

Вопрос №19

Символы и элементы языка Паскаль

В язык входят:

26 латинских букв; цифры 0 - 9;

спец. клавиши : = + - * / ^ v > <
() { } [];

Знаки арифметических операций: сложение
+ ; вычитание - ; умножение *; деление /

Знаки логических операций

> ; < ; >= ; <= ; = ; < >

Div - деление нацело без остатка;

Mod - нахождение остатка от деления нацело

Примеры

$$13 \operatorname{div} 5 = 2$$

$$13 \bmod 5 = 3$$

$$3 \operatorname{div} 8 = 0$$

$$3 \bmod 8 = 3$$

$$5 \operatorname{div} 5 = 1$$

$$5 \bmod 5 = 0$$

Ключевые слова:

and -и;	array -массив;
begin -начало;	case -вариант;
const – константа;	do -выполнить;
downto -уменьшая до;	else - иначе;
end -конец;	for -для;
function -функция;	go to -идти к метке n;
if -если;	label -метка;
not -не;	of -из;
or -или;	procedure -процедура;
program -программа;	repeat -повторять;
then -то;	to -увеличивая до;
type -тип;	until - выполнять до;
var -переменная;	while - пока.

Вопрос №20

Описание данных. Стандартные функции.

$\sqrt[2]{x}$

1. **integer**- описание целых чисел;
2. **real**- описание дробных чисел;
3. **boolean (true - истина, false-ложь)** — описание логических констант.;
4. **char**- описание символьных констант.
5. **ABS(X)** — $|x|$
6. **SQR(X)** — x^2 ;
7. **SQRT(X)** — $\sqrt[2]{x}$;
8. **LN(X)** — $\ln x$;
9. **LN(X)/ LN(N)** — $\log_n x$;
0. **LN(X)/ LN(10)** — $\lg x$;
1. **EXP(X)** — e^x ;
2. **EXP(n*LN(X))** — x^n ;
3. **SIN(X)** — $\sin x$;
4. **COS(X)** — $\cos x$;
5. **SIN(X)/COS(X)** — $\operatorname{tg} x$;
6. **COS (X)/SIN (X)** — $\operatorname{ctg} x$;
7. **ARCTAN(X)** — $\operatorname{arctg} x$;
8. **PI** — π ;
9. **RANDOM(X)** — случайное число из диапазона от 0 до x;
0. **RANDOMIZE** — обеспечение несовпадений случайных чисел, генерируемых процедурой **RANDOM**;
1. **TRUNC(X)** — отбрасывание дробной части;
2. **ROUND(X)** — округление до ближайшего целого;

Арифметические выражения с высокой степенью записываются по формуле:

$$x^n = e^{n \ln x};$$

1. $x^4 = \exp(4 * \ln(x)); \quad x^4 = \text{sqr}(x) * \text{sqr}(x)$

2. $\sin^5 x = \exp(5 * \ln(\sin(x)));$

3. $\sin^5 x^5 = \exp(5 * \ln(\sin(\exp(5 * \ln(x)))))$;

Пример описания массивов:

var a:array[1..5] of real ;

var a:array[1..5,1..40] of integer.

Оператор присваивания: $a:=b$; $x:=0.25$;
 $x:=x+0.25$ и т.д.

Объединение таких простых предложений называется составным. В начале составного предложения ставится слово **begin** а в конце **end**

```
begin  
     $a:=b$ ;  
     $x:=0.25$ ;  
     $x:=x+0.25$   
end.
```

Вопрос №21 Операторы ввода-вывода

`read(a,b,c) ; write(a,b,c)` → ввода и вывода
`readln(a,b,c) ; writeln(a,b,c)` → после вв. и выв.
осуществляется переход к новой строке

Пример

`write(a:8:3)` → 8 всего позиций, 3 в дробной части;

`write(a, ' ', :4, b)` → между переменными **a** и **b** - 4 пробела

Вопрос №22

Операторы условного и безусловного перехода

Оператор безусловного перехода:

go to *n*

*где n - метка оператора на который
осуществляется переход*

Условные операторы

if *a* then *b*;

if *a* then *b* else *c*

*где a – условие , b,c – арифметическое
выражение*

$$a = \begin{cases} b^2, & x < 1 \\ x^2, & x \geq 1 \end{cases}$$

Применение условного оператора

```

program ime;
var a,b,x:real;
begin
  read(b,x);
  if x<1 then a:=sqr(b)
  else a:=sqr(x);
  write('a=',a);
end.

```

Применение, безусловного и условного операторов с меткой label

```

program ime;
label 1,2;
var a,b,x:real;
begin
  read(b,x);
  if x<1 then goto 1;
  a:=sqr(x); goto 2;
1: a:=sqr(b);
2: write('a=',a);
end.

```

Вопрос №23

Организация цикла с параметрами

Используется оператор **for i:= m_1 to m_2 do m_3**
где i - переменная цикла; m_1, m_2 - начальное и
конечное значения цикла; m_3 - тело цикла.

Пример: Вычислить $y = x^2 / 2x$; $x = 1, 5$; $h = 1$

```
program msx;  
var y,x,h,i:integer;  
begin  
  x:=1; h:=1;  
  for i:=1 to 5 do  
    begin  
      y:=sqr(x)/(2*x);  
      writeln(' y=',y);  
      x:=x+h;  
    end;  
  end.
```

Вопрос №24

Организация цикла с предусловием

Используется оператор: **while** m_1 **do** m_2

где m_1 - логическое выражение; m_2 - тело цикла;

```
program msx;
```

```
var y,x,h:integer;
```

```
begin
```

```
  x:=1;h:=1;
```

```
  while x<= 5  do
```

```
    begin  y:=sqr(x)/(2*x);
```

```
      writeln(' y=',y);
```

```
      x:=x+h;
```

```
    end;
```

```
end.
```

Вопрос №25

Организация цикла с постусловием

Используются операторы **repeat** m_1 и **until** m_2
где m_1 -тело цикла; m_2 - условие окончания цикла.

```
program msx;  
var y,x,h:integer;  
begin  
  x:=1; h:=1;  
  repeat  
    y:=sqr(x)/(2*x);  
    writeln(' y=',y);  
    x:=x+h;  
  until x> 5  
end.
```

Вопрос №26

Построение графических изображений в языке Pascal.

1. **line(x1,y1,x2,y2)** - линия;
2. **circle(x,y,r)** - окружность;
3. **arc(x,y, α 1, α 2,r)** - дуга радиуса;
4. **rectangle(x1,y1,x2,y2)** - прозрачный прямоугольник;
5. **ellipse(x,y, α 1=270°, α 2=90°,xr,yr)** - эллиптическая дуга;
6. **bar(x1,y1,x2,y2)** - залитый цветом прямоугольник;
7. **pieslice(x,y, α 1, α 2,r)** - залитый сектор окружности;
8. **fillellipse(x,y,xr,yr)** - залитый цветом эллипс;
9. **floodfill(x,y,c)** - заполнение области вокруг точки;
10. **setcolor(c)** - установка цвета изображения;
11. **setbkcolor(c)** - установка цвета фона:
0 - чёрный; 1 - синий; 2 - зелёный; 3 - бирюзовый
4 - красный; 5 - малиновый; 6 - коричневый;
7 - светлосерый; 8 – тёмнокоричневый; 9 - яркосиний;
10 - яркозелёный; 11- яркобирюзовый; 12 – яркокрасный;
13 - яркомалиновый; 14- жёлтый; 15 - белый.

Штриховка

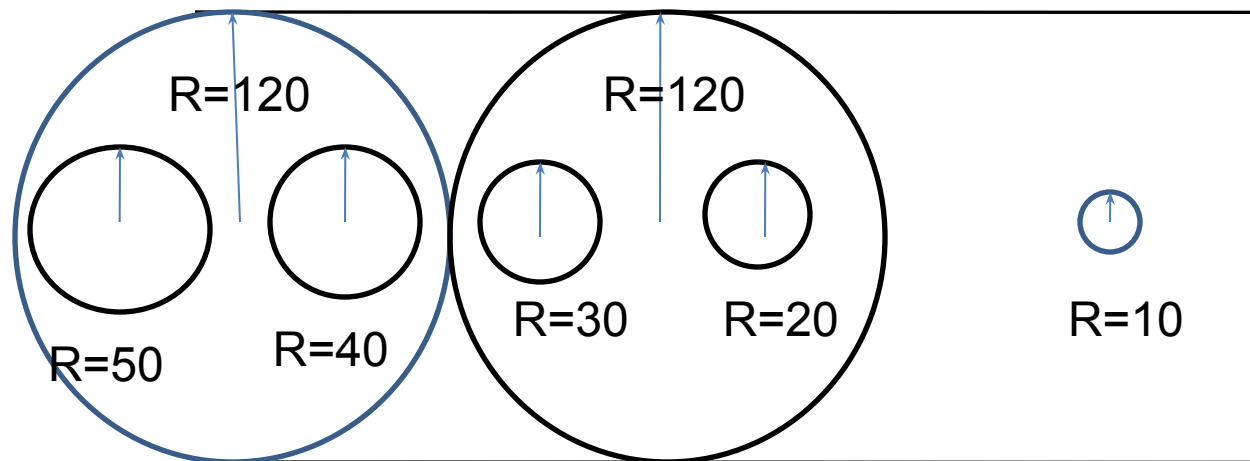
Setfillstyle (тип заливки, цвет штриховой заливки)

- 0 - нет штриховки;
- 1 - сплошная
- 2 - штриховая - - - -;
- 3 - ///тонкая;
- 4 - ////толстая;
- 5 - \\\тонкая;
- 6 - \\\толстая;
- 7 - сетка;
- 8 - штрих-пунктирная;
- 9 - решётка;
- 10 - крупные точки;
- 11 - мелкие точки

Линия штриховки

Setlinestyle (тип линии [0,1], 0, ширина[1,3]);

- | | | |
|-----------------|---------------------|------------------|
| 0 — сплошная; | 0- постоянное число | 1- тонкая линия |
| 1 - прерывистая | | 3- толстая линия |



```
program nnn; uses graph, crt;  
var gd,gm:integer; t:char;  
begin  
    gd:=detect; initgraph(gd,gm,'');  
    setcolor(4); setlinestyle(0,0,3); setfillstyle(6,10);  
    circle(130,175,120); floodfill(130,175,4);  
    setcolor(13); setlinestyle(0,0,3); setfillstyle(7,6);  
    circle(370,175,120); floodfill(370,175,13);  
    setcolor(0);  
    setfillstyle(1,0);pieslice(70,175,0,360,50);  
    setfillstyle(1,1);pieslice(190,175,0,360,40);  
    setfillstyle(1,4);pieslice(310,175,0,360,30);  
    setfillstyle(1,10);pieslice(460,175,0,360,20);  
    setcolor(4); circle(590,175,10);  
    setcolor(14);  
    line(130,55,600,55); line(600,55,600,295); line(130,295,600,295);  
    t:=readkey;closegraph;  
end.
```


Вопрос № 30
Основные понятия электронных таблиц

Электронная таблица - это интерактивная система обработки информации, упорядоченная в виде таблицы с поименованными строками и столбцами.

Возможности электронных таблиц

- создание таблиц с большим количеством строк и столбцов
- оформление таблиц с результатами экспериментальных исследований
- ввод и редактирование в каждой ячейке числа, текста или формулы;
- хранение в памяти ЭВМ таблиц большой размерности
- сортировка данных по возрастанию или убыванию;
- выполнение математических, логических, финансовых, статистических и других операций с табличными данными;
- разнообразное выделение данных в таблице:
- округление всех чисел до одного и того же количества знаков.
- вставка в таблицу изображений и рисунков;
- применение в одном пакете нескольких двумерных таблиц;
- работа в многооконном режиме с несколькими таблицами;
- применение макрокоманд для выполнения последовательности нескольких операций над табличными данными;
- вывод всей таблицы или любой ее части на печать;
- представление табличных данных в виде графиков и диаграмм;
- защита данных от несанкционированного использования.

Вопрос № 31

Базовые элементы структуры электронных таблиц

Таблица - совокупность отдельных рядов, столбцов, клеток или блоков, имеющих конкретные адреса.

- **адрес ряда** образуется из числа, стоящего в его самой левой позиции.
- **адрес столбца** образуется из букв, стоящих в его самой верхней позиции.
- **клетка** стоит на пересечении столбца и ряда. Ее адрес образуется из объединения их адресов.
- **блок** - прямоугольная часть таблицы, адрес которой задается ее верхним левым и нижним правым углами (клетками).
- **диапазон** - общее название для любой прямоугольной области таблицы, начиная от одной клетки и кончая всей таблицей.
- клетка с курсором называется **текущей клеткой**. ряд и столбец с текущей клеткой называются соответственно **текущий ряд** и **текущий столбец**.

Вопрос № 32

Электронная таблица Excel

Экран *Excel* включает в себя следующие основные позиции:

Строка меню - содержит имена команд основных режимов работы программы.

Панель инструментов - содержит определенное количество кнопок, предназначенных для быстрой активизации выполнения определенных команд меню. Поле имени информирует об активной на данный момент ячейке.

Строка формул (ввода) отображает вводимые в ячейку данные. В ней можно просматривать или отредактировать содержимое текущей ячейки. Особенность строки формул это возможность просмотра формулы или функции текущей ячейки, а не ее результата.

Рабочее поле – пространство электронной таблицы, состоящее из ячеек, названий столбцов и строк. Элементы управления включают кнопки навигации, ярлычки листов и полосы прокрутки.

Рабочая книга – это многостраничный документ электронной таблицы, каждая страница которого называется рабочим листом; активная страница отображается в окне документа. В рабочий лист могут входить таблицы, диаграммы, макросы. Все листы рабочей книги сохраняются в одном файле.

Ввод данных осуществляется в текущую ячейку или в строку формул.

Формула может иметь ссылки, т.е. адреса ячеек, содержимое которых используется в вычислениях.

Автоматическое заполнение ячеек листа

Автозавершение - ввод одного столбца рабочего листа текстовых строк, среди которых есть повторяющиеся

Автозаполнение данными - перетаскивание маркера заполнения для операции размножения содержимого ячейки

Автозаполнение формулами - то же самое что и предыдущий пункт, только операции осуществляются с формулами.

Использование стандартных функций осуществляется указанием в формуле - имени функции.

Пункт **Макросы** позволяет реализовать ряд повторяющихся действий одной командой.

Итоговые вычисления предполагают получение сумм, средних значений и других статистических характеристик.

В EXCEL строятся диаграммы различного типа с помощью кнопки «**Мастер диаграмм**».

Вопрос № 55

Структура web-документа на языке HTML

<HTML> - начало документа

<HEAD> начало заголовка документа

<TITLE> начало задания имени документа

Заголовок документа (отображается в строке заголовка)

</TITLE> - конец задания имени документа

</HEAD> конец заголовка документа

<BODY> Начало ввода текста документа

Рабочий блок или тело документа

</BODY> конец ввода текста документа

</HTML> - конец документа

Между тегам **<HEAD>** ... **</HEAD>** заключен раздел заголовков. Он содержит информацию о документе в целом.

Символы, указываемые после тега **<TITLE>**, отображаются в строке заголовка браузера после открытия web-документа.

Основной текст программы пишется после тега **<BODY>**.

Вопрос № 56

Алгоритм создания web-документа

1. Открыть Блокнот.
2. Набрать (скопировать) структуру web-документа.
3. Сохранить документ, присвоив ему расширение htm или html.
4. Закрыть Блокнот.
5. Открыть в браузере созданный web-документ.
6. Из контекстного меню, открываемого щелчком правой кнопки мыши на документе, выбрать команду *Просмотр в виде HTML*.
7. Внести изменения в текст документа. Сохранить его. Обновить содержимое документа в браузере. Продолжить правку.

Вопрос № 57

Работа с графикой в Интернет

При создании web-документов имеется возможность оформлять их с использованием графических объектов.

В случае публикации документа в сети Интернет - объекты должны содержаться в файлах с расширением **jpg**, **gif** или **png**. При подготовке документа на локальном компьютере используются файлы с расширением **bmp**.

Для внедрения рисунка применяется одиночный тег ****. Он всегда должен содержать атрибут **SRC**.

Примеры.

1) **** - вставка в документ из текущего каталога графического объекта, содержащегося в файле **ris.jpg**.

2) **** - вставка в документ из корневого каталога диска **C:** графического объекта, содержащегося в файле **ris.gif**.

Если рисунок по размерам слишком мал или слишком велик, то его размеры корректируются атрибутами **WIDTH** и **HEIGHT**. Первый из них влияет на ширину рисунка, второй – на его высоту. При использовании данных атрибутов необходимо следить за тем, чтобы искажение рисунков не влияло на способность их качественного восприятия. Целесообразно придерживаться отношения 4:3 или 16:9 для горизонтальных и вертикальных сторон.

Пример.

**** - изображение, хранящееся в файле **winter.jpg**, будет иметь 200 пикселей в длину и 150 в высоту.

Некоторые браузеры настроены на работу так, чтобы при загрузке документа не выводить на экран рисунок. Поэтому при внедрении рисунка желательно задавать его краткое описание. Для этого в теге **** применяется атрибут **ALT**. Его формат: **ALT="текст"**. Значение атрибута называют *альтернативным текстом*.

Пример.

**** - при наведении указателя мыши на рисунок, появляется текст «Хороший рисунок зимы».

Для внедрения в web-документ мультимедийных объектов применяется тег **<EMBED>**. Обязательный атрибут этого тега – **SRC**.

Пример.

<EMBED SRC=winter.avi> - вставка в документ из текущего каталога анимационного объекта, содержащегося в файле **winter.avi**.

Рисунок можно использовать в качестве фона страницы. Для этих целей применяется тег **<BODY BACKGROUND=URL>**, где **URL** –адрес к файлу, содержащему рисунок.