обычное письмо, только в отличие от электронной почты ее может прочитать любой, кто подпишется на эту группу. Часто Usenet называют телеконференциями или эхо – почтой, эхо – конференциями. Телеконференции действуют на основе уставов – правил принятых самими участниками. Usenet больше всего напоминает периодическую печать. Подписавшись на группы с интересующей темой, абонент сети регулярно получает статьи самых различных авторов, которые, также подписаны на эту группу. Но в отличие от газет и журналов, абонент может, например, не согласиться с автором и послать опровержение в ту же группу, или наоборот, добавить свою информацию в подтверждение. Одним словом, участвовать в дискуссии. Существует возможность

отправлять и получать статьи практически в/из любой сети мира. Поскольку число абонентов постоянно увеличивается, новые группы создаются непрерывно.

#### **IRC (Internet Relay Chat)**

Очень похож на News, но диалог ведется в реальном времени, как на селекторном совещании. Множество различных групп. Есть русскоязычные группы (например, сервер irc.funet.fi канал #russian). Одновременно поддерживается обмен сообщениями между двумя пользователями. Есть возможность передавать файлы по IRC-каналам.

#### **FTP (File Transfer Protocol)**

Позволяет пользователям одной машины получать доступ к файловой системе и получать (передавать) файл с другой машины (на другую машину). На многих машинах INTERNET имеются огромные коллекции (терабайты информации) архивов программных средств, всевозможной документации, художественной литературы, компьютерные фильмы, музыка и др. Как правило, этот сервис предоставляется бесплатно, при условии, что в качестве имени пользователя вы зададите anonymous, а в качестве пароля свой e-mail адрес.

**Программа-обозреватель** (броузер) – программа для просмотра Web-страниц.

**URL-адрес** – уникальный адрес каждой Web-страницы в сети Интернет.

URL-адрес состоит из схемы, хоста, пути.

Схема – указание на протокол, необходимый для доступа к ресурсу

Хост – доменный адрес компьютера, на котором находится ресурс

Путь – полный путь к файлу, в котором хранится документ.

Например: http://www.abc.com/main/

схема хост путь

#### **IV. Интерактивное упражнение**

**Задача:** используя программу – обозреватель, по заданным адресам открыть ряд предложенных сайтов, исследовать их структуру, сохранить полученные данные.

#### **Порядок работы:**

1. Осуществить вход в Интернет на Web-сайт по адресу **www.gala.net**. Ознакомиться с его структурой
2. Найти на данном сайте **www.gala.net** последнюю политическую новость, записать ее, сохранив в каталоге **Web\_inform** диска С как web-страницу
3. Осуществить вход на Web-сайт по адресу **www.google.ru**
4. Найти на открытом сайте **www.google.ru** прогноз погоды для одного из регионов Украины. Добавить данную ссылку в пункт меню **Избранное**
5. Выполнить вход на сайт **www.samsungmobile.com**. Ознакомиться с имеющимися ссылками
6. Сохранить информацию об одном из новых моделей мобильных телефонов Samsung в текстовом документе под именем **Samsung\_2007** в папке **Мои документы** диска С. В документ включить текстовую и **графическую информацию, а также одну из ссылок.**

#### **V. Подведение итогов, оценивание результатов урока**

Задание	1	2	3	4	5	6
Баллы	1	3	1	2	1	4
Отметка о выполнении						

### Урок 3

**Тема урока:** Основы языка гипертекстовой разметки HTML.

**Цель урока:**

- Освоить основные теги форматирования символов, создания гиперссылок и графики языка гипертекстовой разметки HTML.
- Приобрести практические навыки по созданию Web-документов, которые содержат теги форматирования символов, гиперссылки и графику.
- Научиться создавать Web-страницы в режиме текстового редактора.

**Ход урока**

**I.** Мотивация и актуализация

1. Основное количество информации в Интернет пользователи получают через службу WWW. Создание хороших сайтов – актуальный вопрос. Спрос на Web-программистов.
2. Способы создания Web-документов: автоматизированный и вручную. Объем Web- страниц и его зависимость от способа создания документа. Необходимость знания языка гипертекстовой разметки.

**II.** Представление темы, целей и ожидаемых результатов

Результатом данной работы должно стать следующее:

- создание Web-документа с помощью языка HTML в режиме Блокнота и его сохранение;
- освоение методов просмотра Web-страницы в режиме документа и в режиме HTML-кода.

**III.** Подача необходимой информации





## 1. Программы для создания Web-документов.

### Структурные теги

<b>&lt;HTML&gt;</b>	Данный тег используется для открытия HTML-документа. Каждая Web-страница начинается тегом <b>&lt;HTML&gt;</b> и заканчивается закрывающим тегом <b>&lt;/HTML&gt;</b> .
<b>&lt;HEAD&gt;</b>	Любой HTML-документ состоит как минимум из двух частей: заголовка и собственно документа. Данный тег определяет заголовок Web-страницы и должен иметь обязательный закрывающий тег <b>&lt;/HEAD&gt;</b> . Также обязательным атрибутом этого тега является тег <b>&lt;TITLE&gt;</b> .
<b>&lt;TITLE&gt;</b>	Каждый HTML-документ имеет название, заключенное между тегами <b>&lt;TITLE&gt;</b> и <b>&lt;/TITLE&gt;</b> . По названию документа HTML браузеры могут найти информацию, поэтому место для названия всегда определено - оно находится вверху и отдельно от содержимого HTML-документа. Отображается название в заголовке окна браузера, поэтому на него накладывается ограничение: максимальная длина названия - 40 символов.
<b>&lt;BODY&gt;</b>	Данный тег включает в себе непосредственно документ. Также необходим закрывающий тег <b>&lt;/BODY&gt;</b> .

## 2. Язык HTML. Структурные теги. Теги форматирования текста.

### Тег заголовка

<b>&lt;H1&gt;...&lt;/H1&gt;</b> ----- <b>&lt;H6&gt;...&lt;/H6&gt;</b>	Теги <b>&lt;H<sub>i</sub>&gt;...&lt;/H<sub>i</sub>&gt;</b> (где <i>i</i> — цифра от 1 до 6) описывают заголовки шести различных уровней от H1 до H6. Текст, окруженный тегами <b>&lt;H1&gt;...&lt;/H1&gt;</b> , получается самым большим - это основной заголовок. Если текст окружен тегами <b>&lt;H2&gt;...&lt;/H2&gt;</b> , то он выглядит несколько меньше (подзаголовок); текст внутри <b>&lt;H3&gt;...&lt;/H3&gt;</b> еще меньше и так далее до <b>&lt;H6&gt;...&lt;/H6&gt;</b> .
---	---

### Тег создания абзаца

<b>&lt;P&gt;...&lt;/P&gt;</b>	Этот тег описывает абзац. Все, что находится между <b>&lt;P&gt;</b> и <b>&lt;/P&gt;</b> , воспринимается как <u>один абзац</u> . Закрывающий тег <b>&lt;/P&gt;</b> не обязателен.
-------------------------------	---

**Теги форматирования символов**

<b>&lt;B&gt;...&lt;/B&gt;</b>	С помощью этого контейнера можно установить <b>жирный шрифт</b> .
<b>&lt;I&gt;, &lt;/I&gt;</b>	Эти теги устанавливает <b>наклонный шрифт</b> .
<b>&lt;U&gt;...&lt;/U&gt;</b>	Эти теги задают <b>подчеркнутое написание символов</b> .
<b>&lt;TT&gt;...&lt;/TT&gt;</b>	Текст, размещенный между этими тегами, будет написан шрифтом, имитирующим пишущую машинку, то есть имеющим фиксированную ширину символа.

## 3. Вставка графических объектов и требования к ним.

## Теги форматирования

<b>&lt;FONT&gt;...&lt;/FONT&gt;</b>	Тег описания шрифта текста, помещенного в контейнер. Имеет следующие атрибуты:
атрибут <b>face="..."</b>	Задаст имя шрифта или несколько возможных шрифтов.
атрибут <b>size="..."</b>	Устанавливает размер шрифта
атрибут <b>color="..."</b>	Задаст цвет шрифта.
атрибут <b>style="..."</b>	Определяет описание стиля (данные атрибута берутся из CSS)

### 4. Создание гиперссылок.

#### Основные понятия:

**HTML** – язык гипертекстовой разметки;

**Тег** – команда в языке гипертекстовой разметки

**Атрибут** – это параметр, предназначенный для модифицирования документа. Его значение заключается в кавычки. Атрибуты располагаются внутри тегов.

Например:

```
<IMG src="image1.jpg"> (Вставка графического файла в Web-документ)
<A href="indexpic/image2.jpg" <font size=+1
color=red>Погружение</font></A>
```

(Гиперссылка в виде текста "Погружение", вызывающая файл im-

### Тег вставки изображения

<b>&lt;IMG&gt;</b>	Тег для вставки изображения на Web-страницу. Имеет следующие атрибуты:
атрибут <b>src="..."</b>	определяет URL-адрес графического файла

age2.jpg из папки **indexpic**. Размер шрифта текста гиперссылки на 1

### Тег вставки гиперссылки

<b>&lt;A&gt;...&lt;/A&gt;</b>	Тег для описания указателя ссылки, как-то: подчеркнутый текст, маркер — точка, звездочка; пиктограммы — экранные кнопки, рекламные вставки и т.д. На указателе нужно щелкнуть мышью, чтобы сделать переход по ссылке.
<p>Общий вид структуры тега ссылки:      <b>&lt;A href="filename"&gt;текст_ссылки&lt;/A&gt;</b>  <b>filename</b> — имя файла или адрес Internet, на который необходимо сослаться;  <b>текст_ссылки</b> — текст гипертекстовой ссылки, который будет непосредственно показан в HTML-документе.</p>	

ед. больше текущего, цвет – красный)

#### IV. Интерактивное упражнение

Задача: Создать Web-страницу согласно предложенному плану.

#### Порядок работы:

1. Откройте программу Блокнот и наберите текст следующего содержания:

```
<html>
<head>
<title> Первая Web-страница </title>
</head>
<body>
    В.Высоцкий
</body>
</html>
```

Новые дескрипторы и атрибуты: **<html>**, **<head>**, **<title>**, **<body>**

2. Сохраните файл под именем **Фамилия.html** в папке **С:\Мои документы\11 класс** (Файл – Сохранить как)
3. Открыть созданный файл с помощью программы-обозревателя.
4. Отобразить документ в режиме HTML-кода (Вид – Просмотр HTML-кода)
5. Вставьте в html-документ отрывок из стихотворения В. Высоцкого, соблюдая расположение строки и используя теги форматирования символов:

**Я не люблю  
 Я не люблю фатального исхода,  
 От жизни никогда не устаю.  
 Я не люблю любое время года,  
 Когда веселых песен не пою.**

Новые дескрипторы и атрибуты: `<br>`, `<p>`, `<hr>`, `align`, `color`, `size`

6. Отредактировать текст так, чтобы словосочетание "Я не люблю" перед стихотворением было выделено как заголовок. (Парный тег `<H1>...</H1>`)
7. Создайте гиперссылку вызывающую поисковый сервер **rambler.ru** (`<A href="адрес ресурса"><font color=цвет size=+1>Текст ссылки</A>`)

Новые дескрипторы и атрибуты: `<A>`, `href`

#### V. Подведение итогов, оценивание результатов урока

Задание	1	2	3	4	5	6	7
Кол-во баллов	1	2	1	1	3	2	2
Отметка о выполнении							

Всего баллов \_\_\_\_\_

### Урок 4

**Тема урока:** Информационно-поисковые системы.

#### Цель урока:

- Освоить основные понятия, связанные с поиском информации в глобальной сети.
- Приобрести практические навыки работы с информационно-поисковыми системами.
- Научиться применять программу Internet Explorer для поиска необходимой информации.

#### Ход урока

##### I. Мотивация и актуализация

1. Где взять интересующую информацию, если недоступны соответствующие книги и периодическая литература?
2. Как разобраться в огромном количестве информации глобальной компьютерной сети?
3. По каким принципам производить быстрый и эффективный поиск информации?

##### II. Представление темы, целей и ожидаемых результатов

Результатом данной работы должно стать следующее:

- ознакомление с поисковыми системами, принципами их работы и способами поиска;
- исследование структуры и возможностей выбранной поисковой системы;
- поиск и сохранение необходимой информации согласно заданию.

##### III. Подача необходимой информации

1. Поисковые серверы. Поисковые каталоги. Специализированные поисковые системы.
2. Принцип работы поисковых систем. Поиск по ключевым словам.
3. Общие сведения о синтаксисе языка запросов в поисковых системах.

### Поисковые серверы

Сеть Интернет растёт очень быстрыми темпами, и найти нужную информацию среди десятка млн. документов становится всё сложнее. Для поиска информации используются специальные поисковые серверы, которые содержат более или менее полную и постоянно обновляемую информацию о Web-страницах, файлах и других документах, хранящихся на десятках миллионов серверов Интернет.

Различные поисковые серверы могут использовать различные поисковые механизмы поиска, хранения и предоставления пользователю информации. Однако общим является то, что к моменту запроса пользователя вся информация о документах Интернет в компактном виде хранится в базе данных поискового сервера.

Имеющиеся поисковые системы можно условно разделить на три группы:

- 1) справочники Интернет;
- 2) поисковые системы общего назначения;
- 3) специализированные поисковые системы;

*Справочник Интернет (каталог)* является аналогом тематического указателя в библиотеке: он предоставляет перечень наиболее важных документов по заданной теме. Примеры таких систем – поисковые серверы:

<http://WWW.yahoo.com>

<http://narod.yandex.ru/rubrics/>

Поисковые системы общего назначения позволяют находить документы во всемирной паутине по ключевым словам. Принцип, на котором основано большинство таких систем, состоит в том, что специальные программы – роботы автоматически «обходят» WWW-серверы, читают и индексируют все встречающиеся документы, выделяя при этом ключевые слова, относящиеся к данному документу, и запоминая их с URL этого документа в базе данных. Большинство поисковых систем разрешают также автору новые Web-страницы самому внести информацию в базу данных.

Примером такой поисковой системы является Rambler: <http://www.rambler.ru>

Специализированные поисковые системы позволяют находить вам информацию, находящуюся в других информационных «слоях» Интернет, смежных со всемирной паутиной, например, FTP-серверах. Примером такой системы является Lycos: <http://ftpsearch.lycos.com>

Так как информация в Интернет постоянно меняется, поисковые роботы не всегда успевают отследить все эти изменения. Информация, хранящаяся в базе данных поисковых серверов, может отличаться от реального состояния Интернет, и поэтому пользователь может получить ссылку на несуществующий или перемещённый документ.

Интерфейс поисковых систем примерно одинаков. Начальная страница поисковой системы содержит список разделов, уточняющих область поиска, и поле поиска. В поле поиска пользователь может ввести ключевые слова для поиска документа, то есть слова, которые, по мнению пользователя, позволяют идентифицировать документ.

### **Как найти адрес (URL)?**

Способов несколько. Самый простой способ – это увидеть доменные имена в различных рекламных материалах. В настоящее время любая уважающая себя фирма, в том числе любое издательство книг, журналов, редакции телеканалов имеют свой адрес в Internet и обычно указывают его на упаковке, в книгах, журналах, в эфире телепрограмм. Как правило, указывается короткий адрес, состоящий только из доменного имени, например: <http://www.life.com.ua>. Введите такой адрес в адресную строку и гуляйте по сайту.

Но как быть, если адрес совершенно неизвестен? Для этого обратимся к хранилищам URL адресов, к специальным поисковым системам. **Поисковые системы** (поисковые каталоги, поисковые серверы) выполнены в виде Web-страниц, но на этих страницах располагается не информация как таковая, а аннотированные ссылки на другие ресурсы (страницы, сайты).

Для вызова поисковой системы необходимо в адресной строке программы Internet Explorer указать ее адрес. Поисковых систем в сети Internet сейчас достаточно много. В таблице ниже перечислены лишь наиболее популярные поисковые системы.

Наименование	Адрес
Яндекс (рус.)	<a href="http://www.yandex.ru">http://www.yandex.ru</a>
Rambler (рус.)	<a href="http://www.rambler.ru">http://www.rambler.ru</a>
Апорт (рус.)	<a href="http://www.aport.ru">http://www.aport.ru</a>
Google (рус.)	<a href="http://www.google.com.ru">http://www.google.com.ru</a>

Как работает поисковая система? Первый этап работы поисковой системы – это **индексирование информации**, находящейся в Internet. Сетевой робот поисковой системы просматривает огромное количество страниц и заносит адреса и краткое содержание этих страниц к себе в базу данных, точнее в поисковый индекс. Владельцы сайтов будут очень рады, если их сайт просмотрит сетевой робот. Для того чтобы робот обязательно просмотрел сайт и внес его в поисковый индекс, владельцы сами регистрируют свои сайты в поисковой системе (как правило, в нескольких поисковых системах). Процесс занесения адреса и описания страницы в поисковую систему называется **индексация**. Второй этап работы поисковой системы – это выдача накопленной и отобранной, отсортированной и классифицированной информации по запросам пользователей.

#### Поиск по ключевым словам

Наиболее простым и результативным поиском является поиск по **ключевым словам**. Давайте подключимся к Internet, откроем программу Internet Explorer и введем в адресную строку адрес поисковой системы: <http://www.yandex.ru>. Мы подробно разберемся с правилами поиска по ключевым словам в поисковой системе **Яндекс**. В других системах эти правила могут отличаться, но основные положения будут идентичны.

#### Основной алгоритм поиска следующий:

1. ввести ключевые слова в **поле запроса**
2. нажать кнопку **Найти**.

После нажатия кнопки **Найти**, Яндекс выведет **список ссылок** на документы, наиболее точно соответствующие запросу.

#### Синтаксис языка запросов

В системе **Яндекс** существует специальный **язык запросов**, использовать который более сложно, чем форму расширенного поиска но при его использовании можно получить наилучший результат. **Поисковый запрос** вводится в поисковое поле, он может содержать ключевые сло-

ва и специальные символы, позволяющие установить взаимосвязи между этими словами и ввести дополнительные параметры. Некоторые из этих символов представлено в следующей таблице.

Символ	Назначение	Пример
" "	поиск фразы	"серый волк"
+	обязательное наличие слова в найденном документе	+быть или +не быть
~	не должно быть слова в пределах предложения (И НЕ)	банки ~ закон
!	искать только указанную форму слова	!Наутилус

#### IV. Интерактивное упражнение

Задача: Найти предложенную информацию с помощью предложенных поисковых серверов.

#### Порядок работы:

1. Запишите в тетрадь адреса известных вам
  - поисковых тематических каталогов \_\_\_\_\_
  - поисковых серверов \_\_\_\_\_
2. Из заданного списка выберите русскоязычный поисковый сервер:
 

<a href="http://www.yandex.ru">www.yandex.ru</a>	<a href="http://www.rambler.ru">www.rambler.ru</a>
<a href="http://www.google.com">www.google.com</a>	<a href="http://www.altavista.com">www.altavista.com</a>
<a href="http://www.yahoo.com">www.yahoo.com</a>	<a href="http://www.aport.ru">www.aport.ru</a>
<a href="http://www.meta.ua">www.meta.ua</a>	<a href="http://www.infoseek.com">www.infoseek.com</a>
3. По выбранному адресу загрузите Web-страницу сервера. Ознакомьтесь со структурой главной страницы.
4. Используя тематический каталог, найдите информацию о лекарстве "Аспирин". Сохраните найденную информацию в файле **Аспирин (C:\Мои документы)**
5. Методом поиска по ключевым словам найдите данные о космическом корабле многоцелевого использования "Буран". Добавьте ссылку на найденный вами сайт в *Избранное*.
6. Найдите реферат на одну из предложенных тем:
  - История возникновения компьютеров
  - Компьютерные вирусы
  - Электронная почта
  - История возникновения Интернет
  - Основные услуги сети Интернет.
 Откройте реферат для просмотра. Сохраните его в папке **C:\Мои документы**.

#### V. Подведение итогов, оценивание результатов урока

Задание	1	2	3	4	5	6
Кол-во баллов	1	1	1	3	3	3
Отметка о выполнении						

### Урок 5

**Тема урока:** Электронная почта в глобальной сети Интернет.

#### **Цель урока:**

- Освоить основные приемы работы с программой Microsoft Outlook Express.
- Приобрести практические навыки создания, получения и отправки почтового сообщения.
- Научиться применять один из почтовых серверов для регистрации электронного почтового ящика.

#### **Ход урока**

#### I. Мотивация и актуализация

1. Возможно ли сегодня обойтись без личных контактов на расстояниях? Проблемы, связанные с обычной почтовой службой – время и средства.
2. Как заказать редкую книгу? Как отправить информацию человеку, живущему в далекой стране?
3. Возможности Интернет в области почтовых пересылок.

#### II. Представление темы, целей и ожидаемых результатов

Результатом данной работы должно стать следующее:

- ознакомление с почтовой программой-сервером и программой-клиентом;
- регистрация почтового ящика на одном из предложенных почтовых серверов;
- создание учетной записи в почтовой программе Outlook Express;
- выполнение задания по созданию и обработке сообщения.

#### III. Подача необходимой информации

1. Электронная почта в Интернет. Адресация в электронной почте.
2. Программы-серверы и программы-клиенты.
3. Настройка почтовой программы.
4. Создание и отправка сообщения.
5. Получение сообщения. Адресная книга.
6. Присоединение файлов.

**Электронная почта** – наиболее распространённая **услуга Интернет**, так как она является исторически первой информационной услугой компьютерных сетей и не требует обязательного наличия высокоскоростных и качественных линий связи.

Принципы функционирования электронной и обычной почты аналогичны. Вы посылаете людям письма по их конкретным адресам, они, в свою очередь, пишут вам на ваш почтовый адрес. Однако электронная почта имеет несколько серьёзных преимуществ перед обычной почтой. Наиболее важное из них – **это скорость пересылки сообщений**. Если письмо обычной почтой может идти до адресата дни и недели, то письмо, посланное по электронной почте, достигнет его за несколько секунд или, в худшем случае, за несколько часов. Другое преимущество в том, что электронное письмо может содержать **не только текстовое сообщение**, но и вложенные файлы (программы, графику, звук и т. д.). Однако не рекомендуется, чтобы размер пересылаемого файла был слишком большим, так как некоторые почтовые серверы вводят ограничение на размер пересылаемого сообщения. Любой пользователь Интернет может получить свой почтовый ящик на одном из почтовых серверов Интернет (обычно на **почтовом сервера провайдера**), в котором будут накапливаться передаваемые и получаемые электронные письма.

#### Адрес электронной почты

Адрес электронной почты записывается по определенной форме и состоит из двух частей: **имя \_ пользователя @ имя \_ сервера**

*Имя пользователя* имеет произвольный характер и задается самим пользователем.

*Имя сервера* жестко связано с выбором пользователем сервера, на котором он разместил свой почтовый ящик. Например: user-name@mail.ru

Для того чтобы отправить электронное письмо, отправитель должен подключиться к Интернет и передать на свой почтовый сервер сообщение. Почтовый сервер сразу же отправит это письмо через систему почтовых серверов Интернет на почтовый сервер получателя, которое попадает в его почтовый ящик. Получатель получит письмо после

того, как соединится с Интернет и «заберет» почту из своего почтового ящика.

Прием писем из Интернет осуществляется по протоколам:

- POP (Post Office Protocol) – протокол почтового офиса,
- IMAP (Internet Message Access Protocol) – протокол доступа к сообщениям сети Интернет

#### Функции почтовых программ

- Получение и чтение сообщений
- Подготовка и отправка сообщений
- Ведение адресной книги

#### Структура электронного письма

*Заголовок письма:*

**Кому: (To:) oleg@vgpi.voronezh.ru**

**Копии: (CC:) master@florida.nasa.gov, Nick@school.msk.ru  
(кому еще)**

**Тема: (Subject:) Happy Ney Year!**

**Текст сообщения**

Поле заголовка «Кому: » заполняется обязательно – это адрес получателя.

Поле «Копии» – при необходимости.

Поле «Тема» – желательно заполнять, оно помогает получателю ориентироваться в содержании, назначении приходящих к нему писем.

#### Краткое описание программы Outlook Express и работа в ней.

В Интернет роль почтальона выполняют программы-клиенты. Одной из таких программ является Outlook Express. Программа Outlook Express устанавливается на компьютер вместе с операционной системой Windows.

Интерфейс программы включает меню, панель инструментов. Для настройки программы на работу с Вашим почтовым ящиком, необходимо создать учетную запись. Для этого необходимо выбрать пункт меню "Сервис->Учетные записи...".

Появится диалоговое окно "Учетные записи Интернет".

Нажмите кнопку "Добавить" и выберите пункт "Почта...".

Появится диалоговое окно "Мастер подключения к Интернет".

Выполните указания мастера и заполните все необходимые поля.

В поле "Сервер входящих сообщений" необходимо ввести адрес сервера, с которого программа будет принимать почту, а в поле "Сервер исходящих сообщений" необходимо ввести адрес сервера, на который программа будет отправлять Ваши исходящие письма. Эти адреса могут быть предоставлены Вашим провайдером или необходимо будет прочесть раздел помощи на соответствующем Web-сайте в Интернет.

Далее необходимо ввести учетную запись и пароль.

После создания новая учетная запись будет отображаться в списке записей.

Теперь, когда вы создали учетную запись, можете проверить, не пришла ли почта на почтовый ящик, которой соответствует созданная учетная запись.

Для этого выберите пункт меню "Сервис->Доставить почту" и выберите соответствующую запись.

Если Вам пришли новые сообщения, то они будут отображаться в папке "Входящие".

Чтобы прочитать письмо, необходимо выбрать его в списке писем. Содержание письма будет отображаться ниже.



Чтобы написать кому-либо письмо, выберите пункт меню "Файл->Создать->Почтовое сообщение". Появится редактор создания сообщений. Укажите, кому Вы отсылаете сообщение и тему сообщения. Потом напишите текст сообщения и нажмите кнопку "Отправить" на панели инструментов. При этом программа попытается сразу отправить сообщение.

Для занесения данных в адресную книгу, выполните команды в группе Контакты: "Создать контакт->Имя->Заполнить поля адресной книги"

#### Краткие сведения о почтовом сервере на примере mail.ru

Запуск программы осуществляется из программы-обозревателя путем открытия Web-страницы по адресу [www.mail.ru](http://www.mail.ru)

##### **1. Структура главной страницы сервера mail.ru**

- строка поиска
- меню
- область регистрации
- блок новостей

##### **2. Структура почтовой программы сервера mail.ru**

- меню сервера
- строка заголовка с вкладками (почта, адреса, еженедельник...)
- Строка команд (проверить почту, написать письмо, найти письмо)
- Папки (входящие, сомнительные, отправленные, черновики, корзина)

##### **3. Порядок регистрации и структура окна регистрации**

- \* e-mail
- \* пароль
- \* подтверждение
- \* имя, фамилия, дата рожд., пол, страна, регион
- \* "Зарегистрировать почтовый ящик"

#### **IV. Интерактивное упражнение**

Задача: зарегистрироваться на одном из предложенных серверов и провести операции по созданию, получению и отправке электронного сообщения.

#### Порядок работы:

Выберите одну из предложенных почтовых служб: [www.yahoo.com](http://www.yahoo.com), [www.ukr.net](http://www.ukr.net), [www.yandex.ru](http://www.yandex.ru), [www.mail.ru](http://www.mail.ru).

Запустите на выполнение программу -обозреватель и введите выбранный URL-адрес.

1. Найдите элемент окна почтового сервера, с которого начинается регистрация. Заполните регистрационную карточку почтового ящика на сервере согласно указаниям и создайте свой учебный электронный ящик.
2. Запустите на выполнение программу Outlook Express. Создайте учетную запись (имя пользователя, адрес электронной почты, имена серверов входящих и исходящих сообщений сообщает учитель).
3. Заполните адресную книгу электронным адресом созданного в п.2 электронного ящика. В поле *Имя* запишите "Учебный ящик"
4. Создайте электронное сообщение на адрес учебного ящика, введя в поле *Копия* свой электронный адрес, в поле *Тема* - "Проверка почты", а в область сообщений - "Сообщение из Арктики".
5. Отправьте электронное сообщение. Выполните команду получения почты. Проверьте папку, в которой находится входящая корреспонденция.
6. Дайте ответ на сообщение, которое к вам пришло. Введите текст ответа - "Письмо доставлено". Присоедините к ответу файл

**Samsung\_2007**, находящийся в папке **Мои документы** диска С.  
Отправьте письмо.

**V. Подведение итогов, оценивание результатов урока**

Задание	1	2	3	4	5	6	7
Кол-во баллов	1	2	1	1	2	2	3
Отметка о выполнении							

**Урок 6**

**Тема урока:** Форумы и их назначение.

**Цель урока:**

- Познакомиться с понятием форума, его назначением и отличием от телеконференций.
- Приобрести практические навыки работы с форумами.
- Познакомиться с правилами этикета на форумах.

**Ход урока**

**I. Мотивация и актуализация**

1. Как привлечь к обсуждению актуальной темы максимальное число участников?
2. Кому и как задать вопрос по интересующей Вас тематике, ответ на который Вы не знаете?

**II. Представление темы, целей и ожидаемых результатов**

Результатом данной работы должно стать следующее:

- овладение теоретическим материалом по форумам, а также понимание их отличия от телеконференций;
- нахождение сайта, содержащего форум и его анализ;
- создание собственного сообщения по обсуждаемой теме.

**III. Подача необходимой информации**

1. Что такое телеконференции USENET? По какому принципу строятся названия телеконференций?
2. Чем USENET как источник информации отличается от World Wide Web (WWW)?
3. Что такое Web-форумы и чем они отличаются от телеконференций USENET?
4. Как работать в форуме?

Что такое USENET?

**USENET** - это сеть тематических телеконференций (по-английски они называются *newsgroups* - "группы новостей"). Приставка "теле" указывает здесь не на наличие телевизионной связи между участниками телеконференции, а всего-навсего на их возможную удаленность друг от друга, поэтому телеконференции не нужно путать с видеоконференциями. Каждая телеконференция имеет определенное название и посвящена определенной теме (например, в конференции под названием *sci.med.pharmacy* обсуждаются вопросы, связанные с фармакологией, а в *soc.history.medieval* - с историей Средневековья). Сейчас существуют десятки тысяч телеконференций. Название телеконференции USENET состоит из нескольких "полей", разделенных точками (например, *sci.physics* или *soc.culture.russian.moderated*). Крайнее левое поле обозначает так называемую иерархию, которой принадлежит данная конференция, последующие поля конкретизируют ее тематику. Некоторые иерархии USENET перечислены в таблице ниже:

comp	(от computer) темы, связанные с компьютерами и программированием
misc	(от miscellaneous) - темы, не входящие ни в один из остальных классов или относящиеся сразу к нескольким
news	вопросы развития самой системы телеконференций

	USENET, советы и инструкции для новичков (см. news.answers, news.newusers.questions и др.)
rec	(от recreation) - отдых, хобби, развлечения
sci	(от science) - темы, связанные с наукой
soc	(от social) - вопросы общественной жизни
talk	конференции для любителей "просто поговорить" на ту или иную тему
humanities	гуманитарные науки и вопросы
biz	бизнес
alt	"альтернативные" конференции на какие угодно темы; их можно создавать по упрощенной процедуре
Региональ- ные иерар- хии	ориентированы на участников из определенной страны или региона, например de - Германия, dk - Дания, ukr - Украина, ba - Bay Area (район Сан-Франциско в США)
relcom	Региональная иерархия, ориентированная в основном на аудиторию из стран бывшего СССР (название происходит от названия московской фирмы <u>Relcom</u> [2] - одного из первых Интернет-провайдеров в бывшем СССР).
fido	Конференции компьютерной сети FIDONET, транслируемые в USENET
fido7	Русскоязычные конференции компьютерной сети FIDONET, транслируемые в USENET (цифра "7" - телефонный код бывшего СССР и нынешней России)
Корпора- тивные иерархии	Создаются и поддерживаются компанией или организаци-ей; многие корпоративные телеконференции предназна-чены только для "своих", но некоторые из них открыты для публичного доступа - например, netscape, microsoft или русскоязычные медицинские medlux

#### Чем USENET как источник информации отличается от World Wide Web (WWW)?

Информация, которая попадает в USENET, меньше "отфильтрована" (т.е. содержит больше "мусора"), но зато гораздо более динамична - каждое заметное событие, будь то появление новой версии компьютерной программы или только что закончившийся футбольный матч, моментально вызывает "волну" обсуждения на USENET'е. Со временем часть этой информации попадает и на WWW. Если Вы не можете найти нужной Вам информации, Вы можете задать вопрос и получить на него ответ - и в этом важное преимущество USENET'а перед "пассивным" поиском на WWW. Вообще же USENET и WWW как источники информации хорошо дополняют друг друга.

#### Работа в форумах.

Общение в форуме происходит не в реальном времени. Форумы похожи на электронную почту, но, в отличие от нее, сообщения, размещаемые на форуме адресуются всем и все могут ответить на них. Форум представляет собой электронную доску, разделенную на отдельные части для общения на определенные темы. Все желающие могут прийти и прочитать всё написанное на доске и написать свою мысль по поводу затрагиваемых вопросов. Форумы, как правило, организуются по определенной тематике и создаются для того, чтоб помочь людям в вопросах, связанных с тематикой сайта. Большинство форумов требует регистрации. Регистрируясь, вы обязуетесь выполнять правила, существующие на форумах. После регистрации Вы сможете войти на форум, как полноправный участник.

Чтоб создать новую тему, необходимо найти кнопку или гиперссылку с надписью "Новая тема", или "New topic". Вы сможете получить ответ на созданное Вами сообщение не сразу, поэтому периодически необходимо проверят наличие ответа.

#### FAQ

Аббревиатурой FAQ (от английского Frequently Asked Questions - "часто задаваемые вопросы") обозначают подборки ответов на часто задаваемые вопросы по определенной теме. По-русски иногда такие подборки называют ЧАВО (Частые Вопросы и Ответы). Ниже вопросы помечены буквой Q ("questions"), а ответы - буквой А ("answers").

FAQ также размещаются на досках объявлений, при этом позволяют формировать и отсылать свой вопрос. Как и в форумах, отосланное сообщение появляется на доске объявлений через некоторое время, а ниже помещается вопрос.

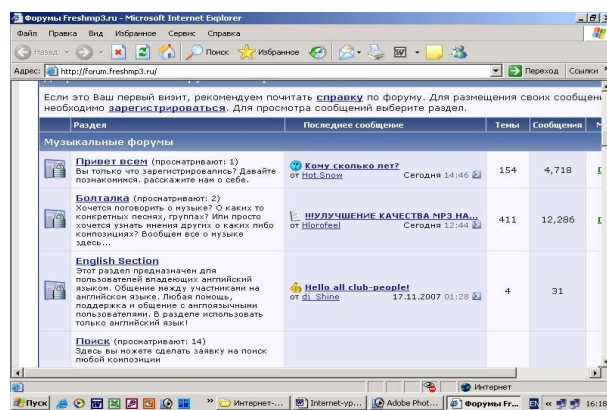
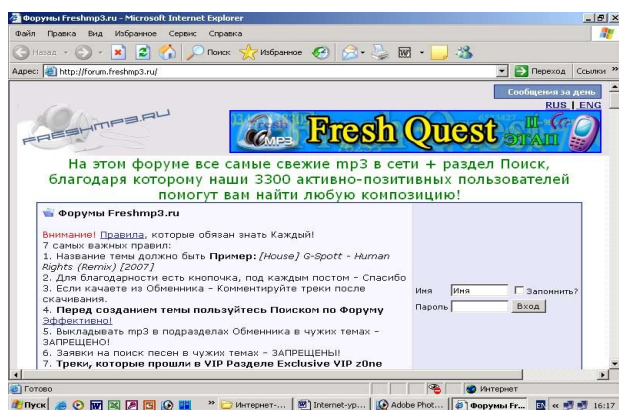
#### Модерируемые и немодерируемые конференции?

Сообщения, отправляемые участниками дискуссии в *немодерируемую* конференцию, публикуются автоматически. Если конференция *модерируется*, то сообщения сначала попадают к модератору, который решает, публиковать их или нет. Модератор может отклонять сообщения, которые, по его мнению, не соответствуют тематике конференции, нарушают принятые в ней правила поведения либо не содержат новой информации. В некоторых конференциях есть *робомодератор*. Это специальная программа, которая оценивает сообщение по некоторым формальным критериям и, в зависимости от результата, может автоматически отклонять сообщение, автоматически публиковать либо пересылать на рассмотрение модератора-человека.

#### **IV. Интерактивное упражнение**

**Задача:** По заданным адресам открыть сайты, содержащие форумы и FAQ, выполнить предложенные задания.

#### Порядок работы:



7. Загрузить сайт <http://forum.freshmp3.ru/>
8. Ознакомиться с правилами работы в форуме и сохраните текст данных правил в файле **Правила-форум (C:\Мои документы)**
9. Ознакомиться со структурой сайта и описать ее. Просмотреть имеющиеся сообщения.
10. Выбрать раздел **Поиск** – подраздел **Поиск русской музыки**. Просмотреть объявления за последнюю неделю.
11. Скачать интересующий Вас музыкальный файл или текст песни. Сохранить информацию в **C:\Музыка-mp3**.
12. Открыть страницу <http://www.germes-lada.ru/sale/lada>.
13. Просмотреть информацию по интересующему Вас автомобилю. Четко сформулировать вопрос по рассматриваемой модели автомобиля. Перейти по ссылке FAQ.

14. Просмотрите задаваемые вопросы и ответы на них. В разделе "Ваш вопрос" запишите сформулированный ранее вопрос. Укажите адрес Вашей электронной почты. Отправьте сообщение.

15. С помощью поискового сервера самостоятельно найти сайт с форумом на другую тему, просмотреть сообщения и сохранить информацию каталоге **Web\_inform** диска С в виде Web-страницы.

**V.** Подведение итогов, оценивание результатов урока

<b>Задание</b>	<b>1-2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6-7</b>	<b>8</b>	<b>9</b>
<b>Кол-во баллов</b>	1	1	1	2	2	2	3
<b>Отметка о выполнении</b>							

**Курява Тетяна Дмитрівна** – учитель інформатики

НВК «Школи - ліцею» №69

**Курявий Олег Володимирович** – учитель інформатики та

математики спеціалізованої школи №5.

## **Тема 6. Комп'ютерне моделювання. Основи алгоритмізації та програмування**

### **6.1. Комп'ютерне моделювання. Основи алгоритмізації. (5 годин)**

№ урока	Тема урока	
1	Моделювання як метод пізнання. Види моделей.	
2	Основні етапи розв'язку прикладної задачі з застосуванням ПК. Математична модель. Алгоритм, його властивості.	
3	Базові алгоритмічні структури. Види алгоритмів.	
4	Розв'язок задач.	
5	Тематичне оцінювання з теми "Основи алгоритмізації"	

#### **Урок 1**

**Тема уроку:** Моделювання як метод пізнання. Види моделей.

**Тип уроку:** урок засвоєння нових знань

**Мети уроку:** – засвоїти поняття моделі, види моделей, поняття інформаційної моделі;

- засвоїти основні етапи рішення задачі за допомогою ЕОМ;
- розвивати критичне мислення, пам'ять;
- формувати вольові якості, наполегливість;
- виховувати потребу та вміння працювати в колективі, інформаційну грамотність, виховувати увагу, інтерес до предмета.

**Дидактичні матеріали:** Таблиця «Моделі», «Етапи розв'язку задач за допомогою комп'ютера»

**Додаткова підготовка:** підготувати таблицю, розробити тест.

#### **Структура уроку.**

- І. Оргмомент. Мотивація навчальної діяльності школярів (2 хв.)
- ІІ. Повідомлення теми, мети та задач уроку ( 2 хв.)
- ІІІ. Вивчення теоретичного матеріалу та його осмислення учнями (23 хв.)
- ІV. Закріплення учнями матеріалу (15 хв.)
- V. Підсумки уроку та домашнє завдання (3 хв.)

#### **Хід уроку.**

**I.** 1) Оргмомент

2) Бесіда «Моделі в житті людини»

Людина в будь-якій діяльності постійно користується моделями. У дитинстві граються іграшками – зменшеними копіями реальних об'єктів. Пізніше знайомляться з іншими моделями. У кожного віку свої моделі. Дорослі люди також користуються моделями.

Спробуємо навести моделі із життя людини в різних віках.

Чому не можна дослідити сам оригінал, навіщо створювати моделі? (не може існувати в часі, реально не можна побачити, процес небезпечний для життя)

У процесі пізнання і практичної діяльності людина широко застосовує різноманітні моделі. Моделювання допомагає людині приймати обґрунтовані рішення та передбачати наслідки своєї діяльності.

**ІІ.** Оголошення теми, мети та задач уроку.

### III. 1) модель

**Модель** – це праобраз, опис або зображення якогось реального об'єкту. Якщо розширювати це поняття – це формальний опис, який відображає най суттєві властивості об'єкту (явища, процесу) і зв'язки між ними. Під час введення виду моделі в ході бесіди учні за допомогою вчителя наводять приклади моделей. За способом представлення моделі бувають:

Матеріальні – це образи реальних об'єктів: іграшки, глобус, макети.

Абстрактні – описи, формули, зображення, схеми, графіки.

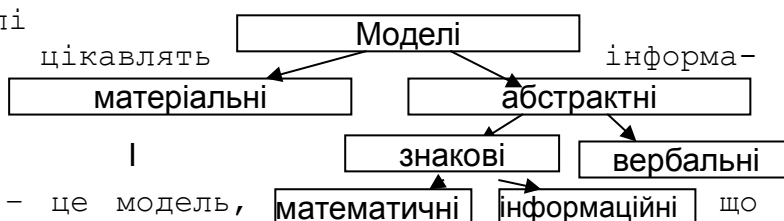
Знакові – за допомогою знаків, їх можна зобразити на папері.

Вербальні – моделі, які отримані в наслідок висновків, представлені в розумовій або словесній формі.

**Математична модель** – це модель, що сформульована мовою математики і логіки (логічні задачі, фізичні процеси).

#### 2) інформаційні моделі

В інформатиці нас цікавлять інформаційні моделі, тобто такі, що стосуються інформаційних процесів.



Інформаційна модель – це модель, що описує інформаційні процеси або містить інформацію про властивості і стан об'єктів, процесів, явищ (загадки, тексти довідників, енциклопедій).

Знакові моделі можуть бути комп'ютерні та некомп'ютерні.

Комп'ютерна модель – це модель, яка реалізована за допомогою програмних засобів.

#### 3) моделювання

Моделювання – створення та дослідження моделей. Якщо за допомогою комп'ютера – комп'ютерне моделювання.

Нехай нам необхідно вирішити будь-яку задачу, і ми хочемо скористатися послугами ПК. З чого почати? **При поясненні складати схему на дошці та в зошиті.** Насамперед потрібно розібратися що дано, що потрібно одержати, як зв'язані вихідні дані і результати.

1 ЕТАП: Постановка задачі.

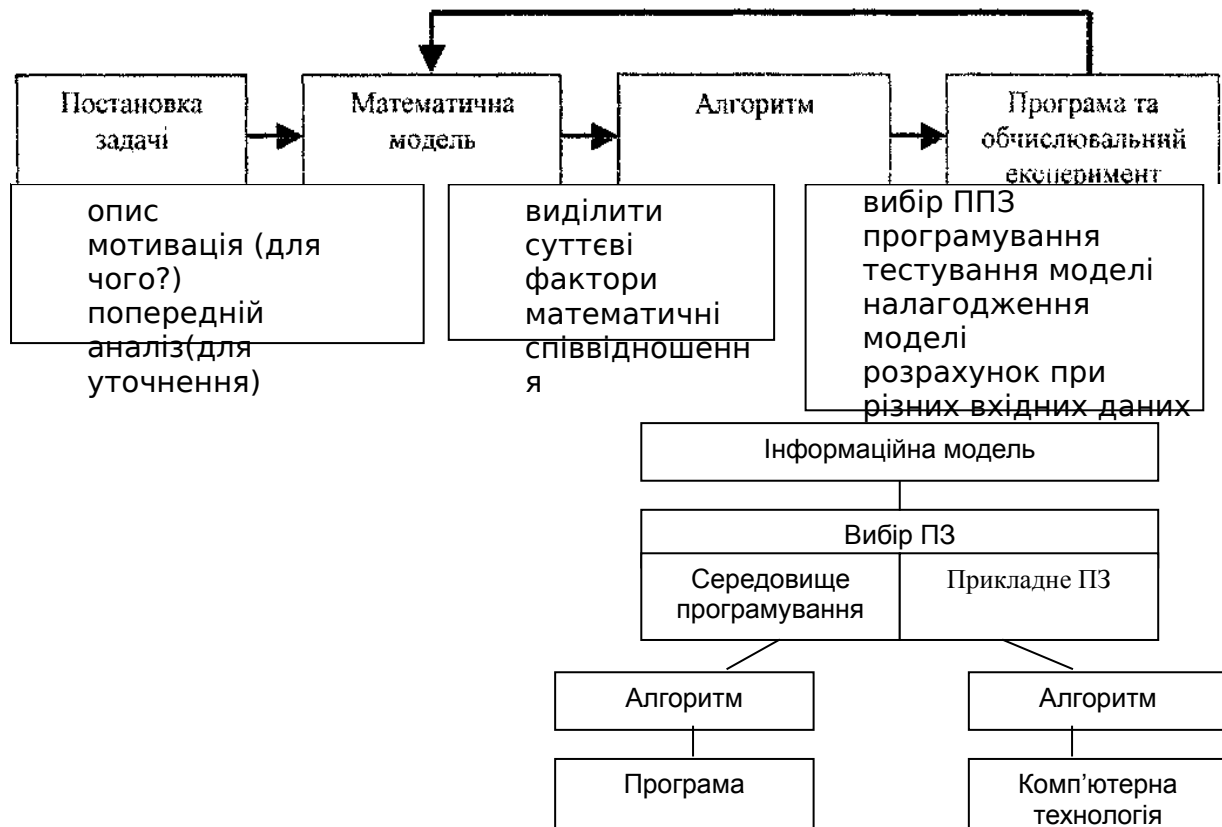
Чітко сформулювати задачу – це значить висловити ті припущення, що дозволяють у море інформації про досліджуване чи явище об'єкті вивудити вихідні дані, визначити, що буде служити результатом, і яка зв'язок між вихідними даними і результатом. Усе це – припущення, вихідні дані, результати і зв'язок між ними – **модель задачі.**

2 ЕТАП: Далі необхідно всі припущення, вихідні дані і результати представити числами, а зв'язок між ними – математичними співвідношеннями. Виконавши такий "переклад" задачі на мову математики, одержуємо математичну модель задачі.

3 ЕТАП: Щоб машина зробила необхідні обчислення й одержала відповідь, потрібно скласти для неї чітку інструкцію, сірого вказати необхідну послідовність дій. Така інструкція називається АЛГОРИТМОМ рішення задачі.

4 ЕТАП: ЕОМ можуть виконувати алгоритми без участі людини, автоматично. Але для цей алгоритм повинний бути записаний у "зрозумілої" для машини формі, тобто на спеціальній мові, яка зрозуміла ЕОМ, що називається **мовою програмування.**

Алгоритм, записаний мовою програмування, називається **програмою**.  
Уточнення моделі



Для розробки комп'ютерної моделі важливим є вибір програмного забезпечення.

#### 1) Інтерактивна вправа «Посиденьки»

Усі отримують аркуш з однаковим текстом. Прочитати текст, та поставити до нього по 3 питання, записати їх в зошит. Відповіді на запитання повинні бути в тексті. Далі по колу кожен учень задає питання сусідові праворуч або будь-кому з однокласників на вибір. Учень, якому задали запитання, відповідає на нього, користуючись текстом. Так проходить круг чи два.

Зразок тексту

### Класифікація моделей

Як ви вже переконалися, кількість моделей та їх різноманітність дуже велика. Щоб не загубитися в цьому розмаїтті, необхідно мати певну класифікацію моделей. Розглянемо найбільш суттєві ознаки, що за ними класифікуються моделі: галузі використання; урахування в моделі фактора часу; спосіб представлення моделей.

Розглянемо моделі з позиції галузі використання. Вони бувають:

*навчальні*— наочні посібники, тренажери, навчальні програми;

*дослідні* — створюються для дослідження характеристик реального об'єкта (модель теплохода перевіряється на стійкість, а модель літака - на аеродинамічні характеристики);  
*науково-технічні* - для дослідження процесів та явищ (ядерний реактор або синхротрон);

*ігрові моделі* - для вивчення можливої поведінки об'єкта в запрограмованих або непередбачених ситуаціях (військові, економічні, спортивні ігри тощо);

*імітаційні моделі*— виконується імітація реальної ситуації, що багато повторюється для вивчення реальних обставин (випробування лікарських препаратів на мишах або інших тваринах, політ собаки в космос).

З урахуванням *фактора часу* моделі можуть бути *динамічні* та *статичні*. В першому випадку над об'єктом виконуються дослідження протягом деякого терміну, а в другому - робиться одноразовий зріз стану (наприклад, постійний нагляд сімейного лікаря та одноразове обстеження в поліклініці).



- 2) тест
1. Інформаційною (знаковою) моделлю є:
    - а) анатомічний муляж;      в) модель корабля;
    - б) макет будинку;      г) діаграми.
  2. Генеалогічне дерево сім'ї є:
    - а) таблична інформаційна модель;
    - б) ієрархічна інформаційна модель;
    - в) мережева інформаційна модель;
    - г) предметна інформаційна модель.
  3. Предметною моделлю є:
    - а) макет літака;      в) креслення;
    - б) карта;      г) діаграма.
  4. Інформаційною моделлю організації занять у школі є:
    - а) звіт правил поведінки учнів;      в) розклад уроків;
    - б) список класу;      г) перелік підручників.
  5. Результатом процесу формалізації є:
    - а) описова модель;      в) графічна модель;
    - б) математична модель;      г) предметна модель.
  6. Модель є заміщенням вивчаємого об'єкту іншим об'єктом, який відображає:
    - а) всі сторони даного об'єкту; в) деякі сторони даного об'єкту;
    - б) істотні сторони даного об'єкту;
    - г) неістотні сторони даного об'єкту.
  7. Модель містить інформації:
    - а) стільки ж, скільки об'єкт, що моделюється;
    - б) менше, ніж об'єкт, що моделюється;
    - в) більше, ніж об'єкт, що моделюється;
    - г) не містить інформації;
  8. Інформаційна модель розв'язку розрахункової задачі є:
    - а) фізична;      б) хімічна;      в) математична;
    - г) обчислювальна.
- IV.** 1) Підсумки уроку:
- Чому поняття «моделі» є одним з ключових у курсі інформатики? Назвіть основні етапи розв'язку задач за допомогою ПК.
- 2) Домашнє завдання: конспект, Зарецька Глава 9§ 1,2. Скласти кросворд з новими поняттями і термінами.

## Урок 2

**Тема уроку:** Основні етапи розв'язку прикладної задачі з застосуванням ПК. Математична модель. Алгоритм та його властивості.

**Тип уроку:** урок засвоєння нових знань

**Мети уроку:** - закріпити поняття інформаційної моделі; основні етапи рішення задачі за допомогою ЕОМ;

- засвоїти поняття алгоритму, його властивості форми та способи подання алгоритмів;
- відробити навички створення математичних моделей;
- розвивати алгоритмічне мислення, пам'ять;
- формувати вольові якості, ініціативність, наполегливість;
- виховувати потребу та вміння працювати в колективі, інформаційну грамотність, виховувати увагу, інтерес до предмета.

**Дидактичні матеріали:** «Етапи розв'язку задач за допомогою комп'ютера», "Екологічна модель", "Приклади математичних моделей".

**Додаткова підготовка:**

### Структура уроку.

- I. Оргмомент. Актуалізація опорних знань (5 хв.)
- II. Повідомлення теми, мети та задач уроку ( 2 хв.)
- III. Вивчення теоретичного матеріалу та його осмислення учнями (20 хв.)
- IV. Закріплення учнями матеріалу (15 хв.)
- V. Підсумки уроку та домашнє завдання (3 хв.)

### Хід уроку.

- I. 1) Оргмомент
- 2) Фронтальне опитування:
  - Перелічити основні етапи рішення задач за допомогою ПК.
  - З чого починається рішення задачі?
  - Для чого необхідна постановка задачі?
  - Для чого необхідно описувати методи рішення задач?
  - Які етапи рішення задач на ЕОМ виконує людина?
  - На якому етапі використовується ЕОМ?
  - Що таке алгоритм? Що називається програмою?
  - Що ми розуміємо під обчислювальним експериментом?
  - Що таке математична модель?
- 3) зібрати кросворди для перевірки.
- II. Повідомити тему, мету та задачі уроку.
- III. 1) Успішно вирішувати задачі можна тільки при чіткому й однозначному визначенні кінцевих результатів.

**Точність постановки задачі** – це можливість забезпечити однакове розуміння цих задач різними людьми. Точність визначень і формулювань повинна бути такою, щоб за допомогою їхній можна було однозначно мислити. Така точність формулювань і тверджень характерна для математики. А тому що ПК – це математичний пристрій для обробки даних, тому постановка задач повинна виражатися в математичній формі.

Опис властивостей об'єктів за допомогою математичних формул і рівнянь називається побудовою математичної моделі цього об'єкта.

Для запису математичних постановок задач використовується така форма:

- Задача.           Формулювання зміст  
Дано:           Початкові дані (аргументи)  
Знайти:       Результати  
Рішення:      Залежність між аргументами і результатами.  
Метод:        Визначити метод розв'язку (може бути відсутнім)  
При            Умова допустимості аргументів

Приклади математичних моделей розглянути по таблицях.

Задача №1. Знайти суму скінченої послідовності заданих чисел.  
(Метод: рекурентне сумування).

#### 2) алгоритм

Термін *алгоритм* існував задовго до появи комп'ютера. Він походить від імені давнього філософа і математика з Хорезму, що жив у IX столітті – Аль – Хорезмі. Саме він у своїх трактатах описав правила (алгоритми) додавання, віднімання, множення та ділення багатозначних чисел, якими користуємося сьогодні.

**Алгоритм** – це правило, що задає послідовність команд, які потрібно виконати над вхідними даними для отримання результату.

Вчинки людей підпорядковані досягненню конкретної мети. Часто люди не замислюються над тим, що їхні дії є прикладами виконання алгоритмів. Обдумуючи плани на день, ми складаємо алгоритми

розв'язування побутових задач чи задач, пов'язаних з професійною діяльністю. Отже, ми є виконавцями алгоритмів.

Завдання. Навести приклади алгоритмів з побуту.

Для полегшення фізичної та інтелектуальної праці люди створили різні технічні пристрої, машини, автомати, роботи, ПК.

**Виконавець алгоритму** – це деяка абстрактна чи реальна система, яка здібна виконувати дії, які задані алгоритмом.

Сукупність команд, які можуть бути виконані виконавцем, називається **системою команд виконавця (допустимі команди)**.

Команди, які не може виконати виконавець, називаються **недопустимими командами виконавця**.

*Розглянути слідуючи виконавців: людина, робот, ПК.*

Завдання. Навести приклади допустимих та недопустимих команд виконавців: а) людини; б) робота; в) деякого механічного пристрою.

На практиці найбільш поширені такі форми уявлення алгоритмів: словесна, формульна, графічна, на алгоритмічній мові

*Для пояснення властивостей застосувати вправу «Броунівський рух» та роботу в малих групах.* В кінці роботи визначити основні властивості алгоритму.

Основні властивості алгоритму:

- |                      |                                  |
|----------------------|----------------------------------|
| 1. Визначеність.     | 1. Правильність                  |
| 2. Скінченність.     | 2. Однозначність (формальність). |
| 3. Результативність. | 3. Масовість.                    |

#### IV.

Задачі №1. Скласти математичні моделі задач.

а) розрахувати кількість шпалер для обклейки стін;

б) розрахувати кількість фарби для фарбування полу.

Завдання №2. Скласти математичну модель знаходження площі трикутника за заданими сторонами.

V. 1) підсумки та виставлення оцінок

-Що являє собою математична модель?

-Навіщо потрібна математична постановка задачі?

-Чи завжди виникає необхідність в побудові математичної моделі?

- Що виражає розділ "зв'язок" у математичній постановці прикладної задачі?

-З яких міркувань вибирають обмеження на вихідні дані?

-Що таке алгоритм?

-Хто чи що може бути виконавцем алгоритму?

-Що відрізняє ПК та людину як виконавців алгоритму?

2) Домашнє завдання: конспект, Глава 9, § 3 – 6 (відповіді на питання).

### Урок 3

**Тема уроку:** Базові алгоритмічні структури. Види алгоритмів.

**Тип уроку:** урок засвоєння нових знань.

**Мета уроку:** – закріпити поняття алгоритму, виконавця;

- засвоїти поняття блоку, блок – схеми;

- засвоїти базові алгоритмічні структури та види алгоритмів;

- сформувати навички складання блок-схем для розв'язку задач;

- розвивати алгоритмічне мислення, алгоритмічну мову, пам'ять;

- формувати ініціативність, наполегливість;

- виховувати потребу та вміння працювати в колективі.

**Додаткова підготовка:** карки для інтерактивної вправи, питання для диктанту.

**Дидактичні засоби:** – робочий зошит учня «Основи інформатики та обчислювальної техніки 10–11 клас» Ч 2.– Київ, «Просвіта» – 1997.

**Структура уроку**

- I. Оргмомент. Актуалізація опорних знань (10 хв.)
- II. Мотивація навчальної діяльності школярів (1 хв.)
- III. Повідомлення теми, мети та задач уроку (1 хв.)
- IV. Вивчення теоретичного матеріалу та його осмислення учнями (15)
- V. Закріплення учнями матеріалу (15 хв.)
- VI. Підведення підсумків уроку та домашнє завдання (2 хв.)

**Зміст уроку**

- I. 1) Оргмомент
- 2) Диктант.

1 варіант	2 варіант
1. Що таке модель?	1. Що таке моделювання?
2. Що називається інформаційною моделлю?	2. Що називається інформаційною моделлю?
3. Навести приклади матеріальних моделей.	3. Навести приклади абстрактних моделей.
4. Основні етапи розв'язку задач за допомогою ПК.	4. Основні етапи розв'язку задач за допомогою ПК.
5. Навіщо потрібна математична постановка задачі?	5. Навіщо потрібна постановка задачі?
6. Поняття алгоритму.	6. Поняття програми.
7. Які етапи розв'язку задач за допомогою ПК виконує людина?	7. На якому етапі розв'язку задач за допомогою ПК використовується ПК?
8. Виконавець алгоритму.	8. Система команд виконавця.
9. Скласти математичну модель: обчислити площу та периметр прямокутного трикутника за гіпотенузою та гострим кутом.	9. Скласти математичну модель: обчислити площу та периметр прямокутного трикутника за гіпотенузою та катетом.

II. Які існують форми уявлення алгоритмів. Наведіть приклади різних способів. Що таке графічний засіб запису алгоритмів?

III. Записати в зошитах тему уроку. Сформулювати мету та задачі уроку.

IV. Графічний спосіб уявлення алгоритмів є більш компактним та наочним в порівнянні з словесним чи формульним.

При графічному уявленні алгоритм зображається у вигляді послідовності зв'язаних між собою функціональних блоків, кожний з яких відповідає одному чи декільком дій. Таке уявлення називається **блок-схемою**.

Під **блоком** розуміється кінцевий етап обчислювального процесу, визначений в даній схемі, як ціле.

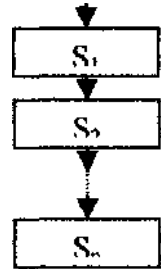
Кожному блоку відповідає геометрична фігура. Блоки з'єднуються **лініями переходу**, які визначають чергу виконання дій. Існує кілька типів блоків:



Алгоритм можна уявити як деякі структури, які складаються з окремих базових елементів. Будь - який алгоритм можна уявити комбінацією трьох базових структур: прямування, розвилка та повторення. Характерною особливістю базових структур є наявність в них одного входу та одного виходу. *Робота з робочими зошитами.*

**1. Базова структура прямування (слідування).** Подається у вигляді послідовності двох (або більше) виконуваних одна за одною простих команд  $S_1, S_2, \dots S_n$ .

**Слідування** - це така форма організації дій, при якій всі дії виконуються одна за однією по черзі.

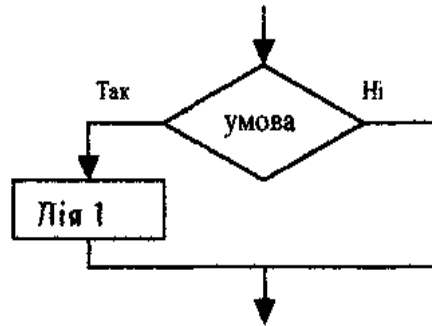
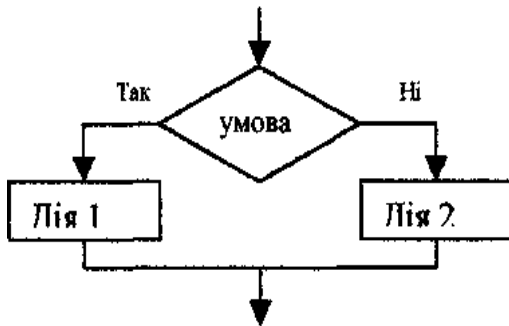


**2. Базова структура розвилка (розгалужування)** - це аналіз певної умови та вибір подальшого напрямку в послідовності виконання дій залежно від значення цієї умови.

**Розвилка** - це така форма організації дій, при якій в залежності від виконання чи невиконання деякої умови виконується чи одна, чи інша послідовність дій.

Повна розвилка

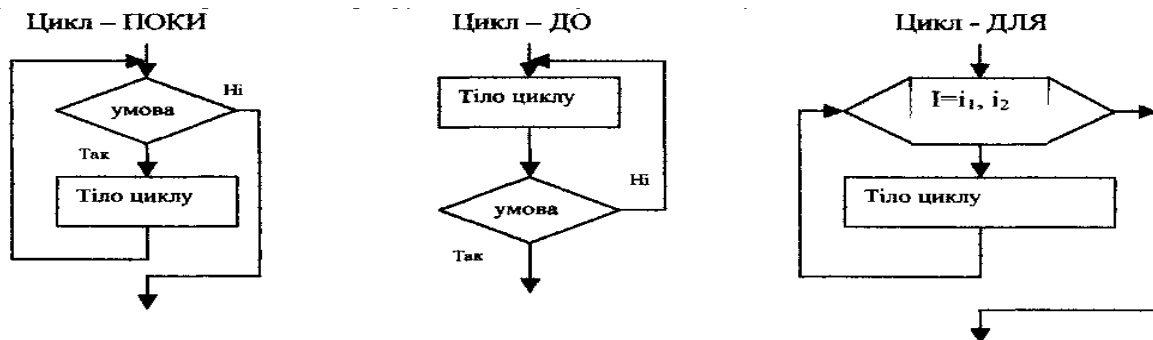
Неповна розвилка



**3. Базова структура повторення** - забезпечує багатократне виконання деякої сукупності дій, яка називається тілом циклу.



**Циклом** – називається така форма організації дій, при якій одна й та ж послідовність дій виконується декілька разів (чи ні разу), доки виконується деяка умова.



Блок – схема алгоритму, подана базовими структурами, називається структурованою.

**Лінійний алгоритм** – це алгоритм, в якому використовується тільки структура прямування.

**Алгоритм з розгалуженням** – це алгоритм, в основі якого лежить структура розвилка.

**Циклічний алгоритм** – це алгоритм, в основі якого лежить структура повторення.

#### V. 1) Інтерактивна вправа

Учитель пропонує учням об'єднатися у малі групи по 4 чол. та сісти в коло. Учні розподіляють між собою ролі: спікер (голова), секретар, посередник, доповідач. Крім основних груп, учитель утворює групу експертів.

Всі групи (і робочі, і група експертів) отримують однакові картки з прикладами різноманітних алгоритмів. Серед прикладів наведено алгоритми з різними базовими структурами і у різних формах представлення.

Завдання для груп:

- 1) визначити ті алгоритми, в основі яких структура слідування;
- 2) визначити ті алгоритми, в основі яких структура розгалуження;
- 3) визначити ті алгоритми, в основі яких структура повторення;
- 4) визначити ті алгоритми, які подані у словесній формі;
- 5) визначити ті алгоритми, які подані у графічній формі.

Використовуючи метод "Коло ідей", спікер групи експертів збирає від доповідачів результати роботи груп.

Група експертів аналізує результати роботи груп. Доповідач групи експертів повідомляє, як групи впоралися із завданням.

Приклади алгоритмів:

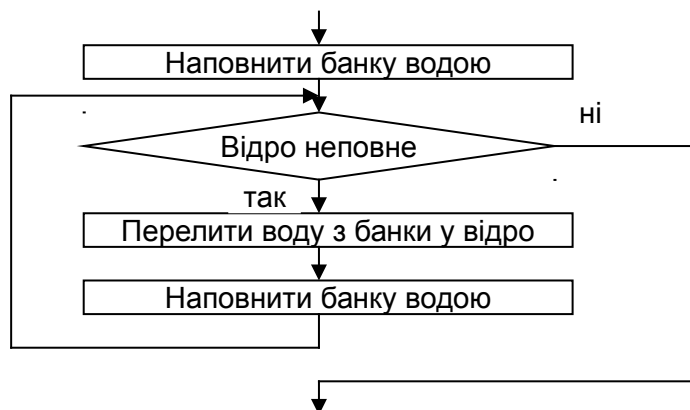
#### Приклад 1.

Алгоритм Ранок

- Встати о 7 годині.
- Виконати гімнастику.
- Умитися.
- Поснідати.
- Вийти з дому в 7-45.

#### Приклад 2.

Алгоритм наповнення.



Приклад 3. Алгоритм виконання домашнього завдання з математики.

- Поки не розв'язані всі задачі:
- Розв'язати чергову задачу.
- Кінець циклу.
- Піти гуляти до вечері.

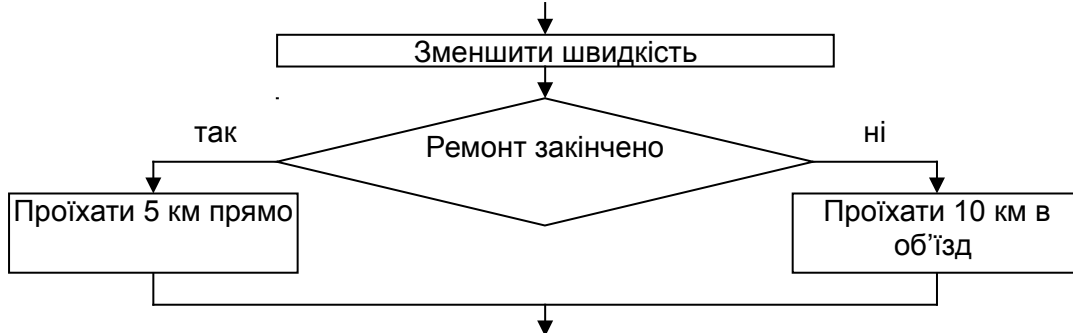
Приклад 4. Алгоритм відкриття дверей.



Приклад 5. Покупка квитків на сеанс у кіно.

- Підійти до каси.
- Якщо білети на сеанс 12.00 є, то купити квитки.
- Відійти від каси.

Приклад 6. Алгоритм проїзду на АЗС, якщо на дорозі був ремонт.



2) Перевести алгоритму у блок - схему та формально їх виконати.

Підручник Гейн. А.Г. стор. 53 № 2 а); стор. 73 № 2а.

№2а) стор.53. Які значення приймуть змінні  $a$  та  $b$  після виконання наступного алгоритму? Вкажіть дії, які не будуть виконуватися при виконанні алгоритму.

- Присвоїти  $a$  значення 12.
- Присвоїти  $b$  значення 13.
- Якщо  $a < b + 1$ , то:
  - Присвоїти  $b$  значення  $b - a$ .
  - Присвоїти  $a$  значення  $a - b$ .
- Інакше:
  - Присвоїти  $a$  значення  $a - b$ .
  - Присвоїти  $b$  значення  $b - a$ .
- Кінець розвилки.
- Присвоїти  $a$  значення  $ab$ .
- Якщо  $a > b^2$ , то:
  - Присвоїти  $b$  значення  $a - b$ .
- Кінець розвилки.

№2а) стор.73. Чому будуть дорівнювати  $a$  та  $b$  після виконання алгоритму? Скільки разів буде виконано тіло циклу?

- Присвоїти  $a$  значення 25.
- Присвоїти  $b$  значення 1.
- Поки  $a - b > a/b$ , повторювати:
  - Присвоїти  $a$  значення  $a + 10$ .
  - Присвоїти  $b$  значення  $-3b$ .
- Кінець циклу.

VI. 1) Підсумки:

- Учитель пропонує учням відповісти на запитання. Запитання для бесіди:

1. Чи досягли мету, що поставили перед собою на початку уроку? Завдяки чому? Що допомогло у засвоєнні матеріалу уроку?



2. Що було легко, а що складно?
3. Що найбільше запам'яталось на сьогоднішньому уроці?
4. Які ви знаєте базові структури алгоритмів?
5. Які ви засвоїли форми подання алгоритмів?
6. Від чого залежить форма подання алгоритмів?
7. Що такс конструювання алгоритмів?
8. Хто визначає, якою саме базовою структурою треба скористатися в тому чи іншому алгоритмі?

2) Домашнє завдання.

Зарецька Глава 10 § 1 пп 1.2, 1.3

Придумати будь-який алгоритм на побутову тему (кулінарний, прибирання кімнати, виконання уроків і т. д.), виділивши в ньому базову структуру.

Скласти алгоритм РАНОК, ВЕЧІР, ШКОЛА.

## Урок 4

**Тема уроку:** Розв'язок задач.

**Тип уроку:** урок використання вмінь та навичок

**Мета уроку:** - закріпити навичками та вміннями записувати та виконувати алгоритми;

- розвивати алгоритмічне мислення;
- розвивати творчий потенціал на основі створення алгоритмів
- виховувати об'єктивне відношення до даних комп'ютерного обчислення;
- виховувати обережне відношення до техніки та інформації;
- формувати наполегливість, ініціативність.

**Додаткова підготовка:** картки з завданням

**Дидактичні матеріали:** Test-W - Алгоритми - Алгоритми 10 клас

### Хід уроку.

- I. Оргмомент. Повідомлення теми, мети та задач уроку (2 хв.)
- II. Актуалізація знань (12 хв.)
- III. Осмислення змісту та послідовність використання практичних дій (15 хв.)
- IV. Самостійне виконання учнями завдань під контролем та за допомогою вчителя (10 хв.)
- V. Звіт учнів про способи та результати виконання роботи. Підсумки уроку та домашнє завдання (6 хв.)

### Зміст уроку.

- I. Оргмомент. Повідомлення теми, мети та задач уроку.
- II. Актуалізація опорних знань проводиться в 2 етапи.  
Робота по карткам (5 варіантів).

Дати означення структури, накреслити блок-схему, описати дію виконання, особливості застосування

III.

1) Задачі (розв'язувати ланцюжком). Записувати алгоритми у вигляді блок-схеми.

№1. Скласти алгоритм додавання двох дробів:

- а) привести дробу до спільного знаменника;
- б) задати перший дріб;
- в) записати відповідь;
- г) знайти спільний знаменник дробів;
- д) задати другий дріб;
- е) додати чисельники дробів, знаменник без змін.

№ 2. Скласти алгоритм знаходження більшого з двох чисел:

- а) задати друге число;
- б) в іншому випадку;
- в) якщо перше число менше другого числа;
- г) більше - число перше;
- д) то;
- е) задати перше число;
- ж) записати відповідь;
- з) більше - число друге.

№ 3. Скласти алгоритм фарбування паркану:

- а) взяти фарбу, пензлик;
- б) відійти від паркану;
- в) зробити крок вправо на ширину дошки;
- г) підійти до лівого краю паркану;
- д) пофарбувати одну дошку;
- е) поставити фарбу та покласти пензлик;
- ж) поки паркан не закінчився;

№ 4. Виконати алгоритм при  $Y=5$  та  $A=5$ :

**IV.** Самостійне виконання учнями завдань під контролем та за допомогою вчителя

№ 5 - Виконати послідовність команд:

- 1) Вважати, що  $A=5$ ;
- 2) Зменшити  $A$  на 2;
- 3) Змінити знак  $A$  на протилежний;
- 4) Збільшити  $A$  в 3 рази;
- 5) Повідомити значення  $A$

- Якого значення набуде  $A$ , якщо змінити перший рядок на Вважати  $A$  дорівнює 7?

№ 6. Скласти блок-схему даного алгоритму.

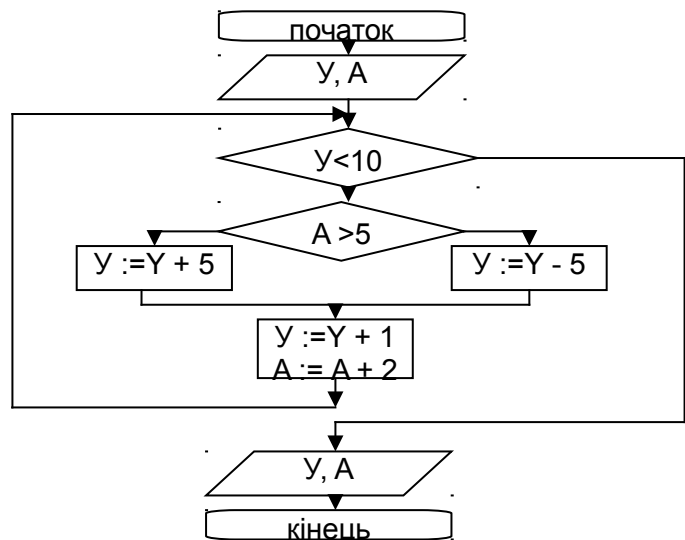
Виконати алгоритм для значень  $A=3, 4, 5$ .

- Початок
- Ввести  $A$
- $A := A - 2$
- Поки  $A \geq 1$ , повторювати:
  - $A := A - 2$
- Вивести  $A$
- Кінець

№ 7. Скласти блок-схему даного алгоритму.

Виконати алгоритм для значень  $A=3, 4, 5$ .

- Початок
- Ввести  $A$
- Якщо  $A > 3$ 
  - то  $A := A - 3$
  - інакше  $A := 0$



- $A := A - 2$
- Вивести  $A$

V. 1) Звіти учнів про виконання роботи

№5  $A = 9; 15.$  № 6.  $A = 1; 0; -1.$  № 7.  $A = -2; -1; 0.$

2) Підсумки уроку у вигляді блиц - диспуту.

Учні сідають у спільне коло. Для підбиття підсумків учитель пропонує їм взяти участь у блиц-диспуті. Для його проведення учням треба роздати по одній картці з хибним або істинними твердженнями. Вони по чергово звертаються до свого сусіда праворуч (наприклад "Прошу бути моїм опонентом Миколу") та оголошують своє твердження з картки. "Опонент" повинен висловитися щодо з приводу істинності твердження: "Так, дійсно...." або "Ні, це неправильно,...". Свою думку "опонент" висловлює за методом ПРЕС: "Я вважаю, що.... ; тому що.... ; наприклад, ; отже (тому), я вважаю "

### **Зміст карток**

1.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що оскільки алгоритм - це чітка послідовність інструкцій, що приводить до шуканого результату, то його може виконати будь-який виконавець.*

2.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що конструювання алгоритмів полягає у створенні ланцюжка базових конструкцій.*

3.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що до базових структур алгоритмів відносять такі конструкції: дискретний алгоритм, масовий алгоритм, формальний алгоритм.*

4.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що особливістю базових структур є те, що кожна з них має один вхід і один вихід.*

5.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що алгоритм можна зобразити тільки графічно у вигляді деякої схеми.*

6.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що набір команд, з яких складається. алгоритм, можна виконувати у будь-якій послідовності.*

7. \_\_\_\_\_ Прошу бути моїм опонентом

*Я стверджую, що певний набір команд, що їх може виконати виконавець, називається комплексом команд виконавця.*

8. \_\_\_\_\_ Прошу бути моїм опонентом

*Я стверджую, що коли виконавець виконує алгоритм, він спочатку творчо опрацьовує команду, а потім її виконує.*

9. \_\_\_\_\_ Прошу бути моїм опонентом

*Я стверджую, що алгоритм повинен застосовуватися до будь-яких допустимих наборів вхідних даних.*

10.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що під час конструювання алгоритму вихід кожної базової структури приєднується до входу іншої. Таким чином, весь алгоритм подається у вигляді лінійної послідовності базових алгоритмічних структур.*

11.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що алгоритми, в яких використовується лише структура "слідування", називаються лінійними.*

12.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що алгоритми, в основі яких лежить структура розгалуження, називають циклічними алгоритмами.*

13.Прошу бути моїм опонентом \_\_\_\_\_

*Я стверджую, що в будь-якому алгоритмі ми діємо враховуючи умови.*

3) Домашнє завдання: повт. за підручником Глава 9 § 6, глава 10.

№1. Побудувати математичну модель та скласти алгоритм переходу вулиці при наявності світлофору.

№ 2. Побудувати математичну модель та скласти алгоритм наповнення ванни відрами води.

## Урок 5

**Тема уроку:** Тематична атестація з теми "Основи алгоритмізації"

**Тип уроку:** урок перевірки знань, вмінь та навичок.

**Мета уроку:** - перевірити рівень засвоєння поняття алгоритму, блок - схеми;

- перевірити рівень засвоєння базових алгоритмічних структур;
- перевірити вміння складати блок - схеми алгоритмів, користуючись базовими алгоритмічними структурами;
- перевірити навички покрокового виконання алгоритмів;
- розвивати аналітичне та алгоритмічне мислення, наполегливість, пам'ять, вміння аналізувати;
- виховувати увагу та кмітливість учнів, вміння оцінювати свої сили та виховувати силу волі.

**Додаткова підготовка:** Test-W - Алгоритми - Алгоритми 10 клас, підготувати завдання для ТО.

**Дидактичні засоби:** підготувати тест «Алгоритми», завдання для ТО.

### Структура уроку.

- I. Оргмомент, Постанова теми, мети та задач уроку (1 хв.)
- II. Інструктаж по проведенню ТО. (2 хв.)
- III. Виконання ТО (40 хв.)
- IV. Підсумки та домашнє завдання (2 хв.)

### Зміст уроку.

II. 1) Тест на ПК: Test-W - Алгоритми-Алгоритми 10 клас розраховано на 10 хв.

1. Алгоритм є

- а) предметною інформаційною моделлю
- б) статистичною інформаційною моделлю
- в) динамічною інформаційною моделлю**
- г) табличною інформаційною моделлю
- д) програмою

2. Опис алгоритмів проводять

- а) граф-схемою
- б) словесно-формульним способом**
- в) малюнком
- г) блок-схемою**
- д) алгоритмічною мовою**

3. Циклічний алгоритм - це алгоритм в якому

- а) виконується деяка серія команд
- б) команди виконуються послідовно в порядку їх запису
- в) виконується група команд в залежності від умови
- г) багатократно виконуються деякі команди**
- д) в основі лежить структура повторення**

4. Співпадіння отриманих при виконанні алгоритму результатів незалежно від користувача чи технічних засобів визначається як

- а) формальність**
- б) дискретність
- в) визначеність**
- г) результативність
- д) правильність

5. Застосування алгоритму до цілого класу однотипних задач визначається як

- а) результативність
- б) масовість**
- в) однозначність
- г) дискретність
- д) формальність

6. Дані, значення яких не змінюється в процесі виконання алгоритму, називаються
- а) вхідними даними      **б) константами**
  - в) результатами      г) змінними      д) аргументами
7. Розгалужений алгоритм - це алгоритм, в якому:
- а) відбувається виконання деякої серії команд
  - б) команди виконуються послідовно в порядку їх запису
  - в) в залежності від умови виконується та чи інша серія команд**
  - г) відбувається виконання деякої серії команд скінченне число раз
  - д) в основі лежить структура розвилки**
8. Лінійним прийнято називати обчислювальний процес, у якому команди виконуються:
- а) в залежності від початкових даних
  - б) багато разів у порядку їх запису
  - в) послідовно, у порядку їх запису**
  - г) в залежності від кінцевих результатів
  - д) в залежності від умов
9. До основних способів опису алгоритмів належать:
- а) словесний**      б) лінійний
  - в) графічний**      г) логічний      **д) алгоритмічний**
10. Структура розвилки лежить в основі:
- а) циклічних алгоритмів      б) лінійних алгоритмів
  - в) допоміжних алгоритмів      **г) розгалужених алгоритмів**
  - д) комбінованих алгоритмів**
11. Основними базовими структурами для побудови алгоритму є:
- а) закінчена      **б) циклічна**
  - в) одноманітна      **г) розгалужена**      **д) лінійна**
12. Чітка послідовність дій, яка приведе до бажаного результату називається
- а) виконавцем      **б) алгоритмом**
  - в) системою команд виконавця
  - г) командою      д) властивість алгоритму
13. Який із об'єктів може бути виконавцем:
- а) ножиці      б) карта
  - в) принтер**      г) книга      **д) комп'ютер**
14. Яке буде значення змінної X після виконання операції присвоювання:  $X := 5, X := X + 1$
- а) 5      **б) 6**      в) 1      г) 10      д) 4
15. Який із документів є алгоритмом?
- а) правила техніки безпеки      в) розклад уроків
  - б) інструкція з отримання грошей у автоматі**
  - г) список класу      **д) рецепт приготування страви**
16. Що змінює операція присвоювання ( $:=$ )
- а) значення змінної**      б) ім'я змінної
  - в) тип змінної      г) тип алгоритму      д) перевіряє умову
17. Властивістю алгоритму є:
- а) результативність**      б) циклічність      **д) масовість**
  - в) можливість зміни послідовності виконання команд
  - г) можливість виконання алгоритму в зворотному порядку
- 2) Письмова робота - учні вибирають завдання за власним вибором. Оцінка за ТО це середнє арифметичне з тесту та письмової роботи.

#### 1 варіант.

1. (1б). Навести приклади алгоритмів, які зустрічаються у шкільному житті.
2. (1 б). Виконати послідовність вказівок.:
  - 1) Вважати, що A дорівнює 5,
  - 2) Зменшити A на 2:

- 3) Змінити знак A на протилежний;
- 4) Збільшити A в 3 рази;
- 5) Повідомити значення A.

Якого значення набуде A, якщо змінити перший рядок на "Вважати A дорівнює 7"?

3. (2б). Запишіть блок - схему підрахунку площі прямокутника за даними його сторонами.
4. (2б). Виконати алгоритм для значень  $A = 3, 4, 5$ .
  - Присвоїти A значення  $A-2$ ;
  - Поки  $A > 1$ . повторювати
    - Присвоїти A значення  $A - 3$
  - Кінець циклу.
5. (2б). Пристрій має дві клавіші A і B. При натисканні на A число збільшується на одиницю, на B - збільшується вдвічі. Написати натискання клавіш A і B, щоб з числа 4 отримати число і 9
5. (3б). Скласти блок - схему алгоритму переходу вулиці при наявності світлофору.
- 7 (4б). Записати алгоритм у вигляді блок - схеми Визначити, чому дорівнює S та I після виконання алгоритму? Скільки разів буде виконуватись тіло циклу?
  - Присвоїти S значення -5
  - Присвоїти I значення 2
  - Поки  $I < 4$ , повторювати:
    - Якщо  $S < -3$ , то:
      - Присвоїти S значення  $S+I$
      - Присвоїти I значення  $I \cdot N$ .
    - Інакше
      - Присвоїти S значення  $S+2$
      - Присвоїти I значення  $I+S$ .
  - Кінець розгалужування.
  - Кінець циклу.
8. (4б). Скласти алгоритм, який обчислює суму найменшого числа та середнього за величиною числа з трьох чисел A, B, C.

## 2 варіант.

1. (1б). Навести приклади алгоритмів, які зустрічаються у Вас вдома.
2. (1б). Виконати послідовність вказівок:
  - 1) Вважати, що A дорівнює 8;
  - 2) Зменшити A в 2 рази;
  - 3) Змінити знак A на протилежний;
  - 4) Збільшити A на 4;
  - 5) Повідомити значення A.

Якого значення набуде A, якщо змінити перший рядок на "Вважати A дорівнює 6"?
3. (2б). Запишіть блок - схему підрахунку периметру прямокутника за даними його сторонами.
4. (2б). Виконати алгоритм для значень  $A = 3, 4, 5$ .
  - Якщо  $A > 3$ , то
    - Присвоїти A значення  $A - 3$ .
  - Інакше присвоїти A значення 0.
  - Кінець розгалужування
  - Присвоїти A значення  $A - 2$ .
5. (2б). Пристрій має дві клавіші A і B. При натисканні на A число збільшується на одиницю, на B - збільшується вдвічі. Написати натискання клавіш A і B, щоб з числа 5 отримати число 18.
6. (3б). Скласти блок - схему алгоритму наповнення ванни відрами

води.

7. (46). Записати алгоритм у вигляді блок - схеми. Визначити, чому дорівнює S та I після виконання алгоритму? Скільки разів буде виконуватись тіло циклу?

- Присвоїти S значення - 4
- Присвоїти I значення -2
- Поки  $S < I$  повторювати:
  - o Якщо  $I < -2$ , то:
    - Присвоїти S значення S - 1
    - Присвоїти I значення I+2.
  - o Інакше Присвоїти S значення S+2
    - Присвоїти I значення I - S.
  - o Кінець розгалужування.
- Кінець циклу.

8. (46). Скласти алгоритм, який обчислює різницю найбільшого числа та середнього за величиною числа з трьох чисел M, N, P.

IV. Збір робіт. Підсумкова оцінка - кількість правильно виконаних завдань, з врахуванням вказаних балів для кожного завдання.

Домашнє завдання: повторити основні етапи розв'язку задач за допомогою ЕОМ. Підручник Зарецької, Глава. 9.

#### Список літератури:

1. Інформатика: Учебн. пособие для 10-11 кл. средн.общеобразоват.-школ/ И.Т.Зарецкая, Б.Г. Колодяжний. А.М. Гуржий, О.Ю.Соколов.-К.: Учебная книга,2002.-496с.:ил.
2. Глинський Я.М. Інформатика: 8-11. Навч. Посібник для загальноосвітніх навчальних закладів: У 2-х кн. - Кн.1. Алгоритмізація і програмування. 3-є видання. - Львів, 2003. - 200с.
3. Забарна А. П. Основи алгоритмізації та програмування. Інтерактивні технології навчання на уроках. - Тернопіль,2006. - 96с.

**Чередниченко Наталья Петровна** – учитель информатики специализированной школы с углубленным изучением некоторых предметов № 4

## 6.2. Основы програмування. (10 годин)

№ урока	Тема урока	
1	Понятие программа. Язык программирования Turbo Pascal	
2	Основные понятия языка Turbo Pascal.	
3	Структура программы на языке Turbo Pascal. Инструкции присваивания, ввода и вывода.	
4	Практическая работа «Создание и реализация простейших линейных программ».	
5	Организация разветвляющихся процессов	
6	Цикл с параметром	
7	Оператор цикла с "предусловием" и "постусловием"	
8	Вложенные циклы.	
9	Одномерные массивы. Работа с элементами. Двумерные массивы. Работа с элементами.	
10	Двумерные массивы. Работа с элементами.	

### Урок 1

**Тема урока:** Понятие программа. Язык программирования Turbo Pascal

**Цель урока:** дать основные сведения о программе и языках программирования, ознакомить учащихся со средой программирования Turbo Pascal.

#### Ход урока.

1) Изучение нового материала. (Учитель рассказывает о языках программирования, ученики записывают основные понятия.)

При разработке компьютерной модели после составления алгоритма следует написать программу. Для этого используется, какой либо язык программирования. Прежде чем перейти к составлению программ, мы изучаем общие понятия, связанные с программированием

☛ **Программа** – это алгоритм, записанный на языке программирования.

История программ начинается с далеких 20-х годов 19 века, когда английский исследователь Ч.Бэббидж выдвинул идею о предварительной записи действий вычислительной машины.

Первые программы предназначались для управления ткацкими станками. Они записывались на перфокарты, которые были придуманы Чарлзом Бэббиджем и французом-изобретателем Жозефом Жаккаром. С этих первых программ началась история программирования. Вместе с Бэббиджем работала его современница Ада Лавлейс (дочь великого английского поэта Джорджа Байрона) которую называют первым в мире программистом. Она ввела в науку о программировании термины и понятия, которые применяются и по сей день. К таким понятиям относятся, например – цикл – важнейшая структура любого современного языка программирования. В честь первой программистки был назван язык программирования «Ада», разработанный уже в наше время. В 20 веке появились цифровые компьютеры. Первые программы, которые составлялись для них, записывались в машинных кодах.

☛ **Машинные коды** – набор двоичных кодов для работы центрального процессора конкретного типа.



Программист, чтобы написать правильную программу, должен был в деталях представлять себе работу вычислительной машины. Это сильно затрудняло общение человека с компьютером, поэтому программисты стали серьезно задумываться над кодированием программ. Они пришли к выводу, что программа должна составляться на языке, более доступном человеку, чем язык машинных кодов. Возникла необходимость в создании языков программирования.

☛ **Язык программирования** – это искусственный язык для написания команд, выполняемых компьютером. Язык программирования состоит из фиксированного словаря и совокупности правил (синтаксиса) написания команд.

Т.к. язык программирования не понятен для компьютера, то должна существовать специальная программа, которая переводила бы символы этого языка в двоичные символы машинных команд. Эта программа была создана в начале 50-х годов 20 века американской программисткой и одновременно контр-адмиралом морских сил США Грейс Хоппер, и получила название: транслятор (от англ. translation-перевод) – это программа, которая преобразует команды языка программирования в машинный код.

С изобретением трансляторов роль машинных команд в программировании стало резко убывать. Первые трансляторы предназначались для языков Ассемблера (языков низкого уровня, т.е. языков близких числовому коду процессора).

☛ **Язык Ассемблера** – это язык, который переводит понятные человеку символы мнемонически в машинные символы.

В 50-60-е годы начали создаваться языки программирования высокого уровня (язык, состоящий из операторов, похожих на обычные слова).

Например: Basic, Pascal, C++, Delphi, Fortran и т.д.

Итак, для перевода текста программы на язык, понятный компьютеру, должна существовать отдельная программная оболочка – *транслятор*. Трансляторы бывают двух видов:

- ☛ 1) **Интерпретатор** – преобразует небольшой фрагмент исходной программы в машинные коды и, лишь дождавшись, когда процессор их выполнит, переходит к обработке следующего фрагмента.
- ☛ 2) **Компилятор** – транслирует сразу всю программу в машинные коды и помещает их в память компьютера, не выполняя. Откомпилированную программу можно сохранить. Компиляторы во время своей работы занимают меньшую часть ресурсов компьютера.

Современные языки программирования – это не только язык программирования, снабженный компилятором. Языки программирования как программный продукт поставляются на рынок ПО вместе с полным комплектом инструментов для создания программ. Такой комплект называется интегрированной средой программирования.

☛ **Интегрированная среда программирования** – это система программирования, совмещающая редактор для удобного ввода и редактирования программы, транслятор и отладчик ошибок. Примеры: Visual Basic, Delphi и т.д.

Мы будем изучать интегрированный язык программирования Turbo Pascal.

Язык Pascal был разработан в 1970 году Николаусом Виртом – профессором информатики Федерального технического университета в Швейцарии и назван в честь французского философа и математика 17 века Блеза Паскаля. Паскаль создавался как язык для обучения студентов программированию, но позволяет эффективно решать самые раз-

нообразные задачи. При создании этого языка Вирт большое внимание уделял структурному программированию, благодаря которому конструкции языка Паскаль позволяют создавать, надежные, легкопроверяемые программы с ясной и четкой структурой.

Позднее была создана чрезвычайно удобная интегрированная среда. Turbo Pascal, включающая редактор, компилятор, редактор связей и отладчик.

После загрузки среды Turbo Pascal вы увидите на экране окно Turbo Pascal, состоящая из (учитель показывает на компьютере):

- ☛ 1) **Меню** (строка, включающая опции File (управление файлами); Edit (редактирование текста программ), Search (поиск и замена текста), Run (выполнение программы в различных режимах), Compile (компиляция, создание exe - файла), Delete (отладка программы), Tools (отладочные команды сообщения), Options (настройка среды программирования), Windows (управление окнами), Help (доступ к встроенной справочной системе)).
- ☛ 2) **Рабочая область**, занимающая центральную часть экрана и предназначенная для работы с текстом программ.
- ☛ 3) **Строка статуса**, расположена в нижней части окна. В ней отображается также в виде напоминания функциональные клавиши и комбинации клавиш, допустимых в данный момент в активном окне (F1-Help, F2 -Save).

Кроме этого имеются вертикальные и горизонтальные полосы прокрутки, полоса заголовка окна программы (MONAME 00.pas), номер окна, кнопка масштабирования.

Команды меню в Turbo Pascal можно выбрать с помощью клавиатуры (нажать F10, а затем выбрать команду клавишами управления курсором), либо мышью (щелчок мышью по нужной опции меню).

- ☛ **Операции с фрагментами текста или файла**
- ☛ 1. **Выделение фрагмента текста** - <shift>+<клавиши перемещения курсора> или мышкой как MO Word
- ☛ 2. **Перемещение фрагмента текста**
  - ✓ Выделить фрагмент текста
  - ✓ Выбрать в опции Edit команду Cut (вырезать)
  - ✓ Вставить курсор в нужное место программы.
  - ✓ Выбрать в опции Edit команду Paste (вставить)
- ☛ 3. **Копирование фрагмента текста**
  - ✓ Выделить фрагмент текста
  - ✓ Выбрать в опции Edit команду Copy (копировать)
  - ✓ Установить курсор в нужное место
  - ✓ Выбрать в опции Edit команду Paste (вставить)
- ☛ 4. **Удаление фрагмента текста**
  - ✓ Выделить фрагмент
  - ✓ Выбрать в опции Edit команду Clear (очистить)
  - ✓ Если удалили не тот фрагмент, то выбрав в опции Edit команду Undo (отменить), можно отменить удаление.
- ☛ 5. **Сохранение программы (файла)**
  - ✓ Выбрать в опции File команду Save
  - ✓ В диалоговом окне Save File As ввести имя файла с расширением .pas и нажать ОК.

- ☛ 6. Компиляция и запуск программы - <Ctrl>+<F9> или в опции Run выбрать команду Run
- ☛ 7. Открытие окна пользователя - <Alt>+<F5>
- ☛ 8. Выход из среды программирования - <Alt>+<X>

2) Практическая работа «Работа в среде программирования Turbo Pascal» .

#### Практическая работа

«Работа в среде программирования TURBO PASCAL»

**Цель работы:** ознакомиться с основными частями главного окна TURBO PASCAL, отработать навыки работы с командами главного меню, научиться работать с фрагментами текста программы, усвоить компиляцию программы и запуск программы.

#### Ход работы.

1. Произведите запуск TURBO PASCAL.
2. Найдите основные части главного окна TURBO PASCAL.
  - Строка главного меню, состоящая из опций File (управление файлами), Edit (редактирование текста программы), Search (поиск и замена текста), Run (выполнение программы в различных режимах), Compile (компиляция программы, создание exe - файла), Debug (отладка программы), Tools (отладочные команды сообщения), Options (настройка среды программирования), Window (управление окнами), Help (доступ к встроенной справочной системе).
  - Рабочая область, занимающая центральную часть экрана и предназначенная для работы с текстом программы.
  - Строка статуса, расположенная в нижней части окна (F1 - Help, F2 - Save и т.д.) - строка подсказок.
3. Отработайте навыки работы с командами главного меню. Чтобы выбрать нужную команду при помощи клавиатуры, надо:
  - Нажать клавишу F10 ( в результате один из пунктов меню будет выделен цветом) .
  - Используя клавиши перемещения курсора влево и вправо, выделить название нужного пункта меню и нажать клавишу <Enter> (в результате этого появится список команд выбранного пункта меню) .
  - Используя клавиши перемещения курсора вверх и вниз, выделить название нужной команды и нажать <Enter>.
  - Для завершения диалога без выполнения команды, следует выбрать кнопку Cancel (отмена) или клавишу < Esc>.
4. Наберите, предложенную ниже программу, вставляя вместо многоточия свое имя.
 

```

program Hallow;
begin
    Writeln ('Привет,....!');
    Writeln ('Вас приветствует TURBO PASCAL!');
end.
```
5. Отработайте навыки работы с фрагментами текста.
  - a) Выделение фрагмента текста:
    - Используя клавиши перемещения курсора, установить курсор на символ, с которого начинается фрагмент текста;
    - Нажать клавишу <Shift>, удерживая ее и используя клавиши перемещения курсора, поместить курсор на последний символ фрагмента.

- Отпустить клавишу <Shift>.

Выделите в набранной вами программе слова `Writeln ('Привет, ...!');`

б) Перемещение фрагмента текста:

- Выделить фрагмент текста.
- Из меню `Edit` выбрать команду `Cut` (Вырезать). В результате этого выделенный фрагмент текста будет удален с экрана и помещен во внутренний буфер редактора.
- Установить текстовый (мигающий) курсор за тем символом текста, после которого нужно поместить вырезанный фрагмент текста. Это можно сделать, перемещая курсор с помощью клавиши перемещения курсора.
- Из меню `Edit` выбрать команду `Paste` (Вставить).

Используя команды перемещения фрагмента текста, преобразуйте набранную программу к виду:

```
program Hallow;
begin
    Writeln ('Вас приветствует TURBO PASCAL!');
    Writeln ('Привет,...!');
end.
```

в) Копирование фрагмента текста:

- Выделить фрагмент текста.
- Из меню `Edit` выбрать команду `Copy` (копировать). В результате этого копия выделенного фрагмента будет помещена во внутренний буфер редактора.
- Установить текстовый курсор за тем символом текста, после которого нужно поместить копию.
- Из меню `Edit` выбрать команду `Paste` (Вставить).

Скопируйте и вставьте в текст программы дважды строку `Writeln ('Привет,...!');`

г) Удаление фрагмента текста:

- Выделить фрагмент текста.
- Из меню `Edit` выбрать команду `Clear` (очистить). В результате этого выделенный фрагмент текста будет удален с экрана.

Удалите строку программы `Writeln ('Привет,...!');`

**Примечание:** если вы случайно удалили нужный фрагмент текста или переместили его не туда куда надо, то можно привести текст к исходному виду, отменив выполненную команду. Для этого надо из меню `Edit` выбрать команду `Undo` (отменить).

б. Сохранение текста программы:

- Из меню `File` (файл) выбрать команду `Save` (сохранить). В результате этого на экране появится диалоговое окно `Save File As` (сохранить как).
- В поле `Save File As` (сохранить как) введите имя файла (с расширением `pas`) и нажать <Enter> или щелкнуть ОК.

**Примечание:** 1) В имени файла допускается использовать только буквы латинского алфавита и цифры, причем первым символом должна быть буква.

2) Количество символов имени файла не должно превышать восьми.

3) Имя файла должно быть связано с его содержанием.

7. Компиляция программы осуществляется при помощи команды `Compile` (компилировать) меню `Compile`.

8. Чтобы запустить программу на выполнение надо из меню Run (выполнить) выбрать команду Run или нажать клавиши <Ctrl> + <F9>.

*Примечание:* по завершении работы программы, вновь появляется главное окно TURBO PASCAL. Посмотреть результат работы программы можно, переключившись на окно пользователя. Для этого надо из меню Debug (отладка) выбрать команду User Screen (окно пользователя) или нажать клавиши <Alt> + <F5>. Вернуться в главное окно TURBO PASCAL можно нажатием < Esc>.

**Совет:** чтобы по завершении программы окно пользователя сразу не исчезало с экрана, добавьте в конец текста программы инструкцию `Readln`.

9. Чтобы завершить работу в TURBO PASCAL, надо из меню File (файл) выбрать команду Exit или нажать клавиши <Alt> + <X>.

3) Домашнее задание. Составить программу «Приветствия» для своего друга на языке Turbo Pascal.

## Урок 2

**Тема урока:** Основные понятия языка Turbo Pascal.

**Цель урока:** дать основные сведения о языке Turbo Pascal, типах данных, операциях и выражениях.

### Ход урока.

1) Опрос по теме «Понятие программа. Среда программирования Turbo Pascal».

2) Изучение нового материала. (Учитель каждому ученику выдает карточки по новой теме (карточки остаются у учеников), поясняя информацию)

*Раздаточный материал*

### - Основные понятия языка Turbo Pascal.

Понятие	Описание	Пример
Алфавит языка	1) Буквы латинского алфавита 2) Цифры 3) Специальные символы 4) Зарезервированные слова	A..Z; a..z 0..9; +; -; /; *; \$; { }; < > ; [ ] и т.д. Begin; end; real
Идентификаторы	Имена объектов (констант, типов данных, переменных, функций, программ). Правила записи 1) Все идентификаторы состоят из букв латинского алфавита, цифр, начинаются с буквы или знака «_»; 2) Прописные и строчные буквы не различаются; 3) Идентификаторы могут иметь различную длину, но используются только первые 63 символа	Pi Test My_Program A1; b; a
Элементы данных	1) Константа – ячейка памяти, значение которой в ходе выполнения программы остается постоянным 2) Переменная – ячейка памяти, значение которой может изменяться в ходе выполнения программы.	45; Pi  A; b; A1; x



$$\frac{x^2+2}{3x-4}$$

Например, выражение  $\frac{x^2+2}{3x-4}$  на Pascal записывается  $(Sqr(x)+2)/(3*x-4)$ .

2. В выражениях используются только круглые скобки, причем количество открывающихся скобок должно соответствовать количеству закрывающихся.

3. Нельзя записывать подряд два знака арифметических операций. Вычисления выполняются слева направо соответственно старшинству операций.

Задание: Записать выражение на языке Pascal

$$\frac{x^2-4}{2x+arctgx}$$

1)  $\frac{x^2-4}{2x+arctgx}$  Ответ:  $(SQR(x)-4)/(2*x+arctan(x))$ .

$$\frac{\sqrt{arctg^2(x+1)+x^3}}{2^{5x+4}}$$

2)  $\frac{\sqrt{arctg^2(x+1)+x^3}}{2^{5x+4}}$  Ответ:  $(SQRT(SQR(arctan(x+1)))+x*x*x)/(\text{Exp}((5*x+4)*\text{Ln}(2)))$ .

3) Самостоятельная работа

Записать выражения на языке Pascal

I Вариант	II Вариант
1) $\frac{5x-x^2}{\sqrt{x+4x^2}}$	1) $\frac{ x +2\sqrt{x}}{3(x^2+7)}$
2) $\frac{\sin x^2+2\sqrt{x}}{2^x}$	2) $\frac{3^x}{\text{Ln}\sqrt{x+5x^2}}$
3) $\frac{5x^2+x}{\sqrt{\cos^2(x+4)+ 2x }}$	3) $\frac{\sqrt{\sin^2(2x)+x^2}}{6^{5x^2-x}}$

Ответы:

I Вариант: 1)  $(5*x-x*x)/SQRT(x+4*x*x)$

2)  $(\sin(SQR(x))+2*SQRT(x))/\text{Exp}(x*\text{Ln}(2))$

3)  $(\text{Exp}((2*SQRT(x)+x)*\text{Ln}(5)))/SQRT(SQR(\cos(x+4))+\text{Abs}(2*x))$

II Вариант: 1)  $(\text{Abs}(x)+2*SQRT(x))/(3*(x*x+7))$

2)  $\text{Exp}(x*\text{Ln}(3))/(\text{Ln}(SQRT(x))+5*x*x)$

3)  $SQRT(SQR(\sin(2*x)+x*x))/\text{Exp}((5*x*x-x)*\text{Ln}(6))$

4) Домашнее задание. Написать на языке Pascal выражение 1)  $\frac{x^3+5x}{2*x^2+|x|}$ ;

$$\frac{2^{x+3}}{\sin 2x+\cos \frac{\pi}{x}}$$

2)

### Урок 3

**Тема урока:** Структура программы на языке Turbo Pascal. Инструкции присваивания, ввода и вывода.

**Цель урока:** дать представление об операторах присваивания, вывода и ввода.

Ход урока.

1) Опрос по теме «Основные понятия языка программирования Turbo Pascal».

Задание. Записать выражения на язык математических символов.

$$\text{SQR}((X) + \text{SIN}(5 * X)) / (3 * \text{SQRT}(X)) + 4$$

$$\text{ARCTAN}(\text{EXP}(2 * \text{LN}(5))) * X * X * X - \text{ABS}(X - 5 * \text{SQR}(X))$$

Ответы.  $\frac{(x + \sin 5x)^2}{3\sqrt{x}} + 4; \arctan 5^{2x} * x^3 - |x - 5x^2|$

2) Изучение нового материала. (Учитель рассказывает о структуре программы на языке Turbo Pascal, ученики записывают основные понятия. В целях экономии времени, рекомендуется выдать ученикам карточки с основными понятиями, и учитель, в таком случае, только поясняет новую информацию.)

### 1. Структура программы на языке Pascal.

#### *I. Заголовок программы*

program <имя программы>

Например: program My\_primer;

*II. Раздел описаний*, в котором описываются все идентификаторы объектов, используемые в данной программе. Описать идентификатор - значит указать его имя и тип.

label <метки>

const <объявление констант>

var <объявление переменных>

Пример

```
label
M1, Tok, 123;
      const
nf = 1024
x = 13.14
c16 = $13D
slow = 'Связь'
var
a, b, c: real;
text: string
```

*III. Раздел операторов* (тело программы), в котором указывается последовательность действий, которые необходимо выполнить над описанными объектами. Начинается служебным словом begin и заканчивается словом end.

begin

оператор 1;

.....

оператор n;

end.

*Замечание:* в разделе описаний можно еще дать учащимся описание модулей, описание типов и описание процедур и функций, но на мой взгляд, эти описания лучше ввести учащимся непосредственно при изучении программ с использованием этих описаний.

### 2. Инструкции присваивания, вывода и ввода.

#### *а) Инструкция присваивания*

Общий вид инструкции

Имя: = выражение.

Например:

i := 0 - значение переменной i становится равным 0,

a := v+c - значением переменной a будет число, равное сумме значений переменных v и c

j := j+1 - значение переменной j увеличится на единицу.

#### *б) Инструкция вывода WRITE и WRITELN.*

Общий вид:

Write (a, v, c)

Например:

Write (Summa) - выведет на экран значение переменной Summa



Write ('Сумма=') - 'напечатает' на экране слово Сумма =

После имени переменной через двоеточие можно поместить описание поля ввода значения переменной.

Например: Если переменная Summa имеет значение равное 25,43, то при выполнении инструкции Write (Summa:5:1) на экран будет выведено число 25.4.

Инструкция Writeln отличается от инструкции Write только тем, что после вывода сообщения или значений переменных курсор переходит в начало следующей строки.

в) *Инструкции ввода READ и READLN*

Общий вид:

Read (Переменная1, переменная2, ..., переменная N);

Например: Read (a);

Read (Ток;Напр);

При выполнении инструкции read программа приостанавливает свою работу и ждет, пока на клавиатуре будут набраны нужные данные и нажата клавиша <Enter>. После нажатия <Enter> введенные значения присваиваются переменным, имена которых указаны в инструкции (по порядку)

Инструкция Readln отличается от Read тем, что после ее выполнения курсор переводится на следующую строку.

*Примечание:* после каждого оператора в программе ставится символ «;» (точка с запятой). В конце программы ставится точка.

Примеры программы

а) Написать программу, которая выводит на экран четверостишие :  
Унылая пора! Очей очарованье!

Приятна мне твоя прощальная краса-

Люблю я пышное природы увяданье, -

В багрец и золото одетые леса.

А.С.Пушкин

```
Program Primer_1; {инструкция ввода}
```

```
Begin
```

```
  Writeln('Унылая пора! Очей очарованье!');
```

```
  Writeln('Приятна мне твоя прощальная краса-');
```

```
  Writeln('Люблю я пышное природы увяданье, - ');
```

```
  Writeln('В багрец и золото одетые леса.');
```

```
  Writeln;
```

```
  Writeln('                               А.С.Пушкин');
```

```
end.
```

б) Написать программу вычисления периметра прямоугольника, если известны его стороны.

```
Program Primer_2; {инструкция присваивания ввода и вывода}
```

```
var a,b:real;{стороны прямоугольника}
```

```
    p:real; {периметр прямоугольника}
```

```
Begin
```

```
  Writeln('Вычисление периметра прямоугольника');
```

```
  Writeln('Введите стороны прямоугольника в сантиметрах');
```

```
  Readln(a,b);
```

```
  p:=2*(a+b);
```

```
  Writeln('периметр прямоугольника p=',p:5:2,'см');
```

```
End.
```

3) Практическая работа.

1.Наберите программы Primer\_1 и Primer\_2. И посмотрите результат их работы.

2. Написать программу вычисления площади прямоугольника, если известны его стороны.

Ответ:

```
program Ploshad_prjamoouolnica;
  var a,b:real; {стороны прямоугольника}
      S:real; {площадь прямоугольника}
  begin
    Writeln('Вычисление площади прямоугольника');
    Writeln('Введите стороны прямоугольника');
    Readln(a,b)
    S:=a*b
    Writeln('Площадь прямоугольника S=',S:5:2);
  end.
```

4) Домашнее задание

✓ Написать программу, которая выводит на экран строки:  
«Нет большего врага человеку, как он сам себе».

Св. Марк Подвижник.

Ответ:

```
Program Stroka;
Begin
  Writeln('«Нет большего врага человеку, как он сам себе.»');
  Writeln;
  Writeln('                Св. Маркс Подвижник.');
```

end.

✓ Написать программу вычисления площади квадрата, если известна его сторона.

#### Практическая работа

«Создание и реализация программ на ввод и вывод данных».

*Цель работы:* изучение операторов ввода и вывода величин в TURBO PASCAL, развивать навыки составления простейших программ.

Ход работы.

1. Приступая к решению задач данной практической работы, следует знать, что:

а) Для вывода на экран монитора сообщений и значений переменных служит инструкция Write (Writeln). Например: Write(Summa), Writeln('Результат вычислений'), Write('Корни уравнения x1=' , x1,' x2= ', x2), Writeln('x1=' , x1:5:2,' ', 'x2=' , x2:4:3).

б) Для ввода с клавиатуры во время работы программы исходных данных (значений переменных) служит инструкция Read( Readln). Например: Read(a), Readln( Cena,kol).

2. Наберите, приведенные ниже, программы и посмотрите различное оформление экрана пользователя при работе программы.

а) Program primer1;

```
  Begin
    Writeln('Буря мглою небо кроет,');
    Writeln('Вихри снежные крутя.');
```

```
  Writeln('То как зверь она завоет,');
  Writeln('То заплачет, как дитя.');
```

```
  Writeln('                ', ' А.С. Пушкин' );
```

```
  End.
```

б) Program primer2;

```
  begin
    Write ('x1=' , 'x2=');
```

```
    Write ('y1=' , 'y2=');
```

```
  End.
```

```

c) Program primer 3;
   Var x1 : real;
       x2 : real;
   begin
       writeln(' Введите два числа');
       Readln (x1,x2);
       Writeln (' Первое число' , x1, 'второе число', x2);
   End.

d) Program primer 4;
   Var a,b: real;
   Begin
       Write(' Введите число');
       Read (a);
       b:=2*a;
       Writeln('Первое число',a:2:1,'                ', 'второе число',
               b:3:2);
   End.

```

3. Выполните самостоятельно задания.

1. Написать программу, которая выводит на экран ваши имя и фамилию.
2. Написать программу, которая выводит на экран сообщение TURBO PASCAL 7.0.
3. Написать программу, которая выводит на экран четверостишие:  
 В одном мгновеньи видеть вечность,  
 Огромный мир - в зерне песка,  
 В единой горсти - бесконечность  
 И небо - в чашечке цветка.  
 У. Блейк. Пер. С. Маршак
4. Написать программу, которая вычисляет сумму двух чисел x1 и x2.
5. Написать программу, которая вычисляет значения функции  $y = \frac{x+2}{x^2+4}$

#### Урок 4

**Тема урока:** Практическая работа № 3 «Создание и реализация простейших линейных программ».

**Цель урока:** развитие навыков создания линейных программ с использованием встроенных функций и процедур.

#### Ход работы

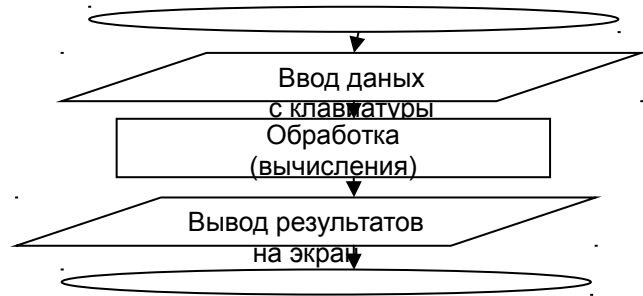
I. Приступая к решению задач с линейной структурой, следует вспомнить:

1) встроенные функции и процедуры

Математические	Комментарий
Abs(выражение)	Абсолютное значение аргумента
Sqr(выражение)	Квадрат аргумента
Sqrt(выражение)	Квадратный корень из аргумента
Sin(выражение)	Синус аргумента, выраженного в радианах
Cos(выражение)	Косинус аргумента, выраженного в радианах
Arctan(выражение)	Арктангенс аргумента
Exp(выражение)	Экспонента аргумента
Ln(выражение)	Натуральный логарифм аргумента
Int(выражение):real	Целая часть аргумента (не округляет)
Trunc(выражение):longint	Целая часть аргумента (не округляет)

Round(выражение): longint Str(выраже- ние;var строка: string)	Округление значения аргумента до ближайшего целого Преобразование числового выражения в строку
--	--

2) Алгоритм программы с линейной структурой.



II. Изучите примеры линейных программ

А) Пересчета расстояния из миль в километры

```

Program Primer_1;
  Var
    Mil: real;{расстояние в милях}
    Kilom: real;{расстояние в километрах}
  begin
    Writeln('Пересчет расстояния: Мили→ Километры');
    Writeln('Задайте расстояние в милях:');
    Readln(Mil);
    Kilom:= Mil * 1.6;
    Writeln(Mil:5:2,' миль - это', Kilom:5:2, ' км');
  end.
  
```

Вид экрана пользователя:  
 Пересчет расстояния: Мили→ Километры  
 Задайте расстояние в милях: 25  
 25.00 миль - это 40.00 км.

Б) Програма пересчета величины временного интервала (150 минут), в величину, выраженную в часах и минутах.

```

Program Primer_2;
  Var
    Min:integer;{интервал в минутах}
    H:integer;{количество часов}
    M:integer;{количество минут}
  Begin
    Writeln('Введите величину временного интервала в минутах');
    Writeln('-');
    Readln(min);
    H:=min div 60;
    M:=min div 60;
    Writeln;
    Writeln('min,'мин. - это',h,' часов',m,' минут');
  End.
  
```

В) Программа, которая преобразует введенное с клавиатуры дробное число 23,6 (десятичная дробь) в денежный формат 23 грн. 60 коп.

```

Program Primer_3;
  Var
    n:real;{дробное число}
    r:integer;{целая часть числа, гривны}
    k:integer;{дробная часть числа, копейки}
  Begin
    Writeln('Введите дробное число - ');
    Readln(n);
  
```

```

r:=Round(n*100) div 100;
k:=Round(n*100) div 100;
Writeln;
Writeln('n:6:2,'грн. - это',r,'грн.',k,'коп.');
```

End.

III. Выполните приведенные ниже задания самостоятельно согласно своего варианта.

1) Написать программу вычисления площади прямоугольника, если известны его стороны.

Вид экрана пользователя:

Вычисление площади прямоугольника

Введите исходные данные:

Длина (см) → 9

Ширина (см) → 7.5

Площадь прямоугольника: 67.50 кв.см.

2) Написать программу вычисления площади треугольника, если известна длина основания и высота, опущенная на основание.

Вид экрана пользователя:

Вычисление площади треугольника

Введите исходные данные:

Основание (см) → 8.5

Высота (см) → 10

Площадь треугольника 42.50 кв.см.

3) Написать программу вычисления площади треугольника, если известны длины двух его сторон и величина угла между этими сторонами.

4) Написать программу вычисления сопротивления электрической цепи, состоящей из двух последовательно соединенных сопротивлений.

5) Написать программу вычисления силы тока в электрической цепи, если известны сопротивление и напряжение в цепи.

6) Написать программу вычисления стоимости покупки, состоящей из нескольких тетрадей и карандашей.

7) Написать программу вычисления стоимости покупки, состоящей из нескольких тетрадей и такого же количества обложек к ним.

8) Написать программу вычисления стоимости покупки, состоящей из нескольких килограмм яблок и груш.

9) Написать программу вычисления сопротивления электрической цепи, состоящей из двух параллельно соединенных сопротивлений.

10) Написать программу вычисления расстояния между населенными пунктами, изображенными на карте.

11) Написать программу вычисления объема параллелепипеда, если известны длины трех его измерений.

12) Написать программу вычисления объема куба, если известна длина его ребра.

13) Написать программу вычисления объема цилиндра, если известны радиус его основания и высота цилиндра.

14) Написать программу вычисления площади поверхности цилиндра, если известны радиус основания и высота цилиндра.

15) Написать программу вычисления стоимости поездки на автомобиле на дачу (туда и обратно). Исходными данными являются: расстояние до дачи (в км); количество бензина, которое потребляет автомобиль на 100 км пробега; цена одного литра бензина.

16) Написать программу вычисления величины дохода по вкладу. Процентная ставка и время хранения (в днях) задаются во время работы программы.

- 17) Написать программу пересчета величины временного интервала, заданного в минутах, в величину, выраженную в часах и минутах.  
Совет: при составлении программы используйте операции а) целочисленного деления (div), б) деления по модулю (mod).
- 18) Написать программу, которая преобразует введенное с клавиатуры дробное число в денежный формат.
- 19) Написать программу вычисления площади параллелограмма, если известны длины его сторон и меньшая диагональ.
- 20) Написать программу вычисления площади треугольника, если известны длины его сторон.
- 21) Написать программу вычисления высот треугольника, если известны длины его сторон.
- 22) Написать программу вычисления площади прямоугольного треугольника, если известны длины гипотенузы и одного из катетов.
- 23) Написать программу вычисления большей диагонали параллелограмма, если известны длины его сторон и меньшая диагональ.
- 24) Написать программу вычисления радиусов вписанной и описанной окружностей треугольника, если известны длины сторон треугольника.
- 25) Составить программу для вычисления сопротивления цепи, состоящей из трех параллельно соединенных сопротивлений.
- 26) Составить программу, результатом выполнения которой будет выведено на экран заявление. (Значения <...> должны вводиться при выполнении программы).

Директору <школы>  
<Ф.И.О. директора>  
<Ф.И.О. родителя>

Заявление

Прошу зачислить моего сына (дочь) <Ф.И.О. ребенка> в <класс> Вашей школы.

<Дата>

<Подпись, фамилия родителя>

- 27) Составить программу для вычисления площади квадрата, если известны координаты двух противоположных вершин.
- 28) Составить программу для вычисления площади треугольника, если известны координаты его вершин.

### **Литература**

- 1) Гаевский А.Ю. Информатика:7 - 11. Учеб. Пособие. - К.:А.С.К., 2002. - 464с.:ил.
- 2) Культин Н.Б. Программирование в Turbo Pascal 7.0 и Delphi. - 2 - е изд., перераб. и доп. - СПб.:БХВ - Петербург, 2003. - 416с.: ил.
- 3) Культин Н.Б. Turbo Pascal в задачах и примерах. - СПб.:БХВ - Петербург, 2003. - 256с.: ил.
- 4) Малярчук С.Н. Основы информатики в определениях, таблицах и схемах: Справочно-учебное пособие / Под ред. Н.В. Олефиренко. - 4 - е изд.. - Х.:Веста: Изд-во «Ранок», 2005. - 112с.

## Урок 5

**Тема урока:** Организация разветвляющихся процессов

**Тип урока:** Изучение нового материала

**Цель урока:**

- Познакомить учащихся с организацией разветвляющихся процессов. Сформировать практические навыки программирования разветвляющихся процессов.
- Развивать у учащихся логическое мышление, память, скорость реакции, смекалку.
- Воспитывать творческий интерес к изучению нового материала.

### Структура урока

- I. Проверка домашнего задания (5 мин.)
- II. Актуализация опорных знаний (7 мин.)
- III. Мотивация обучения (3 мин.)
- IV. Изучение нового материала (10 мин.)
  1. Условный оператор.
  2. Метки. Оператор безусловной передачи управления.
  3. Оператор варианта.
- V. Закрепление материала (15 мин.)
- VI. Итоги урока (5 мин.)
- VII. Домашнее задание

### Ход урока

**I.** Проверка домашнего задания.

Вопросы.

1. Как записывается оператор вывода?
2. Как записывается оператор ввода?
3. Как записывается оператор присвоения?
4. Чем заканчивается программа?
5. Выберите правильный ответ в записи выражения:  
$$ab - \cos(a + c) - e^a / (\sqrt{\ln x} + 5) + 2a;$$
  - a)  $a*b - (\cos(a+c) - \exp(a)) / \text{sqrt}(\ln(x)) + 2a;$
  - b)  $a*b - (\cos(a+c) - \exp(a)) / (\text{sqrt}(\ln(x)+5) + 2*a);$
  - c)  $a*b - (\cos(a+c) - \exp(a)) / (\text{sqrt}(\ln(x)+5) + 2*a);$
  - d)  $a*b - (\cos(a+c) - \exp(a)) / (\text{sqrt}(\ln x + 5) + 2*a);$
6. Укажите, какие из записанных указаний ввода являются правильными, а какие нет и почему?
  - a) `read(x+1,y);`                      b) `read(x);`
  - c) `read(x),(y);`                      d) `read(x, y);` e) `read[x.y];`
7. Укажите какие из записанных указаний вывода являются правильными, а какие нет и почему?
  - a) `write(x,y);`                      b) `write(a b);`                      c) `write a,b;`
  - d) `write [a.b];`                      e) `write 5;`                      f) `write(a,b*c.);`
  - g) `write(a,b);`                      h) `write(-a,b.);`

**II.** Актуализация опорных знаний учащихся.

1. Перечислите базовые структуры. Почему они называются базовыми?
2. Что означает структура ветвления? Приведите примеры алгоритмов, использующих базовую структуру ветвления.

**III.** Мотивация обучения.

**IV.** Изучение нового материала.

1. Условный оператор необходим для выбора одного варианта решений из двух или более возможных. Условный оператор может принимать одну из двух конструкций:

1) Полная форма: `if <условие> then<оператор 1>else<оператор 2>;`

2) Сокращенная форма: `if<условие> then<оператор 1>;`  
где <условие> – выражение булевого типа.

Условие может быть простым или составным, например:

1) Простое условие `a>5, c<=2.5, x=0.`

2) Составное условие: `(a>5)and(a<10), (a=3)or(b=4.5).`

Составные условия записываются с помощью логических операций: `and` – и, `or` – или, `not` – не. При записи условия можно использовать всевозможные операции отношения.

<оператор 1>, <оператор 2> – отдельные операторы или операторы, сгруппированные вместе при помощи операторных скобок: `begin` и `end`. Полная форма условного оператора выполняется следующим образом:

– если условие истинно, то выполняется оператор 1, а оператор 2 выполняет действие, которое находится в программе после указания разветвления;

– условие ложно, то выполняется оператор 2, а оператор 1 выполняет действие, которое находится в программе после указания разветвления.

Сокращенная форма условного оператора выполняется следующим образом:

– если условие истинно, то выполняется оператор 1, а затем выполняется действие, которое находится в программе после условного оператора;

– условие ложно, то выполняется оператор 2, а затем выполняется действие, которое находится в программе после условного оператора.

## 2. Метки.

Иногда возникает необходимость нарушить порядок действий в программе. Первое указание управления – условный оператор мы рассмотрели. Второй способ изменения порядка выполнения действий осуществляются с помощью меток и оператора `goto`.

Метка – это произвольное целое число в интервале от 0 до 9999 или произвольный идентификатор. Метка записывается перед оператором и отделяется от него двоеточием. Все метки, которые используются в программе, должны быть описаны в разделе описания меток. Этот раздел открывается служебным словом `label`, записывается:

```
label метка1, метка2, ..., метка n;
```

```
Например: label m1,1,2,3;
```

Идентификатор метки не должен совпадать с идентификатором переменной или константы.

Указание безусловного перехода `goto` имеет вид:

```
goto метка;
```

```
Например: goto m1;
```

Оператор безусловного перехода нарушает принцип структурного подхода к написанию программ на языке программирования Паскаль, поэтому использовать их необходимо в случае крайней необходимости.

## 3. Оператор варианта.

Если необходимо осуществить проверку более сложных условий, чем да/ нет, целесообразно использовать оператор варианта:

```
case <селектор>of
```

```
<список меток 1>: <оператор 1>;
```

```
<список меток 2>: <оператор 2>;
```

```
...
```



<список меток n>:<указания n>

end;

где <селектор> - выражение любого скалярного типа, кроме вещественного;

<оператор> - любой оператор языка; <список меток> - список разделенных запятыми значений выражения;

Оператор варианта выбирает для использования тот оператор, одна из меток которого равна текущему значению выражения селектор. По окончании выполнения выбранного оператора управления передается в конец оператора case

Пример. Пусть i - переменная целого типа, тогда

case i of

1: y: =sin(x);

2: y: =cos(x);

3: y: =abs(x);

end;

**V. Закрепление материала.**

1. Составить программу вычисления значения функции:

$$y = \begin{cases} x^2, & \text{если } x < 0 \\ 5x, & \text{если } 0 \leq x < 10 \\ 5x^3 + 1, & \text{если } x \geq 10 \end{cases}$$

Решение:

```
program primer;
  var x,y: real;
begin
  write('Введите числовое значение x=');
  readln(x);
  if x<0 then y: =sqr(x)
    else if (x>=0) and (x<10)
      then y: =5*x
      else y: = 5*x*sqr(x) + 1;
  writeln('при x=', x:5:2, 'y=', y:5:2);
end.
```

2. Составить программу нахождения корней квадратного уравнения.

```
program primer;
var a,b,c,d,x1,x2:real;
begin
write('Введите коэффициенты квадратного уравнения a, b, c');
readln (a,b,c);
d:=sqr(b)-4*a*c;
if d<0 then writeln('Уравнение имеет комплексные корни')
  else if d>0 then begin
  x1:=(-b-sqrt(d)) / (2*a);
  x2:=(-b+sqrt(d)) / (2*a);
  writeln('Корни:x1=', 1:6:2, 'x2=', x2:6:2);
end
else begin
  x1:= - b/2*a;
  writeln('x=', x1:6:2);
end;
end.
```

3. Составить программу с выполнением на ПК, которая определяет сезон года по порядковому номеру месяца.

```
program primer;
var month:integer;
```

```

begin
write('Введите порядковый номер месяца:');
readln(month);
if(month<1)or(month>12)
then writeln('Такого месяца нет')
else
  case month of
3,4,5:writeln ('Весна ');
6,7,8: :writeln ('Лето');
9,10,11: writeln ('Осень');
12,1,2:writeln ('Зима')
end;
end.

```

#### VI. Итоги урока

Учитель обращает внимание учащихся о целесообразности использования каждого из операторов языка Паскаль, организующих разветвляющиеся процессы.

#### V. Домашнее задание.

1. Прочитать учебник А. Ф. Верлань, п.2.5 стр. 135-141;
2. Выполнить упражнения 13, 14 стр. 141.

## Урок 6

**Тема урока:** Цикл с параметром

**Тип урока:** Изучение нового материала.

#### **Цель урока:**

- Познакомить учащихся с организацией циклических процессов, использующих цикл с параметром. Сформировать практические навыки и умения программирования циклических процессов, использующих цикл с параметром.
- Развивать логическое мышление, память, смекалку.
- Воспитывать творческий интерес к изучению нового материала.

#### **Структура урока**

- I. Проверка домашнего задания (5 мин.)
- II. Актуализация опорных знаний (7 мин.)
- III. Мотивация обучения (3 мин.)
- IV. Изучение нового материала (10 мин.)
  - 1.Понятие цикла.
  - 2.Цикл с параметром.
- V. Закрепление материала (15 мин.)
- VI. Итоги урока (5 мин.)
- VII. Домашнее задание.

#### **Ход урока**

#### **I. Проверка домашнего задания.**

Вопросы.

- 1.Как записывается и выполняется условный оператор в полной форме?
- 2.Как записывается и выполняется условный оператор в сокращенной форме?
- 3.Какие особенности использования символа ";" в условном операторе?
- 4.Как описываются и для чего предназначены метки?
- 5.Как описывается и для чего используется оператор безусловной передачи управления goto?
- 6.Как описывается и выполняется оператор варианта?
- 7.Что такое селектор и укажите требования его использования?

#### **II. Актуализация опорных знаний.**

1. Запишите выражение в форме оператора присвоения на языке программирования Паскаль:

$y = \cos^2 5i + \sin^3 i$  .

2. В приведенной программе, записанной на доске, допущены ошибки. Исправьте допущенные ошибки.

```
Program primer
var x:real;
begin
  read(x);
  y:=sqrt(x+1)/t
  writeln('y=', y);
and.
```

3. Какое слово используется для описания величин целого типа на языке программирования Паскаль:

a) real; b) var; c) integer.

4. Что такое алгоритм циклической структуры?

5. Что необходимо предусмотреть для организации цикла?

6. Что такое параметр цикла?

**III.** Мотивация обучения.

**IV.** Изучение нового материала.

1. Программа, в которой предусмотрено неоднократное выполнение одной и той же последовательности действий, при различных значениях переменных, называется программой циклической структуры.

При записи циклических алгоритмов на языке Паскаль, используют три структуры цикла:

- цикл с параметром;
- цикл с предусловием;
- цикл с постусловием.

2. Для описания циклических процессов, где число повторений заранее известно, используют оператор цикла с параметром.

Оператор цикла с параметром записывается в виде:

1. for<параметр>:=<выражение 1> to <выражение 2>do

```
begin
  <оператор 1>;
  ...
  <оператор n>;
end;
```

2. for<параметр>:=<выражение 1> downto <выражение 2>do

```
begin
  <оператор1>;
  <оператор 2>;
  ...
  <оператор n>;
end;
```

где параметр- переменная скалярного типа, кроме действительного;

<выражение 1>(начальные значения параметра)- константа такого же типа, как и параметр цикла;

<выражение 2> (конечное значение параметра)- константа такого же типа, как и параметр цикла.

При использовании в циклах служебного слова to- значение параметра цикла увеличивается на 1, а при downto- уменьшается на 1.

Примечание.

1. Оператор цикла с параметром удобно использовать, в тех случаях, когда предварительно можно определить количество повторений.

2. Параметр цикла не должен изменять своего значения в середине тела цикла.

3. Параметр цикла должен быть описан в разделе описания переменных.

4. Значение параметра цикла после окончания цикла имеет конечное значение, поэтому перед повторным его использованием ему необходимо вновь присвоить новое значение.

#### V. Закрепление материала.

1. Составить программу вычисления значения функции.

$y=5i^2+1$ , где  $i \in [1;n]$ , с шагом 1.

```
program primer;
  var i:integer;y:real;
begin
  for i:=1 to n do
    begin
      y:=5* sqr(i) +1;
      writeln('при i=',i , 'y=',y:6:2);
    end;
end.
```

2. Составить программу и выполнить на рабочем месте.

Вычислить значение суммы:  $S=a+a^2+\dots+a^n$ , при фиксированных значениях  $n$  и  $a$ .

```
program primer;
var n,i:integer;a1,s,a:real;
begin
  write(' введите числовое значение a=');
  readln(a);
  write(' введите числовое значение n=');
  readln(n);
  S:=0; a1=a;
  for i:=1 to n do
  begin
    s:=s + a1;
    a1:=a1*a;
  end;
  writeln('S=', s:7:2);
end.
```

3. Составить программу вычисления факториала числа  $n$ :  $n! = 1*2*3*\dots*n$ .

```
program primer;
var n,i:integer; factorial:longint;
begin
  write( 'Введите значение натурального числа n=');
  readn(n);
  factorial:=1;
  for i:=1 to n do factorial:=factorial*i;
  writeln('n!=',factorial);
end.
```

#### VI. Итоги урока.

Учитель обращает внимание учащихся о целесообразности использования при организации циклических процессов цикла с параметром.

#### VII. Домашнее задание.

Прочитать стр. 145-147; выполнить №4(а) стр. 150.

## Урок 7

**Тема урока:** Оператор цикла с предусловием и постусловием

**Тип урока:** Изучение нового материала

**Цель урока:**

- Познакомить учащихся с организацией циклических процессов, использующих цикл с предусловием и постусловием на языке программирования Паскаль. формировать практические навыки и умения программирования циклических процессов с дальнейшей реализацией на ПК.
- Развивать логическое мышление, память.
- Воспитывать творческий интерес к изучению нового материала.

#### **Структура урока.**

- I. Проверка домашнего задания (5 мин.)
- II. Актуализация опорных знаний (7 мин.)
- III. Мотивация обучения (3 мин.)
- IV. Изучение нового материала (10 мин.)
  1. Цикл с предусловием
  2. Оператор цикла с постусловием
- V. Закрепление нового материала (15 мин.)
- VI. Итог урока (5 мин.)
- VII. Домашнее задание

#### **Ход урока**

I. Проверка домашнего задания.

##### Задания.

- a. Определить значения переменной S, после выполнения следующих операторов: `s:=0; n:=5; for i:=2 to n do s:=s+sqr(i);`
- b. Как записывается оператор цикла с параметром? Как он работает?
- c. Как организуется выход с указания повторения?
- d. Можно ли в середине тела цикла изменить параметр цикла в циклах с параметром?

II. Актуализация опорных знаний.

1. Что необходимо предусмотреть для организации цикла в циклических алгоритмах, использующих циклы с постусловием и предусловием.
2. Какая разница в организации циклов с "постусловием" и "предусловием"?

III. Мотивация обучения.

IV. Изучение нового материала.

1. Цикл с "предусловием" используется для программирования процессов, в которых число повторение оператора цикла неизвестно, а задается некоторое условие его окончания.

Выполнение оператора цикла с "предусловием" начинается с проверки условия, записанного после слова `while`. Если оно соблюдается, то выполняется <тело цикла>, а затем вновь проверяется условие и так далее. Как только на определенном шаге окажется, что условие не соблюдается, то выполнение <тела цикла> прекратится.

Вид оператора:

```
While <проверяемое условие> do
  begin
    <оператор 1>;
    ...
    <оператор n>;
  end.
```

Примечание.

- Если проверяемое условие ложно при первой же проверки, то тело цикла не выполняется ни одного раза.
- Проверяемое условие необходимо подбирать таким образом, чтобы в определенный момент оно изменило свое значение с `true` на `false`, иначе возникает ситуация "зацикливания" программы.

- Необходимо предусматривать в теле цикла изменение параметра цикла.
- Вхождение в цикл возможно только через заголовок этого цикла.

2. Для программирования циклических процессов с неизвестным числом повторений существует еще один оператор - оператор цикла с постусловием, который имеет следующий вид:

```
repeat
    <оператор 1>;
    ...
    <оператор n>;
until <условие>;
```

Этот оператор аналогичен оператору цикла с "предусловием", но отличается от него тем, что проверка условия производится после очередного выполнения тела цикла. Это обеспечивает его выполнение хотя бы один раз.

Примечание.

1. Тело цикла, которое состоит из группы операторов, не нужно брать в операторные скобки (begin-end), поскольку эту роль выполняют служебные слова repeat - until.
2. Тело цикла выполняется хотя бы один раз.
3. Вход в цикл осуществляется через заголовок цикла.

#### V. Закрепление нового материала.

1. Найдите сумму всех натуральных чисел от одного до n (используя цикл с предусловием).

```
Program primer;
var i, n, s: integer;
begin
    write(' Введите количество натуральных чисел');
    readln (n);
    i:= 1; s:=0;
    while i<=n do
        begin
            s:=s+i;
            i:=i+1;
        end;
    writeln('S=', s);
end.
```

2. Найдите сумму натуральных чисел от 1 до n (цикл с постусловием)

```
program primer;
var i,n,s:integer;
begin
    write('Введите количество натуральных чисел');
    readln(n);
    i:=1; s:=0;
    repeat
        s:=s+i;
        i:=i+1;
    until i>n;
    writeln(' S= ', s)
end.
```

3. Составить программу нахождения НОД(x,y) с реализацией на ПК.

```
program primer;
uses crt;
var x,y:integer;
begin
    write(' x='); readln(x);
```

```

write ('y='); readln(y);
repeat
  if x>y then x:=x mod y else y:=y mod x
until (x = 0) or (y = 0);
writeln(' НОД=', x+y);
end.

```

**V.** Подведение итогов урока.

Учитель акцентирует внимание учащихся на принципиальной разнице в выполнении циклов с "постусловием" и "предусловием".

**VI.** Домашнее задание: прочитать п.2.6 стр.141, выполнить №5 стр.150.

## Урок 8

**Тема урока:** Вложенные циклы.

**Тип урока:** Изучение нового материала.

**Цель урока:**

- Познакомить учащихся с вложенными циклами.
- Сформировать практические навыки и умения использования вложенных циклов.
- Развивать логическое мышление, память.
- Воспитывать творческий интерес к изучению нового материала.

### Структура урока.

- I. Проверка домашнего задания (5 мин.)
- II. Актуализация опорных знаний учащихся (7 мин.)
- III. Мотивация обучения (3 мин.)
- IV. Изучение нового материала (10 мин.)
- V. Закрепление материала (15 мин.)
- VI. Итоги урока (5 мин.)
- VII. Домашнее задание.

### Ход урока.

**I.** Проверка домашнего задания.

1. Как записывается и выполняется оператор цикла с предусловием?
2. Как записывается и выполняется оператор цикла с постусловием?
3. Какая разница между циклами с предусловием и постусловием?

**II.** Актуализация опорных знаний.

1. Какой результат выполнения фрагмента программы?

```

f:=2, i:=16;
while i>0 do
  begin
    f:=2*f-1; i:=i-4;
  end;
write f;
end.

```

2. Указать значения переменной S после выполнения оператора цикла:

- 1) s:=3; i:=1; while i<6 do begin i:=i+1; s:=s+i; end;
- 2) s:=2; i:=1; repeat s:=s+i; i:=i+1; until i>3;
- 3) s:=0; for i:=5 to 7 do s:=s+1;
- 4) s:=1; for i:=10 downto 2 do s:=s+1;

**III.** Изучение нового материала.

Вложенные циклы.

При решении некоторых задач приходится использовать вложенные циклы. Внутренний и внешний циклы могут быть любыми из трех рассмотренных ранее видов циклами: с параметром, с предусловием и постусловием. При использовании вложенных циклов необходимо соблюдать следующие условия: внутренний цикл должен полностью укладываться в циклическую часть внешнего цикла.

#### IV. Закрепление материала

1. Составить программу вычисления значения функции с реализацией на ПК:  $y = 3^{-x \cos x} + \ln[c - 6.3] / \sqrt[7]{x^3 + e^{-x} \sin x}$   
xe[xн, xк], шагом h.  
се[сн, ск], шагом dc.

```
program primer;  
uses crt;  
var xn, xk, h, x, y, a, b, c, cn, ck, dc: real;  
begin  
  clrscr;  
  write(' введите xn, xk, h'); readln(xn, xk, h);  
  write('введите cn, ck, dc'); readln(cn, ck, dc);  
  x:=xn;  
  while x<=xk do  
  begin  
    c:=cn;  
    while c<=cn do  
    begin  
      a:=x*sqr(x)+exp(x)+sin(x);  
      b:=c-6.3;  
      if (a<>0) and (b<>0)  
      then begin  
        a:=a/abs(a)*exp(ln(abs(a))/7);  
        y:=(exp(x*cos(2*x)*ln3))+ln(abs(b))/a;  
        writeln('x=', x:6:2, 'c=', c:6:2, ' y=', y:10:3);  
      end  
      else writeln('решение не существует');  
      x:=x+h;  
    end;  
    c:=c+dc;  
  end;  
end.
```

2. Сколько можно купить быков, коров, телят, если плата за быка 10 рублей, корову - 5 рублей, за теленка - полтинник (0.5 рублей), если на 100 рублей надо купить 100 голов скота.

Обозначим: b-количество быков, k-количество коров, t- количество телят. После этого можно записать:

$10b + 10k + 10t = 100$  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.  $0 \leq b \leq 10$ ;

-не более 20 коров, т.е.  $0 \leq k \leq 20$ ;

-не более 200 телят, , т.е.  $0 \leq t \leq 200$ ;

```
program primer;  
uses crt;  
var b, k, t: integer;  
begin  
  clrscr;  
  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;  
    end;  
  end;  
end.
```

3. Что будет выведено на экране монитора после выполнения следующего фрагмента программы?

a:=1;b:=1;



```

for i:=0 to 5 do
  begin
    for j:=1 to b do write('*');
    writeln; c:=a+b; a:=b; b:=c;
  end;

```

#### V. Подведение итогов урока.

Учитель акцентирует внимание учащихся на следующем моменте, что при фиксированном параметре внешнего цикла внутренний цикл выполняется полностью.

#### VI. Домашнее задание.

Выучить конспект урока и выполнить задачу:

Составить программу нахождения натурального числа от 1 до 10000 с максимальной суммой делителей.

## Урок 9

**Тема урока:** Одномерные массивы. Работа с элементами.

**Тип урока:** Изучение нового материала.

### Цель урока:

- Познакомить учащихся с одномерными массивами.
- Сформировать практические навыки и умение составлять и реализовывать программы обработки одномерных массивов на языке программирования Паскаль.
- Развивать у учащихся умения делать выводы и обобщения при изучении нового материала.
- Воспитывать творческий интерес к изучению нового материала.

### Структура урока

- I. Проверка домашнего задания (5 мин.)
- II. Актуализация опорных знаний (7 мин.)
- III. Мотивация обучения (3 мин.)
- IV. Изучение нового материала (10 мин.)
  1. Структурированный тип данных.
  2. Одномерные массивы. Работа с элементами.
- V. Закрепление материала (15 мин.)
- VI. Итоги урока (5 мин.)
- VII. Домашнее задание

### Ход урока

#### I. Проверка домашнего задания.

##### Вопросы.

1. Дать определение вложенного цикла.
2. Принцип выполнения вложенных циклов.
3. Привести пример с вложенным циклом.

#### II. Актуализация опорных знаний учащихся

1. Как записывается и выполняется оператор цикла с параметром?
2. В операторе for...to шаг равняется:
  - а) 2;                      б) 1;                      в) любое другое значение.
3. Сколько раз будет выполняться цикл
 

```
for i:=3 downto -1 do i:=i+1;
```

  - а) 0;                      б) 5;                      в) 1;                      г) 4;
4. Параметр цикла в операторе for может быть такого типа:
  - а) char;                      б) real;                      в) string;                      г) integer?

#### III. Мотивация обучения.

#### IV. Изучение нового материала.

1. В языке программирования Паскаль к структурированным типам относятся:
  - а) массив;

- б) запись;
- в) множество;
- г) файл;

2. Массив – это структурированный тип данных, состоящий из конечного числа данных одного типа. Массив обозначается одним именем. Каждый элемент массива обозначается именем массива с индексом. Массивы бывают: одномерные и двумерные.

Одномерный массив – это линейно упорядоченная совокупность элементов одного и того же типа.

Характерные свойства массива:

1. Тип элементов массива должен быть конкретно описанный. К каждому элементу массива существует прямой доступ.
2. Число элементов массива определяется во время его описания и дальше не изменяется.

Описать массив можно:

1-ый способ:

```
type имя типа = array[тип индекса] of тип компонентов массива;
var идентификатор массива: имя типа;
```

Пример

```
type massiv=array[1..5]of integer;
var a:massiv;
```

2-ой способ:

```
var идентификатор массива:array[индекс]of тип элемента
массива;
```

где идентификатор массива – правильный идентификатор,

тип элемента массива– любой тип языка программирования Паскаль

Пример:

```
var a:array[1..5]of integer;в:array[1..7]of real;
```

При обращении к массиву необходимо указать идентификатор имени массива, а так же индекс, указывающий на местоположение элемента в массиве.

Например: a[i];a[5];

Ввод одномерного массива:

```
for i:=1 to n do
  writeln ('a[',i,']=');
```

Вывод одномерного массива :

```
for i:=1 to n do
  writeln('a[',I,']=',a[i]);
```

**V. Закрепление нового материала.**

1. Составить программу вычисления суммы отрицательных элементов массива a<sub>1</sub>,a<sub>2</sub>,...a<sub>n</sub>.

```
program primer;
  const n=10;
  var a:array [1..n] of real; s:real; i:integer;
begin
  for i:=1 to n do
    begin
      write('Введите элемент массива a[',i,']=');
      readln(a[i]);
    end;
  s:=0;
  for i:=1 to n do
    begin
      if a[i]<0 then s:=s+a[i];
    end;
```

```
writeln ('Сумма отрицательных элементов массива s:=' ,s:6:2);
end.
```

2. Задан массив  $a_1, a_2, \dots, a_n$ . Подсчитайте количество положительных, отрицательных, нулевых значений.

```
program primer;
uses crt;
const n=10;
var i,k,l,m:integer;
    a:array[1..n] of integer;
begin
for i:=1 to n do
    write('a[',i,']='); readln(a[i]);
end;
k:=0;l:=0;m:=0;
for i:=1 to n do
    begin
        if a[i]=0 then k:=k+1 else
        if a[i]>0 then l:=l+1 else m:=m+1;
    end;
writeln(' Количество элементов равных 0=',k);
writeln(' Количество отрицательных элементов = ',m);
writeln(' Количество положительных элементов = ',l);
end.
```

3. Задан массив  $a_1, a_2, \dots, a_n$ . Расположите элементы в том же массиве в порядке возрастания. Выполните программу на ПК.

```
program primer;
const n=20;
var i,j,p:integer; a:array[1..n] of integer;
begin
for i:=1 to n do
    begin
        write('a[',i,']=');
        readln(a[i]);
    end;
for i:=1 to n-1 do
    begin
        for j:=i+1 to n do
            begin
                if a[i]>a[j]
                then begin
                    p:=a[i];
                    a[i]:=a[j];
                    a[j]:=p;
                end;
            end;
        end;
for i:=1 to n do
    writeln('a[',i,']= ',a[i]);
end.
```

**V.** Итоги урока.

Учитель обращает внимание учащихся на необходимость описания в разделе описания массивов, которыми пользуются в программе.

**VI.** Домашнее задание. П.2.8 стр.158, выполнить упр. 11 стр. 165.

## Урок 10

**Тема урока:** Двумерные массивы. Работа с элементами.

**Тип урока:** Изучение нового материала.

**Цель урока:**

- Познакомить учащихся с двумерными массивами
- Сформировать практические навыки, умения составлять и реализовывать программы обработки двумерных массивов на языке программирования Паскаль.
- Развивать логическое мышление, память, смекалку.
- Воспитывать творческий интерес к изучению нового материала.

#### Структура урока

- I. Проверка домашнего задания (5 мин.)
- II. Актуализация опорных знаний (7 мин.)
- III. Мотивация обучения (1 мин.)
- IV. Изучение нового материала (20 мин.)  
Двумерные массивы. Работа с элементами.
- V. Закрепление материала (5 мин.)
- VI. Итоги урока (5 мин.)
- VII. Домашнее задание (2 мин.)

#### Ход урока

- I. Проверка домашнего задания.
  1. Чем отличается массив от файла?
  2. Для чего необходимо описание массива?
  3. По каким признакам данные объединяются в массивы?
  4. Как осуществляется обращение к элементу одномерного массива?
- II. Актуализация опорных знаний учащихся.
  1. Дать определение вложенного цикла. Сформулировать принцип выполнения вложенных циклов.
  2. Как записывается и выполняется оператор цикла с параметром?
  3. Какой из операторов записан верно:
    - a) for a:=3 to -8 do p=p\*a;
    - б) for x:=2 to 68 do z:=x+y;
    - в) for i:=3 downto 55 do writeln (a);

III. Мотивация обучения.

IV. Изучение нового материала.

Массивы, положение элементов в которых описываются двумя индексами называются *двумерными*. Их можно представить в виде прямоугольной таблицы или матрицы.

Существует несколько способов объявления двумерного массива.

Способ 1. В Паскале двумерный массив можно описать как одномерный, элементами которого являются одномерные массивы.

Пример.

```
const n=2;m=3;
type mas1=array[1..m] of <тип элементов>;
      mas2=array[1..n] of mas1;
var v: mas1;
    a: mas2;
```

Способ 2.

```
type имя типа=array[тип индекса] of array[тип индекса]of [тип
компонентов];
var идентификатор массива: имя типа;
```

Пример.

```
type massiv= array [1..5] of array [ 1..4] of integer;
var mamr: massiv;
```

Способ 3.

```
var имя массива :array [1..n,1..m]of <тип элементов массива>;
```

Пример.

```
var matr:array [1..10,1..4]of integer;
```

Обращение к элементам массива осуществляется следующим образом  
matr [i,j]элемент массива matr, который находится на пересечении i-  
строки и j-столбца.

*Ввод двумерного массива:*

```
for i:=1 to n do
  for j:= 1 to m do
    begin
      write(введите элемент массива a['\,i',';',',j,']=');
      readln(a[i,j]);
    end;
```

*Вывод двумерного массива:*

```
for i:=1 to n do
  begin
    for j:=1 m do
      write (a[i,j]);
    writeln;
```

```
end;
```

## V. Закрепление материала.

1. Составить программу вычисления произведения положительных элементов двумерного массива.

```
program primer;
  const n=10;m=6;
  var a: array[1..n,1..m]of real; i,j: integer; p:real;
begin
  for i:=1 to n do
    begin
      for j:=1 to m do
        begin
          write('\введите элемент массива a['\,i',';',',j,']=');
          readln(a[i,j]);
        end;
```

```
end;
```

```
p:=1;
```

```
for i:=1 to do
```

```
  for j:=1 to m do
```

```
    if a[i,j]>0 then p:=p+a[i,j];
```

```
  writeln(`Сумма положительных элементов таблицы =`, p:6:2);
```

```
end.
```

2. Найти максимальный элемент каждого столбца таблицы и выполнить на ПК.

```
Program primer;
  const n=5; m=6;
  var a:array[1..n,1..m]of real; max:real;i,j: integer;
      b:array[1..m]of real;
begin
  for i:=1 to n do
    begin
      for j:=1 to m do
        begin
          write('\введите элемент a ['\,i',';',',j,']=');
          readln(a[i,j]);
        end;
```

```
end;
```

```

for j:=1 to m do
  begin
    max:=a[1,j];
    for i:=1 to n do
      if a[i,j]>=max then max:=a[i,j];
    b[j]:=max;
  end;
for i:=1 to m do writeln (' b[',i,']=',b[i]);
end.

```

3. В квадратной таблице с правого бока от побочной диагонали, включая и ее, найти наименьший элемент и соответствующий номер строки.

```

program primer;
const n=5;
var a: array[1..n,1..n] of real;
    i,j,nomer: integer; min: real;
begin
  for i:=1 to n do
    for j:=1 to n do
      begin
        write('введите элемент массива a [',i,',',j,']=');
        readln(a[i,j]);
      end;
  nomer:=1;
  min:=a[1,n];
  for i:=1 to n do
    for j:=n-i+1 to n do
      if a[i,j]<=min
        then begin
          min:=a[i,j];
          nomer:=i;
        end;
  writeln(' минимальное значение =', min:6:2);
  writeln('номер строки =', nomer);
end.

```

#### **VI.** Итоги урока.

Учитель акцентирует внимание учащихся на необходимость описания в разделе описания двумерного массива, которым пользуются в программе, а так же при обработке элементов двумерного массива необходимо использовать вложенный цикл.

#### **VII.** Домашнее задание.

Прочитать п.2.9 стр.165;выполнить №6 стр.171.