

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ СТАВРОПОЛЬСКОГО КРАЯ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
"НЕФТЕКУМСКИЙ РЕГИОНАЛЬНЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ"**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ПО ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ РАБОТ
ПО ДИСЦИПЛИНЕ
ОП. 05 ОСНОВЫ ПРОГРАММИРОВАНИЯ**

для специальности
09.02.03 Программирование в компьютерных системах

2019 г.

ОДОБРЕНО:
НА ЗАСЕДАНИИ ПМО
специальностей
09.02.03 «Программирование в
компьютерных системах»,
09.02.02 «Компьютерные сети» и
профессии 09.01.03 «Мастер по
обработке цифровой информации»
ПРОТОКОЛ №_1_
«28» августа 2019 г.
Руководитель ПМО


 /И.А.Мазяр /

Методические указания составлены в
соответствии с требованиями
Федерального государственного
образовательного стандарта среднего
профессионального образования по
специальности **09.02.03**

**Программирование в
компьютерных системах**

УТВЕРЖДАЮ:

Заместитель директора по учебно-
методической работе

 /Е.С.Шведова /
ПОДПИСЬ (ФИО)

Составитель: Усенко Анна Геннадьевна, преподаватель ГБПОУ НРПК

Рецензенты: Мазяр Ирина Анатольевна, преподаватель ГБПОУ НРПК

Методические указания по выполнению практических работ студентов по дисциплине **«Основы программирования»** разработаны в соответствии с программой специальности среднего профессионального образования (СПО)

09.02.03 Программирование в компьютерных системах

Разработчик:

Усенко А.Г– преподаватель ГБПОУ НРПК

Рассмотрены и одобрены профессионально методическим объединением педагогов специальностей 09.02.03 «Программирование в компьютерных системах», 09.02.02 «Компьютерные сети» и профессии 09.01.03«Мастер по обработке цифровой информации»

Протокол заседания №1 от «28» августа 2019 г.

Содержание

Пояснительная записка	6
Практическая работа №1	7
Практическая работа №2	8
Практическая работа №3	9
Практическая работа №4	10
Практическая работа №5	11
Практическая работа №6	13
Практическая работа №7	14
Практическая работа №8	15
Практическая работа №9	16
Практическая работа №10	17
Практическая работа №11	18
Практическая работа №12	19
Практическая работа №13	20
Практическая работа №14	21
Практическая работа №15	22
Практическая работа №16	23
Практическая работа №17	24
Практическая работа №18	25
Практическая работа №19	26
Практическая работа №20	27
Практическая работа №21	28
Практическая работа №22	29
Практическая работа №23	30
Практическая работа №24	30
Практическая работа №25	30
Практическая работа №26	30
Практическая работа №27	33
Практическая работа №28	34
Практическая работа №29	34
Практическая работа №30	36
Практическая работа №31	37
Практическая работа №32	38

Практическая работа №33	39
Практическая работа №34	41
Практическая работа №35	42
Практическая работа №36	43
Практическая работа №37	44
Практическая работа №38	46
Практическая работа №39	47
Практическая работа №40	49
Практическая работа №41	50
Практическая работа №42	51
Практическая работа №43	52
Практическая работа №44	53
Практическая работа №45	55
Практическая работа №46	58
Практическая работа №47	60
Критерии оценки отчетных работ	62

Пояснительная записка

Практические работы составлены в соответствии с программой учебной дисциплины «**Основы программирования**».

Практические работы способствуют более глубокому усвоению изучаемого теоретического материала, совершенствуют знания обучающихся о разных видах информации, в том числе и цифровой, совершенствуют практические навыки обучающихся в области использования в своей деятельности как учебной, так и профессиональной.

Результатом выполнения практических работ является овладение обучающимися следующими умениями:

- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.

Правила выполнения практических работ:

1. Студент должен выполнить практическую работу самостоятельно.
2. Каждый студент после выполнения работы должен представить отчет о проделанной работе с анализом полученных результатов и выводом по работе.
3. Если студент не выполнил практическую работу или часть работы, то он может выполнить работу или оставшуюся часть во внеурочное время, согласованное с преподавателем.
4. Оценку по практической работе студент получает, с учетом срока выполнения работы, если:
 - работа выполнена правильно и в полном объеме;сделан анализ проделанной работы и вывод по результатам работы; студент может пояснить выполнение любого этапа работы; отчет выполнен в соответствии с требованиями к выполнению работы.
5. Зачет по практическим работам студент получает при условии выполнения всех предусмотренной программой работ, после сдачи отчетов по работам при удовлетворительных оценках за опросы и контрольные вопросы во время практических занятий.

Задача1. Составить программу, выводящую на экран число π .

```
program zadacha1;
Begin
writeln(pi:9:3);
Readln;
End.
```

Задание: Изменяя числа в формате вывода(:9:3, :9:2, :17:3, :17:7, :3:7 и т.д.), просмотрите и проанализируйте полученные результаты.

Задача 2. Заданы два числа. Найти их сумму.

```
program zadacha2;
vara,b,s: integer;
Begin
writeln('введите целое число a ');
read(a);
write('введитецелоечисло b ');
read(b);
s:=a+b;
writeln('s =',s);
Readln;
End.
```

Задание: Поменяйте тип данных integerbyte на Что получается теперь? Установите тип данных longint. Просмотрите суммы.

Задача 3.Составить программу вычисляющую $s = a/b$ и $p=a*b$.

```
program zadacha3;
vara,b,s,p:real;
Begin
writeln('введитечислоa ');
readln(a);
writeln('введитечислоb ');
readln(b);
s:=a/b;
p:=a*b;
writeln('a/b =',s);
writeln('a*b =',p);
Readln;
End.
```

Задание В данном примере ввод и вывод данных для каждой переменной осуществляется отдельно. А можно было записать их и вместе. Подумайте как и запишите свой вариант задачи!

Задача 4. Заданы x, y . Вычислите $S = \sqrt{a} + \sqrt{x+y^2}$

```
program zadacha4;
vara,x,y,s:real;
Beginwriteln('введите число x ');
readln(x);
writeln('введитечисло y ');
readln(y);
writeln('введитечисло a ');
readln(a);
s:=abs(a)+sqrt(x+sqrt(y));
writeln('s =',s);
Readln;End.
```

Задача 5. Задана сторона куба. Вычислить объем куба и площадь боковой поверхности.

```
program zadacha5;
vara,v,s:real;           {описаниепеременных}
Beginwriteln('Введите сторону куба');           {ввод данных}
read(a);
v:=a*a*a;                (*вычислениеобъёма*)
s:=6*a*a;                {вычисление площади поверхности}
                           {вывод результатов}
writeln('Объём = ',v:8:3,' Площадь поверхности = ',s:8:3);
Readln; End.
```

Практическая работа №2 Арифметические операции в Паскале Освоение простейшей структуры программы

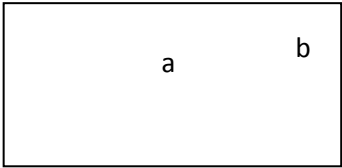
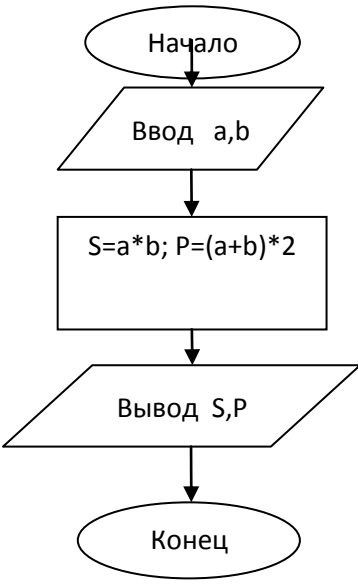
1. Вычислить значение выражения $x=(a+b)-c$
2. Найти произведение 5-ти переменных (a,b,c,d,e)
3. Вычислить значение выражения $y=(a*b)\backslash c$

Методические указания к выполнению пр.раб №2

- Запустить Pascal,
- Определить ко-во переменных, участвующих в задаче.
- Определить тип переменных (если числа целые – то тип INTEGER, Если числа дробные (действительные) – то тип REAL)
- Перед компиляцией программы проверить программу на синтаксис и семантику.

ОБРАЗЕЦ РЕШЕНИЯ ЗАДАЧИ С ПОМОЩЬЮ ЛИНЕЙНОГО АЛГОРИТМА

Задача. По заданным сторонам прямоугольника a и b вычислите его периметр и площадь.

1 этап Математическое решение	2 этап Блок - схема
$P=(a+b)*2; S=a*b$ 	 <pre> graph TD Start([Начало]) --> Input[/Ввод a,b/] Input --> Process[S=a*b; P=(a+b)*2] Process --> Output[/Вывод S,P/] Output --> End([Конец]) </pre>
3 этап На алгоритмическом языке:	4 этап На языке Паскаль:
<pre> алг прямоугольник; цела, b,S,P; нач ввод (a, b,); S=a*b; P=(a+b)*2; вывод(S,P); кон. </pre>	<pre> Program pryamoygolnik; Var a, b,S,P: integer; Begin writeln('Введите стороны прямоугольника'); readln(a, b); S=a*b; P=(a+b)*2; writeln(S,P); Readln; End. </pre>

Практическая работа №3 Реализация алгоритма линейной структуры на языке программирования Pascal. Программирование линейных алгоритмов.

Реализация алгоритма линейной структуры.(3)

Операции div и mod.

Задача 1. Дано трёхзначное число. Чему равны его цифры?

```
program zadacha7;
var a,c1,c2,c3:integer;
Begin
  writeln('введитетрёхзначноечисло '); readln(a);
  c1:=a div 100;
  c2:=(a mod 100) div 10;           { или      c2:=(a div 10) mod 10;}
  c3:=a mod 10;
  writeln('перваяцифра -',c1);
  writeln('вторая цифра -',c2);
  writeln('третья цифра -',c3);
  Readln;
End.
```

Задача 2 . Заданы два целых числа k и d . Используя только арифметические операции, найдите целую и дробную части от деления k на d .

```
program zadacha8;
vara,drobn:real;
      k,d,cel:integer;
Begin  writeln('введитедвачисла');
readln(k,d);
      cel:=kdivd;           {целочисленное деление}
      a:=k/d;               {обычное деление}
      drobn:=a-cel;
      writeln('целаячасть - ',cel);
      writeln('дробная часть - ',drobn);
      Readln;
End.
```

Можно ли было в данной программе все переменные определить как integer или как real? Попробуйте и проанализируйте.

26. Найдите и исправьте ошибки в программе:

```
programzadacha;
vara,b:real; c,e:integer;
Beginwriteln('Введите два целых числа');
readln(a);
c:=a/b;
      d:=a mod b;
      e:=c+d;
      writeln(c);writeln(d);writeln(e);
End.
```

27. Чему равен результат работы программ при $a = 253$:

```
program zadacha2_12a;
vara,x,y,z,b:integer;
Begin
  readln(a);
  x:=a div 100;
  y:=(a div 10) mod 10;
  z:=a mod 10;
  b:=x*100+z;
  writeln(b);
  Readln;
End.
```

```
program zadacha2_12b;
      vara,x,y,z,b:integer;
Begin
  readln(a);
  x:=a div 100;
  y:=(a div 10) mod 10;
  z:=a mod 10;
  b:=y*z;
  writeln(b);
  Readln;
End.
```

Сократите длину записанных выше программ, сохранив их назначение.

Практическая работа №4 Составление программ с помощью линейных алгоритмов. Процедуры вывода. Процедуры ввода.

1. Вывести на экран свою фамилию, имя отчество – одной строкой и любую пословицу – второй строкой.

```
1. Program f_1;  
Begin  
  writeln ('ФИО');  
  writeln ('Пословица');  
End.
```

2. Написать программу на вычисление периметра (P) прямоугольника.

$$P = (a + b) * 2$$

3. Написать программу на вычисление уравнения $x = a^2 + b^3 - c$

```
3. Program g1;  
  Var a,b,c,x:integer;  
Begin  
  Writeln('введите a');  
  readln(a);  
  Writeln(' введите b');  
  readln(b);  
  Writeln(' введите c');  
  readln(c);  
  x:=sqr(a)+sqr(b)*b-c;  
  Writeln('ответ',x);  
  readln;  
End.
```

Практическая работа №5 Использование логических операций при реализации алгоритмов ветвящейся структуры на языке программирования Pascal

1 Дано целое число. Если оно является отрицательным, то прибавить к нему 5; в противном случае прибавить к нему 10. Вывести полученное число.

```
Program if_1;  
Var a:integer;  
Begin  
  Writeln('Введите число');  
  Readln(a);  
  If a<0 then a:=a+5 else a:=a+10;  
  Writeln('ответ ',a);  
  Readln;  
End.
```

2 Дано целое число S. Вывести строку-описание месяцев осеннего сезона, соответствующей числу S(1- «сентябрь», 2 –« октябрь», 3 – «ноябрь». Если S не лежит в диапазоне 1-3, вывести «Память напряги!».

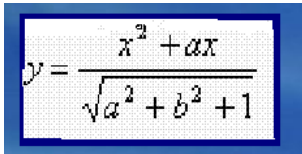
```
Program case_2;  
Var s:integer;  
Begin  
  Writeln('введите оценку');  
  Readln(s);  
  Case s of  
    1:Writeln('сентябрь');  
    2:Writeln('октябрь');  
    3:Writeln('ноябрь');  
    Else Writeln('Память напряги');  
  end;  
  Readln;  
End.
```

3 Дано 2 целых числа. Вывести минимальное из них.

```
Program if_3;  
Var a,b:integer;  
Begin  
  Writeln('Введите число a');  
  Readln(a);  
  Writeln('Введите число b');  
  Readln(b);  
  If a<b then Writeln('мин',a) else Writeln('мин',b);  
  Readln;  
End.
```

Практическая работа №6 Реализация алгоритмов ветвящейся структуры на языке программирования Pascal с использованием оператора выбора

Задача 1 (повторение): Составить программу для вычисления значения функции:


$$y = \frac{x^2 + ax}{\sqrt{a^2 + b^2 + 1}}$$

```
Program My_Program_1;
Var
x,a,b:integer;
y:real;
Begin
Write('Введите значение x:');
Readln(x);
Write('Введите значение a:');
Readln(a);
Write('Введите значение b:');
Readln(b);
y:=(sqr(x)+a*x)/(sqrt(sqr(a)+sqr(b)+1));
Writeln('Значение функции y=',y);
End.
```

2. Дано целое число. Если оно является отрицательным, то прибавить к нему 12; в противном случае отнять от него 11. Вывести полученное число.

```
Program if_1;
Var a:integer;
Begin
Writeln('Введите число');
Readln(a);
If a<0 then a:=a+12 else a:=a-11;
Writeln('ответ ',a);
Readln;
End.
```

3 Дано целое число M. Вывести строку-описание сезонов года, соответствующей числу M (1-«зима», 2-«весна», 3-«лето», 4-«осень». Если M не лежит в диапазоне 1-4, вывести «Ошибка!».

```
Program case_2;
Var
M:integer;
Begin
Writeln('введите оценку');
Readln(s);
Case M of
1:Writeln('зима');
```

```
2:Writeln('весна');
3:Writeln('лето');
3:Writeln('осень');
Else Writeln('Ошибка');
end;
Readln;
End.
```

4. Дано 2 целых числа. Вывести большее из них.

```
Program if_3;
Var a,b:integer;
Begin
Writeln('Введите число a');
Readln(a);
Writeln('Введите число b');
Readln(b);
If a>b then Writeln('макс',a) else Writeln('макс',b);
Readln;
End.
```

Задача 5. Даны три числа x, y, z. Найти максимальное число из них.

```
program zadacha2;
var max,x,y,z:real;
Begin
writeln(' Введите три числа ');
readln(x,y,z);
if x>y then max:=x else max:=y;
if z>max then max:=z;
writeln('большее из трёх чисел =',max);
readln;
End.
```

Задача 6. Задано число a. Кратно ли оно 3.

```
program zadacha3;
var a:longint;
Begin
writeln(' Введите число');
readln(a);
if a mod 3 =0 then writeln(a,'кратно3')
else writeln(a,'некратно3');
readln;
End.
```

Практическая работа №7 Разработка программ циклической структуры с использованием оператора цикла с параметром

Задача 1 .(Пример)(счётный цикл FOR) Найти сумму всех натуральных чисел от 1 до n.

```
program zadacha1_1;  
var i,n,s:integer;  
Beginwriteln(' введите натуральное n');  
readln(n);  
s:=0;  
for i:=1 to n do  
s:=s+i;  
writeln('суммаот 1 до',n,' = ',s);  
Readln;  
End.
```

1.2 ЗаданиеНайти сумму всех натуральных чисел от 1 до n и прибавить к получившейся сумме(s) 100
!!!Подсказка (не забудьте в своей задаче объявить еще одну переменную)

1.3 Задание. Найти произведение всех натуральных чисел от 1 до n и отнять от получившегося произведения число 3.

Задача 2.

2.1-Чему равно число b, если выполнять деление числа b на 2 до тех пор пока b<10?

(Ответ запишите в тетради)

2.2-Чему равно число b, если выполнять деление числа b на 2 до тех пор пока b<3?

(Ответ запишите в тетради)

2.3-Чему равно число b, если выполнять деление числа b на 2 до тех пор пока b<1?

(Ответ запишите в тетради)

```
Programtest2;  
Var b:Real;  
Begin  
b:=100;  
Repeat b:=b/2;  
Until b<10;  
Writeln(b:0:2);  
End.
```

Задача 3 (Пример) Программа вывода на печать 5-ти звездочек

```
Program test3;  
Var j:Integer;  
Begin  
For j:=1 to 5 Do  
Writeln('*');  
Readln;  
End.
```

3.1 Создайте такую же программу, но чтобы она печатала 10 раз ваше имя!

***** 3.2** Создайте подобную программу, но чтобы она сначала запрашивала ЧИСЛО РАЗ для вывода фразы УРА!!!

! Подсказка (Введите дополнительную переменную в раздел VAR и не забудьте об операторах Writeln и Readln)

Практическая работа №8 Разработка программ циклической структуры с использованием оператора цикла с предусловием

Пример 1 Напишите программу «Вводить числа с клавиатуры, пока не будет введено число 0 (Цикл с предусловием)

```
Program chisla;  
Var i:integer;  
Begin  
Writeln(' введите число');  
Readln(i);  
While i<>0 do  
Begin  
Writeln(' введите число');  
Readln(i);  
End.
```

Задание 1 Напишите программу «Вводить числа с клавиатуры, пока не будет введено число 5 (Цикл с предусловием)

ПРИМЕР 2. Найти сумму всех натуральных чисел от 1 до n.

```
program zadacha1_2;  
var i,n,s:integer;  
Begin  
writeln('введите натуральное n');  
readln(n);  
s:=0; i:=1;  
while i<=n do  
begin  
s:=s+i;  
i:=i+1;  
end;  
writeln('сумма от 1 до',n,'=',s);  
Readln;  
End.
```

Задание 2 Напишите ответ, чему будет равна сумма при $n=3$; $n=10$; $n=-10$?

3. Задача.

Дано N кубиков, на которых написаны разные буквы. Сколько различных N-буквенных слов можно составить из этих кубиков. (Слова не обязательно должны иметь смысл).

```
Program Words;  
var F,N,R:integer;  
begin  
writeln('Введите число букв');  
readln(N);  
F:=1; R:=1;  
while R<=N do  
begin  
F:=F*R; R:=R+1;  
end;  
writeln('Из',N,'букв можно составить',F,'слов');  
readln;  
end.
```

Задание 3 Напишите ответ сколько буквенных слов можно составить из 56 букв; 7 букв; 11 букв

Практическая работа №9 Процедуры и функции в языке программирования Pascal. Решение задач с помощью процедур и функций.

1 Программа, в которой используется процедура, с двумя параметрами, для сложения двух чисел.

```
VAR                                {описание переменных для основной программы}
a,x:integer;
PROCEDURE summa (vara,b:integer); {summa - имя; a,b - параметры}
VAR
c:integer;
BEGIN
c:=a+b;
writeln('Сумма чисел = ', c);
END;                                {конец описания процедуры}

BEGIN                                {начало основной программы}
a:=15;
x:=20;
summa(a,x);                        {вызов процедуры}
END.
```

Задание 1!!! Напишите программу для использования процедуры с 3 параметрами, для умножения трех чисел

Задача 2. Программа, в которой используется функция, которая возвращает значение: TRUE или FALSE, вот такого выражения - $(4 > 12) < (5 = x)$

```
FUNCTION proverka: boolean; {начало описания функции}
VAR
x:integer;
BEGIN
write('Введите число x: ');
readln(x);
proverka:=(4>12)<(5=x);
END;                                {конец описания функции}

BEGIN
writeln(proverka);                {вызов функции из основной программы}
END.
```

Задание 2!!! Напишите программу в которой используется функция которая возвращает значение: TRUE или FALSE, вот такого выражения $(6 > 12) < (7 = x)$

Задание 3. Напишите программу для использования процедуры с 4 параметрами, для такого выражения $c = (r1 + R2) * (t1 + t2)$

Задание 4. Напишите программу для использования процедуры с 5 параметрами, для такого выражения $c = (r1 + R2) * (t1 + t2) / h$

Практическая работа №10
Организация ввода-вывода стандартных типов данных.

1. Пример. Для целого числа k от 1 до 99 напечатать фразу "мне k лет", учитывая при этом, что при некоторых значениях k слово "лет" заменить на слово "год" или "года"

```
Program how_old_year;
Const t=10;
Var a,b:Integer;
k:1..99;
Begin
Readln(k);
  If (k>=5) and (k<=19) then Writeln(k,'лет');
a:=k div t;
b:=k-a*t;
  If (b>=5) and (b<=9) then Writeln(k,'лет');
  If b=0 then Write(k,'лет');
  If (b>=2) and (b<=4) then Writeln(k,'года');
  If b=1 then Write(k,'год');
End.
```

Задание 1. Для целого числа E от 1 до 20 напечатать фразу "У меня E долларов"

Пример2. Ввести 2 числа в диапазоне от 1 до 20000.Найти их сумму и частное.

```
Program P20;
typePozitiv=1..20000;
var x, y, z : Pozitiv;
begin
writeln('Introducerinumerelepozitive x, y:');
readln(x,y);
writeln('x=', x);
writeln('y=', y);
z:=x+y; writeln('x+y=', z);
z:=x-y; writeln('x-y=', z);
z:=x*y; writeln('x*y=', z);
z:=x mod y; writeln('x mod y=', z);
z:=x div y; writeln('x div y=', z);
end.
```

Задание 2 Ввести 2 числа(x и y) в диапазоне от 1 до 10000.Найти целочисленное деление (C)

Задание 3. Ввести 3 числа в диапазоне от 1 до 32000.Найти их произведение и остаток от деления

Пример 1. Задан одномерный массив $B(10)$, заполненный произвольным образом.

Подсчитать количество элементов массива, больших заданного числа k .

```
Programmassiv;
Uses crt;
Var b:array [1..10] of integer;
    I, k, s : integer;
Begin
Clrscr; S:=0;
  For i:=1 to 10 do
    Begin
Write('Введите', i, '-й элемент массива ');
Readln (B[i]);
    End;
Write('Введите число k');
Readln(k);
  For i:=1 to 10 do
    If b[i]>k then s:=s+1;
  Write('Количество элементов', s);
End.
```

Задача 1. Задан одномерный массив $B(5)$, заполненный произвольным образом.

Подсчитать количество элементов массива, меньших заданного числа k .

2.Пример. Ввести массив из 10 элементов, найти произведение элементов массива.

```
Program mas_1;
Var a:array[1..10] of integer;
I:integer;
P:longint;
Begin
  For i:=1 to 10 do
    Begin
Write('Введите ',i,' элемент массива ');
Read(a[i]);
    End;
  P:=1;
  For i:=1 to 10 do p:=p*a[i];
  Writeln('произв-е =',p);
  Readln;
End.
```

Задача 2 Ввести массив из 7 элементов, найти сумму элементов массива.

Задача 3. Ввести массив из 5-ти элементов, найти сумму элементов массива и отнять от нее $S=15$.

Задача 4. Ввести массив из 4-ти элементов, найти сумму элементов массива и разделить ее на $S=10$.

5 ЗАДАЧА Дан одномерный массив $A[N]$, $a[1]$, $a[2]$, ..., $a[n]$ - целые. Заменить все большие 7 члены массива числом 7. Найти количество таких членов.

```
Program mas_1;
Uses crt;
Const n=10; (кол-во эл.массива)
Var
a:array[1..n] of integer;
i,count:integer;
BEGIN
ClrScr;
count:=0; (кол-во замен)
Randomize;
for i:=1 to n do
begin
a[i]:=Random(20); (Случ.выбор эл-ов массива)
Write(a[i]:4);
end;
Writeln;
for i:=1 to n do
begin
if a[i]>7 проверка
then
begin
a[i]:=7; то замена эл-та на 7
count:=count+1; счетчик заменув-ся
end;
Write(a[i]:4);
end;
Writeln;
Writeln('кол-во замен ',count);
END.
```

6.Ввести массив из n элементов, найти среднее арифметическое элементов массива.

```
Program mas_2;
Var a:array[1..1000] of integer;
n,I:integer;
P:longint;
s:real;
Begin
Writeln('введите кол-во эл-тов массива '); Readln(n);
For i:=1 to n do
Begin
Writeln('укажите эл-т массива '); Readln(a[i]);
End;
s:=0;
For i:=1 to n do s:=s+a[i];
Writeln('среднее арифметическое =',s/n:4:4);
Readln;
End.
```

1Пример. Дан массив из n элементов. Упорядочить массив по возрастанию.

```
Programmas_12;
uses crt;
Var a:array[1..1000] of integer;
k,n,I,j:integer;
Begin
Writeln('введите кол-во элмассива '); Readln(n);
randomize;
For i:=1 to n do
Begin
a[i]:=random(100)-50; write(a[i]:4);
End;
For i:=1 to n-1 do
For j:=i+1 to n do
If a[i]>a[j] then
begin
k:=a[i];
a[i]:=a[j];
a[j]:=k;
end;
Writeln;
For I :=1 to n do Write (a[i]:3);
Readln;
End.
```

1.1Задание: Дан массив из n элементов. Упорядочить массив по убыванию.

2. Пример Дан массив А из 100 элементов, состоящий из нулей и 1. Заменить 1 – нулями и наоборот.

```
Program mas_17;
Uses Crt;
Vara : array[1..100] of byte;
i : byte;
begin
ClrScr;
Randomize;
for i:=1 to 100 do
begin
a[i]:=Random(2);
Write(a[i]);
end;
Writeln;
for i:=1 to 100 do
begin
a[i]:=1-a[i];
Write(a[i]);
end;
Readln;
End.
```

2.1 Задание Дан массив А из 10 элементов, состоящий из 0 и 1. Заменить нули – 3-ми ,а. 1-цы – 4-ми

3. Ввести массив из 10 элементов, найти произведение элементов массива

4. Ввести массив из n элементов, найти среднее арифметическое элементов массива и прибавить к нему число К.

Практическая работа №13

Работа с одномерными массивами на языке ПАСКАЛЬ. Алгоритмы поиска

Пример 1. Определить самую высокую температуру и самый теплый день в мае.

```
Program massiv;
Uses crt;
Var t:array [1..5] of integer; - массив температур 5 эл-ов
    I, max, n : integer; (I – счетчик цикла, макс-
    максзнач , n-№ дня)
Begin
  Clrscr;
  For i:=1 to 5 do (счетчик цикла)
    Begin
      Write('Введите темп в ', I, '-й день');
      Readln (t[i]);
    End;
  Max:=t[1]; n:=1; (начн макс темп и № дня)
  For i:=2 to 5 do (счетчик увел на 1 т к надо
  начинать проверку со 2 дня)
  Begin
    If t[i] > max then
      Begin
        max:=t[i]; n:=i; (если найдено наиб знач, то
        меняется максимум и запомин в какой день )
      End;
  End;
  Write('макс темп ', max, ' в ', n, ' день');
  End.
```

Задача 1. Определите самую низкую температуру и самый холодный день в январе. Массив состоит из 15 эл-ов.

Пример 2. Ввести массив из n элементов, найти среднее арифметическое элементов массива.

```
Program mas_2;
Var a:array[1..100] of integer;
n, I:integer;
P:longint;
s:real;
Begin
  Writeln('введите кол-во эл-тов массива '); Readln(n);
```

```
For i:=1 to n do
  Begin
    Writeln('укажите эл-т массива '); Readln(a[i]);
  End;
  s:=0;
  For i:=1 to n do s:=s+a[i];
  Writeln('среднее арифметическое =', s/n);
  Readln;
End.
```

Задача 2. Ввести массив из n элементов, найти среднее арифметическое элементов массива и прибавить к нему целое число r = 10.

Пример 3. Программа на нахождение номера максимального эл-та и его значения.

```
program index_max_elem;
const n = 10;
var
  a : array [1 .. n] of integer; { массив }
  i : integer; { номер текущего
  элемента }
  imax : integer; { номер максимального
  элемента }
begin
  writeln('Введите ', n, ' элементов массива');
  for i := 1 to n do read(a[i]);
  imax := 1;
  for i := 2 to n do
    if a[i] > a[imax] then imax := i;
  writeln('Номер максимального элемента: ', imax);
  writeln('Максимальный элемент: ', a[imax]);
end.
```

Задача 3. Напишите программу на нахождение минимального элемента и его значения, и умножьте найденное значение на t=100. (Задайте const n = 5).

Пример 1. Дано целое число N ($32 \leq N \leq 126$). Вывести символ с кодом, равным N .

```
Program stroka2;  
Var a:byte;  
b:char;  
begin  
  writeln ('введите код символа- число больше 31 и меньше 127');  
  readln(a);  
  b:=Chr(a);  
  writeln(b);  
  readln;  
end.
```

Задача 1. Дано целое число V ($45 \leq V \leq 100$). Вывести символ с кодом, равным V .
Что будет выдавать программа, если вы введете число:

- 1) «1»
- 2) «89»
- 3) «99»

Напишите ответ в тетради!

Пример 2 *Строка содержит английский текст. Подсчитать количество слов, начинающихся на букву b.*

```
program string_8;  
var s:string;  
i,k:integer;  
begin write('введите строку:');  
  readln(s);  
  s:=' '+s;k:=0;  
  for i:=1 to length(s) do  
    if (s[i]=' ') and (s[i+1]='b') then k:=k+1;  
  writeln(k);  
  Readln;  
end.
```

Задача 2. *Строка содержит русский текст. Подсчитать количество слов, начинающихся на букву А.*

1 Пример. Ввести двумерный массив из $n \times m$ ($n=2$; $m=4$) элементов, найти среднее арифметическое элементов массива.

```
Program dmas_2;
const n=2;m=4;
Var a:array[1..n,1..m] of integer;
I,j:integer;
s:real;
Begin
For i:=1 to n do
For j:=1 to m do
Begin
Writeln('Введите элемент массива ');
Readln(a[i,j]);
End;
s:=0;
For i:=1 to n do
For j:=1 to m do
s:=s+a[i,j];
Writeln('Среднеарифм =' ,s/(n+m));
Readln;
End.
```

1 Задача Ввести двумерный массив из $n \times m$ ($n=2$; $m=2$) элементов, найти сумму элементов массива.

2 Пример Дан массив из $n \times m$ элементов. ($n=3$; $m=4$) Найти сумму элементов каждой строки.

```
Program dmas_7;
uses crt;
const n=3;m=4;
Var a:array[1..n,1..m] of integer;
s,i,j:integer;
Begin
Randomize;
For i:=1 to n do
begin
For j:=1 to m do
begin
a[i,j]:=random(10);
write(a[i,j]:4);
end;
writeln;
end;
For i:=1 to n do
Begin
s:=0;
For j:=1 to m do
s:=s+a[i,j];
writeln ('суммаэл-ов ',i,' строки=',s);
end;
readln;
end.
```

2 Задача Дан массив из $n \times m$ элементов. ($n=2$; $m=3$) Найти сумму элементов каждой строки.

3 Задача Дан массив из $n \times m$ элементов. ($n=2$; $m=2$) Найти сумму элементов каждого столбца.

4 Задача Дан массив из $n \times m$ элементов. ($n=2$; $m=4$) Найти сумму элементов каждого столбца.

Генератор случайных чисел random задать в размере (100) и расстояние между столбцами задать в размере (6)

Пример 1. Заполнить множество A путем ввода n значений:

```
program rtr_1;
const  n=20;
var    A: set of 1 ..200;
j, x : byte;
begin  A:=[];
for j := 1 to n do
begin
readln(x);
A:=A+[x]
end;
for x := 1 to 200 do
if x in A then writeln(x);
end.
```

Задание 1.1 Заполнить множество A путем ввода n значений: Где n=10; а массив A от 1 до 9.

Пример 2. Описать множество M(1..50). Сделать его пустым. Вводя целые числа с клавиатуры, заполнить множество 5 элементами.

ЧИТАТЬ Пояснение: В разделе описания переменных опишем множество целых чисел от 1 до 50, переменную X целого типа будем использовать для считывания числа-кандидата в множество, целую переменную i используем для подсчета количества введенных чисел.

В начале программы применим операцию инициализации множества M:=[] , так как оно не имеет элементов и является пустым.

Заполнение множества элементами произведем с использованием оператора цикла Repeat, параметр которого i будет указывать порядковый номер вводимого элемента. Операцию заполнения множества запишем оператором присваивания M:=M+[X]. Контроль заполнения множества запишем с использованием операции проверки принадлежности in. Если условие X in M выполняется, выведем сообщение о том, что число X помещено в множество.

```
Program InputMno;
Var
M : set of 1..50;
X, i : integer;
Begin
M := [ ];
i :=1;
repeat
write('введите 'i,'-й Эл-т множества');
readln(X);
if (X in M)
then
begin
write('X', 'уже содержится в множестве');
i := i-1;
end
else
begin
write('X', ' помещен в множество');
M := M+[X];
end;
i := i+1;
until i>5;
End.
```

Задание 2.1 Создать множество K (1..30) Сделать его пустым. Вводя целые числа с клавиатуры, заполнить множество 3-мя элементами.

Пример 3. . Опишите множества M1(1, 2) и M2(2, 1). Сравните эти множества на равенство.

```
Program InputMno;
Var
M : set of 1..10;
M2 : set of 1..10;
Begin
M := [1,2 ];
M2:= [2,2 ];
if (M = M2)
then
write('True')
else
write('False');
End.
```

Задание 3.1 Опишите множества M1(1, 2, 3) и M2(1, 2, 3, 4).

Сравните эти множества по операции <=.

Пример1 Дан текст на русском языке. Найти количество гласных букв

```
Programmn_1;
uses crt;
type mn= set of char;
const glas= ['a','e','и','o','y','ы','э','ю','я'];
var a:string;
k,i,s:integer;
begin
  clrscr;
  writeln('введите текст');
  readln(a);
  s:=length(a);
  i:=0;
  if a[s]<>'.' then a:=a+'.';
  repeat
  begin
    inc(i);
    if a[i] in glas then
      k:=k+1;
    end;
  until a[i]='.';
  writeln (k);
  readln;
end.
```

Задача 1. 2. Дан текст на русском языке. Найти количество согласных букв «в,ф,м,р,т»

Пример 2.

Ввести информацию по 3-м студентам (порядковый номер, Ф.И.О. и три оценки) определить количество отличников и средний бал каждого студента.

```
Program zap_1;
Type student = record
  n : integer ;
  fio : string[40] ;
  bal : array [1..3] of integer
end;
Var spisok : student ;
i, j, kol, s : integer; sr : real;
Begin
  kol:=0;
  With spisok do
  For i:=1 to 3 do
  begin
    n:=i;
    Write ( ' введите фамилию ', i , ' ');
    Readln (fio);
    s:=0;
    For j:= 1 to 3 do
    begin
      write ( 'введите оценку: ' );
      readln ( bal [j] );
      s := s+ bal [j];
    end;
    if s=15 then kol:=kol+1;
    sr := s/3; writeln ( fio, ', средний бал = ', sr:4:1); end;
    writeln( ' количество отличников= ', kol );
    readln;end.
```

Задача 2.1 Ввести информацию по 4-м ученикам (порядковый номер, Ф.И.О. и четыре оценки) определить количество хорошистов и средний бал каждого ученика.

Практическая работа №18 Работа с типизированными файлами на языке ПАСКАЛЬ

1. Составить программу, которая создает типизированный файл, состоящий из 50 случайных чисел целого типа в диапазоне от –100 до 100. Найти максимальное среди этих чисел.

```
program zadacha_1;
uses crt;
var
  f: file of integer; {файловая переменная}
  a: array [1..50] of integer;
  i, n, max: integer;
  st: string;
begin
  clrscr;
  writeln('введите имя файла');
  readln(st);
  assign(f, st);
  rewrite(f);
  for i:=1 to 20 do
  begin
    a[i]:=-random(110)+67;
    writeln('a[' , i, ']=', a[i]);
    write(f, a[i]);
  end;
  readln;
  close(f);
  {$i-}
  reset(f);
  {$i+}
  max:=a[1];
  for i:=1 to 20 do
  begin
    read(f, a[i]);
    if a[i]> max then max:=a[i];
  end;
  writeln('max=', max);
  readln;
  close(f);
end.
```

1.1 Составить программу, которая создает типизированный файл, состоящий из 30 случайных чисел целого типа в диапазоне от –56 до 70. Найти минимальное среди этих чисел.

2. Создать файл и записать в него десять случайных чисел от –100 до 100.

```
program zadacha_9;
uses crt;
var
  f: file of integer;
  a: array [1..10] of integer;
  i, n: integer;
begin
  clrscr;
  assign(f, 'row');
  rewrite(f);
```

```
for i:=1 to 10 do
```

```
begin
```

```
  a[i]:=-random(100)+65;
```

```
  write('a[' , i, ']=', a[i]);
```

```
  readln;
```

```
  write(f, a[i]);
```

```
end;
```

```
close(f);
```

```
end.
```

2.1 Создать файл и записать в него 5 случайных чисел от –10 до 15.

1. Создать текстовый файл и записать в него пять одинаковых фраз.

```
program zadacha_1;
uses crt;
var
f: text; {файловая переменная}
i: integer; {переменная счетчик}
s: string; {имя файла}
fr: string; {фраза}
begin
clrscr;
writeln('Введите имя файла');
readln(s);
assign(f,s);
rewrite(f);
fr:='(Hello! Goodbye!);
for i:=1 to 5 do
begin
writeln(f,fr);
end;
readln;
close(f);
end.
```

!!!- После создания файла. Сохраните на раб.столе программу под именем **program 1**, затем снова откройте программу с раб стола и запустите на выполнение. Закройте снова программу.

- На раб.столе должен появиться ваш файл – Откройте его с помощью Программы «Блокнот»

2. Создайте текстовый файл. Запишите в него фразу "Здравствуй мир!"

```
program zadacha_2;
uses crt;
var
f: text;
st: string;
begin
clrscr;
writeln('Введите имя файла: ');
readln(st);
assign(f,st);
rewrite(f);
write(f,'Здравствуй, мир!');
close(f);
end.
```

!!!- После создания файла. Сохраните на раб.столе программу под именем **program 2**, затем снова откройте программу с раб стола и запустите на выполнение. Закройте снова программу.

- На раб.столе должен появиться ваш файл – Откройте его с помощью Программы «Блокнот»

Практическая работа №20 Работа с модулями на языке ПАСКАЛЬ.

1. Пример.

```
usesGraphABC,Events;
procedureCloseWin;
var i: integer;
begin
for i:=1 to 20 do
begin
TextOut(10,20*i,'Не хочу закрываться!');
Sleep(100);
end;
end;

begin
OnClose:=CloseWin;
end.
```

1.1. Задача . Измените предложение на «Нажми еще раз!». Поменяйте скорость – на более медленную. Поменяйте счетчик выполнения написания предложений до 5 раз.

2. Пример.

```
usesTimers,Events;
procedure f;
begin
write(1);
end;
var t: integer;
begin
t:=CreateTimer(500,f);
end.
```

2.1 Задача. Измените печатаемую цифру. Поменяйте скорость на более медленную.

3 Пример.

```
usesGraphABC;

varw,h,i,j,pic: integer;

begin
SetWindowCaption('Обои');
pic:=LoadPicture('demo.bmp');
w:=PictureWidth(pic);
h:=PictureHeight(pic);
for i:=0 to WindowWidth div w do
for j:=0 to WindowHeight div h do
DrawPicture(pic,i*w,j*h);
end.
```

3.1 Задача. Создайтеобоиизрисунков: 1- ball1.png; 2-ball2.png; 3- apple.bmp ; 4-target.jpg

1. Задача. Определить «Что демонстрирует программа»

```
uses GraphABC, PointRect;
var
r: Rect;
i: integer;
begin
SetWindowSize(700,520);
SetBrushStyle(bsClear);
r:=RectF(100,10,600,510);
for i:=1 to 60 do
begin
InflateRect(r,-4);
Rectangle(r.Left,r.Top,r.Right,r.Bottom);
Sleep(1);
end;
InflateRect(r,-2);
for i:=1 to 59 do
begin
InflateRect(r,4);
Rectangle(r.Left,r.Top,r.Right,r.Bottom);
Sleep(1);
end;
end.
```

2. Задача. Определить «Что демонстрирует программа»

```
uses GraphABC;
var i: integer;
begin
LockDrawing;
for i:=1 to 500 do
begin
ClearWindow;
SetBrushColor(clGreen);
Ellipse(i,100,i+100,200);
Redraw;
Sleep(1);
end;
end.
```

Задача № Определить «Что демонстрирует программа»

```
uses GraphABC;
const speed=1;
var
w,h,i,j,pic: integer;
begin
SetWindowCaption('Перемещение картинки');
pic:=LoadPicture('demo.bmp');
w:=PictureWidth(pic);
h:=PictureHeight(pic);
SetBrushColor(clWhite);
for i:=0 to WindowWidth-w do
begin
DrawPicture(pic,i,0);
if i mod speed = 0 then Sleep(1);
FillRect(i,0,i+1,0+h);
end;
DestroyPicture(pic);
end.
```

Практическая работа №22 Работа с графическими модулями.Создание фоновых обоев.
Работа с модулем GRAPHABC

```
Пример1.programkrug;  
usesgraphabc;  
varx,y,r:integer;  
begin  
setwindowsize (640,480);  
setwindowtitle ('Круг');  
setpencolor (clgreen);  
setpenwidth (4);  
x:=50;  
y:=50;  
r:=40;  
circle(x,y,r);  
end.
```

Задание 1.1:Измените алгоритм примера 1 так, чтобы в результате его выполнения было нарисовано 10 кругов по горизонтали.
Ответ:

Задание 1.2. Измените алгоритм примера 2 таким образом, чтобы в результате его выполнения рисовалось несколько красных (clred) окружностей по диагонали.

Пример 2 Дан алгоритм, в результате которого будет нарисован квадрат.

```
programkvadrat;  
usesgraphabc;  
varx,y:integer;  
begin  
setwindowsize (840,680);  
setwindowtitle ('Квадрат');  
setpencolor (clred);  
setpenwidth (4);  
x:=30;  
y:=30;  
rectangle(x,y,x+40,y+40);  
end.
```

Задание 2.1

Составьте алгоритм рисующий 10 квадратов по горизонтали.

Задание 3. Дан алгоритм, в результате которого будет нарисован дом. Размножьте рисунок в любом направлении: по вертикали, по горизонтали, по диагонали

```
programdom;  
usesgraphabc;  
varx,y:integer;  
begin  
setwindowsize (840,680);  
setwindowtitle ('Дом');  
setpenwidth (2);  
x:=30;  
y:=30;  
begin  
setpencolor (clblue);  
rectangle(x,y+30,x+140,y+140); floodfill(x+12,y+32, clblue);  
line(x-5,y+32,x+70,y);  
line(x+70,y , x+145,y+32); floodfill(x+70,y+20, clnavy);  
setpencolor (clgreen);  
rectangle(x+20,y+60,x+60,y+120); floodfill(x+25,y+65, claqua);
```



```

setpencolor (clred);
rectangle(x+80,y+60,x+120,y+100); floodfill(x+85,y+65, clpurple);
end;
end.

```

Задание 4 Заполните пропуски в программе, в результате выполнения которой будет нарисовано 5 прямоугольников и 5 окружностей.

```

Program figury;
uses graphabc;
var x,y,i:_____ ;
begin
  setwindowsize (940,680);
  setwindowtitle ('Фигуры');
  setpenwidth (2);
  x:=10;
  y:=10;
  _____ i:=1 to _____ do
  begin
    _____(x+10,y+10,x+80,y+40); floodfill(x+15,y+15, clfuchsia);
    _____(x+50,y+90,20); floodfill(x+50,y+90, clteal);
    x:=x+100;
  end;
end.

```

Задание 5 Дан алгоритм, в результате которого будет нарисована улитка. Размножьте рисунок улитки в любом направлении: по вертикали, по горизонтали, по диагонали.

```

program ulitka;
uses graphabc;
var x,y,i:integer;
begin
  setwindowsize (940,680);
  setwindowtitle ('Улитка');
  setpenwidth (2);
  x:=100;
  y:=100;

  begin
    setpencolor (clblack);
    circle(x,y,50); floodfill(x,y, clbrown);
    circle(x,y,35); floodfill(x,y, clolive);
    circle(x,y,20); floodfill(x,y, clgray);
    circle(x+90,y-10,20); floodfill(x+90,y-10, clgray);
    circle(x+80,y-15,3); floodfill(x+80,y-15, clblack);
    circle(x+100,y-15,3); floodfill(x+100,y-15, clblack);
    line(x+85,y,x+95,y);
    line(x+90,y-30 , x+105,y-45);
    line(x+90,y-30 , x+75,y-45);
    line(x+5,y+50,x+90,y+10);
    line(x+45,y+50,x+90,y+10);
    line(x+5,y+50,x+45,y+50);
    floodfill(x+34,y+47, clbrown);
  end;

end.

```



Практическая работа №23 Решение задач с линейной структурой

1. Курс евро на сегодняшний день $1\text{€} = 89$ руб. у Вас в кармане N рублей, рассчитайте сколько евро вы можете купить.
2. Найти стоимость покупки, состоящей из N килограммов мандарин, M бутылок шампанского и S шоколадок. Цены вводятся с клавиатуры.
3. Задача. Найти сумму цифр двухзначного числа. (Подсказка -при нахождении первого числа используется DIV , а при нахождении 2 числа MOD)
4. Дано трехзначное число. Нужно вычислить квадрат суммы всех цифр.
5. Человеку R лет. По предсказанию этот человек проживет до 100 лет. Сколько ему осталось жить Лет? Месяцев? Дней? Часов? Минут? Секунд? (Високосные годы не учитывать)

Практическая работа №24 Решение задач с простым и сложным условием.

1. Дано целое число. Если оно является положительным, то прибавить к нему 4; в противном случае не изменять его. Вывести полученное число.
2. Дано целое число. Если оно является положительным, то прибавить к нему 10; в противном случае вычесть из него 8. Вывести полученное число.
Решение задачи
3. Дано 3 целых числа. Найти количество положительных в исходном наборе.
Решение задачи
4. Составить программу, проверяющую верно ли утверждение, что сумма цифр введенного двухзначного целого числа делится без остатка на 5.

Практическая работа №25 Решение задач с условием

1. Дано 2 целых числа. Вывести большее из них.
Решение задачи
2. Дано 2 целых числа. Вывести порядковый номер меньшего из них.
Решение задачи
3. Дано число. Определить, оканчивается ли данное число на цифру 7.
Решение задачи
4. Дано число. Определить, является ли данное число четным или нечетным.

Практическая работа №26 Решение задач с выбором.

1. Ввести возраст от 1 до 99. В зависимости от введенного возраста вывести фразу «Мне...лет(год, года)»
2. Ввести номер месяца. Определить по введенному номеру – квартал.

Практическая работа № 23,24 (ответы)

1. Дано целое число. Если оно является положительным, то прибавить к нему 4; в противном случае не изменять его. Вывести полученное число.

```
Program if_1;
Var a:integer;
Begin
  Writeln('Введите число');
  Readln(a);
  If a>0 then a:=a+4;
  Writeln('Полученное число ',a);
  Readln;
End
```

2. Дано целое число. Если оно является положительным, то прибавить к нему 10; в противном случае вычесть из него 8. Вывести полученное число.

```
Program if_2;
Var a:integer;
Begin
  Writeln('Введите число');
  Readln(a);
  If a>0 then a:=a+10 else a:=a-8;
  Writeln('Полученное число ',a);
  Readln;
End.
```

3. Дано 3 целых числа. Найти количество положительных в исходном наборе.

```
Program if_4;
Var a,b,c,k1:integer;
Begin
  Writeln('Введите число');
  Readln(a);
  Writeln('Введите число');
  Readln(b);
  Writeln('Введите число');
  Readln(c);
  k1:=0;
  If a>0 then k1:=k1+1;
  If b>0 then k1:=k1+1;
  If c>0 then k1:=k1+1;
  Writeln('Количество положительных чисел ',k1);
  Readln;
End.
```

4. Дано 2 целых числа. Вывести большее из них.

```
Program if_6;
Var a,b:integer;
Begin
  Writeln('Введите число');
  Readln(a);
  Writeln('Введите число');
  Readln(b);
  If a>b then Writeln('Наибольшее ',a) else
  Writeln('Наибольшее ',b)
  Readln;
```

End.

5. Дано 2 целых числа. Вывести порядковый номер меньшего из них.

```
Program if_7;
Var a,b:integer;
Begin
  Writeln('Введите число');
  Readln(a);
  Writeln('Введите число');
  Readln(b);
  If a<b then Writeln('Наименьшее 1-е') else
  Writeln('Наименьшее 2-е ')
  Readln;
End.
```

6. Дано число. Определить, оканчивается ли данное число на цифру 7.

```
Program If_10;
var a,b:integer;
Begin
  writeln ('Введите число');
  readln (b);
  a:=b mod 10;
  if (a=7) then writeln ('Число оканчивается на 7')
  else writeln ('Число не оканчивается 7');
  Readln;
end.
```

7. Дано число. Определить, является ли данное число четным или нечетным.

```
Program If_11;
var a:integer;
Begin
  writeln ('Введите число');
  readln (b);
  if a mod 2 =0 then writeln ('Число - четное') else
  writeln ('Число нечетное');
  Readln;
end.
```

8. Ввести возраст от 1 до 99. В зависимости от введенного возраста вывести фразу «Мне...лет(год, года)»

```
Program case_4;
Var k:integer;
Begin
  Writeln('Введите возраст');
  Readln(k);
  Case k mod 10 of
    1:Writeln('Мне ',k,' год');
    2..4:Writeln('Мне ',k,' года');
    0,5..9:Writeln ('Мне ',k,' лет');
    ElseWriteln('такого возраста не существует');
  end;Readln;End.
```

9. Ввести номер месяца. Определить по введенному номеру – квартал.

```
Program Case_5;
```

```

var  n:integer;
begin
write('Введите номер месяца:');
readln(n);
case n of
1..3 :writeln('Первыйквартал');
4..6 :writeln('Второй квартал');
7..9 :writeln('Третий квартал');
10..12: writeln('Четвертый квартал')
elsewriteln('Некорректный ввод')
end;
Readln ;end.

```

10. Составить программу, проверяющую верно ли утверждение, что сумма цифр введенного двухзначного целого числа делится без остатка на 5.

Фрагментрешения

```

Program if_z30;
uses crt;
var x,a,b,c:integer;
begin
ClrScr;
Writeln('введите двузначное число');
readln(x);
a:=x div 10;
b:= xmod10;
c:= a+b;
if c mod 5 =0
thenwriteln ('Верно. Делится') else writeln
('неверно');
Readln; end.

```


Практическая работа №27 Решение задач на одномерные массивы

Пример. Упорядочить массив шести целых чисел в порядке убывания.

```
program zadacha_5
uses crt;
var
m: array[1..6] of integer;
i,j,k,n:integer;
begin
clrscr;
for i:=1 to 6 do
begin
write ('m[' ,i, ']=');
read (m[i]);
end;
for i:=6 downto 2 do
for j:=1 to i-1 do
begin
if m[j] < m[j+1] then
begin
n:=m[j+1];
m[j+1]:=m[j];
m[j]:=n;
end;
end;
writeln ('новый массив');
for i:=1 to 6 do
begin
write ('m[' ,i, ']=',m[i], ' '); readln;
end;
readln;
end.
```

Задание 1 Упорядочить массив 10 целых чисел в порядке возрастания

Пример 2. Ввести массив из n элементов, найти среднее арифметическое элементов массива.

```
Program mas_2;
Var a:array[1..100] of integer;
n,l:integer;
P:longint;
s:real;
Begin
Writeln('введите кол-во эл-тов массива '); Readln(n);
For i:=1 to n do
Begin
Writeln('укажите эл-т массива '); Readln(a[i]);
End;
s:=0;
For i:=1 to n do s:=s+a[i];
Writeln('среднее арифметическое =',s/n);
Readln;
End.
```

Задача 2 Ввести массив из n элементов, найти среднее арифметическое элементов массива и прибавить к нему целое число $r=10$.

Задача 3 Ввести массив из n элементов, найти произведение элементов массива и прибавить к нему некое число C .

Практическая работа №28 Решение задач на нахождение минимального и максимального элемента.

1 Пример Определить самую высокую температуру и самый теплый день в мае.

```
Program massiv;  
Uses crt;  
Var t:array [1..5] of integer; - массив темп из 5 эл-ов  
    I, max, n : integer; (I –счетчик цикла, макс- максзнач , n-№ дня)  
Begin  
  Clrscr;  
  For i:=1 to 5 do (счетчик цикла)  
    Begin  
      Write('Введите темп в ',I,'-й день');  
      Readln (t[i]);  
    End;  
  Max:=t[1]; n:=1; (начзн макс темп и № дня)  
  For i:=2 to 5 do (счетчик увел на 1 т к надо начинать проверку со 2 дня)  
    Begin  
      If t[i] > max then  
        Begin  
          max:=t[i]; n:=i ; (если найдено наиб знач, то меняется максимум и запомин в какой день )  
        End;  
      End;  
  Write('макстемп ',max , ' в ',n,'день');  
End.
```

1. Составить программу, которая проверяет есть ли файл с заданным именем на диске.

```
program zadacha_3;
uses crt;
var
f: text;
st: string;
IOresult: integer;
begin
clrscr;
writeln('введите имя файла: ');
readln(st);
assign(f,st);
{$I-}
reset(f);
{$I+}
if IOresult=0 then
begin
write('файл существует и нормально открыт');
close(f);
end
else
write('файл с именем ',st, ' не существует');
readln;
end.
```

!!!- После создания файла. Сохраните на раб.столе программу под именем **program 3**, затем снова откройте программу с раб стола и запустите на выполнение. Обоснуйте результат!

2. Посчитать количество строк в текстовом файле.

```
program zadacha_4;
uses crt;
var
f: text;
st,st1: string;
k: integer;
begin
clrscr;
writeln('Введите имя файла: ');
readln(st);
assign(f,st);
{$I-}
reset(f);
{$I+}
while not eof(f) do
begin
readln(f,st1);
k:=k+1;
end;
close(f);
write('k=',k);
readln;
end.
```

!!!- После создания файла. Сохраните на раб.столе программу под именем **program 4**, затем снова откройте программу с раб стола и запустите на выполнение. Обоснуйте результат!

3. Найдти длину строк текстового файла.

```
program zadacha_5;
uses crt;
var
  f: text;
  st, st1: string;
  s: char;
  k, m, i: integer;
begin
  clrscr;
  writeln('Введите имя файла: ');
  readln(st);
  assign(f, st);
  {$i-}
  reset(f);
  {$i+}
  while not eof(f) do
  begin
    readln(f, st1);
    m := length(st1);
    write('m=', m);
    readln;
  end;
  close(f);
end.
```

!!!- После создания файла. Сохраните на раб.столе программу под именем **program 5**, затем снова откройте программу с раб стола и запустите на выполнение. Обоснуйте результат!

```
.
usesGraphABC,PointRect;
var
r: Rect;
i: integer;
begin
SetWindowSize(700,520);
SetBrushStyle(bsClear);
r:=RectF(100,10,600,510);
for i:=1 to 60 do
begin
InflateRect(r,-4);
Rectangle(r.Left,r.Top,r.Right,r.Bottom);
Sleep(1);
end;
InflateRect(r,-2);
for i:=1 to 59 do
begin
InflateRect(r,4);
Rectangle(r.Left,r.Top,r.Right,r.Bottom);
Sleep(1);
end;
end.
```

2. // Анимация без мерцания.

ИспользованиеLockDrawing

```
usesGraphABC;
var i: integer;
begin
LockDrawing;
for i:=1 to 500 do
begin
ClearWindow;
SetBrushColor(clGreen);
Ellipse(i,100,i+100,200);
Redraw;
Sleep(1);
end;
end.
```

// Демонстрация элементарной анимации.

Перемещениекартинки

```
usesGraphABC;
const speed=1;
var
w,h,i,j,pic: integer;
begin
SetWindowCaption('Перемещениекартинки');
pic:=LoadPicture('demo.bmp');
w:=PictureWidth(pic);
```

Практическая работа №31 Разработка простейшего приложения в среде Delphi

Задание 1. Разработайте простейшее приложение

-Для разработки простейшего приложения надо его открыть. При входе в систему оно формируется автоматически. На форме разместить из страницы Standardкомпоненты Label(метку - пиктограммой ^A) и Button(кнопку - пиктограммой ,СШ). Выбрать поочередно компоненты и установить в Инспекторе объектов свойства компонентов:

- для компонента Form1: Caption= Простейшее приложение
- в Fontкомпонента Label1 размер шрифта = 18;
- для компонента Label1: Caption= Для вывода сообщения
- для компонента Button1: Caption= Нажми.

Уменьшить с помощью мыши размер формы. Вид формы после установки свойств компонентов пан на рисунке:



Сформировать метод для реакции приложения на нажатие кнопки Нажми. Для этого дважды щелкнуть на кнопке. Появится заготовка метода. Написать в нем текст метода, приведенный в листинге программы.

Запустить приложение на выполнение командой Run\Runили быстрой кнопкой F9. Вид формы после запуска приложения и нажатия кнопки 'Нажми' дан на рис. 4.8. Завершить приложение нажатием кнопки с крестиком в правом верхнем углу формы.

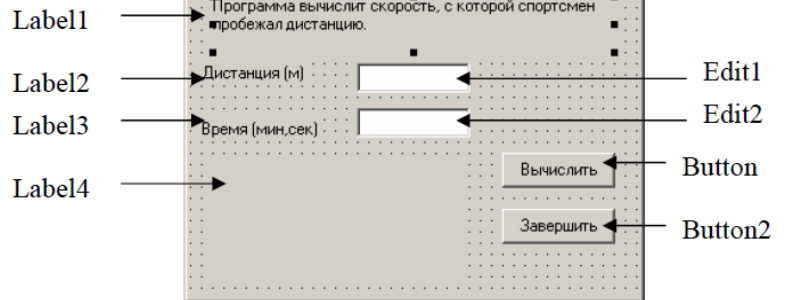
Листинг Текст модуля приложения Unit1.pas

```
unit Unit1; interface
uses Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls; type
TForm1 = class(TForm)
    Label1: TLabel;                                     // - компонент- метка
    Button1: TButton;                                   // - компонент- кнопка
    procedure Button1Click(Sender: TObject);           // - для метода кнопки
private { Private declarations } public { Public declarations }
end;
var Form1: TForm1; implementation{$R *.dfm}
// Метод реакции на нажатие кнопки:
procedure TForm1.Button1Click(Sender: TObject);
begin
    Label1.Color := clAqua;                             // - для изменения цвета фона метки
    Label1.Caption := 'Здравствуй, Delphi!';             // - для вывода текста на метку
end;
end. // - приложения.
```

2 Задание . Создайте свой собственный пример, улучшив программу (цвет, шрифт, добавление новых кнопок)

Практическая работа №32 Разработка приложения в среде BorlandDelphi с компонентами ввода и отображения текстовых данных

Создать новый проект: File - New -Application.
Сразу сохранить его в папке «Проект2» на рабочем столе.
1 На стартовой форме создать следующее окно приложения



2. Установите следующие свойства стартовой формы

	Свойство	Значение		Свойство	Значение
1	Caption	Скорость бега	4	BorderStyle	bsSingle
2	Height	250	5	Font Size	12
3	Width	330			

Поместите на форму указанные на рисунке компоненты (щелчок на компоненте в палитре компонентов, щелчок в нужном месте формы).

4. Установите следующие свойства для **ЭТИХ КОМПОНЕНТОВ**:

Свойство	Компонент			
	Label1	Label2	Label3	Label4
AutoSize	False	True	True	False
WordWrap	True	False	False	True
Caption	Программа вычислит скорость, с которой спортсмен пробежал дистанцию.	Дистанция (м)	Время (мин, сек)	
Top	8	56	88	120
Left	8	8	8	8
Height	35			41
Свойство	Компонент			
	Edit1	Edit2	Button1	Button2
Text	Очистить содержимое этого поля	Очистить содержимое этого поля		
Caption			Вычислить	Завершить
Top	56	88	176	176
Left	128	128	16	112
Height	21	21	25	25
Width	121	121	75	75

5. После оформления внешнего вида формы приступим к программированию кнопок.

- Выделить кнопку **Завершить**, в окне инспектора объектов перейти на вкладку **Events** и дважды щелкнуть в поле события **OnClick**.

В открывшемся окне редактора кода набрать только команду **Form1.Close**;

```
procedure TForm1.Button2Click(Sender: TObject);
```

```
begin
```

```
Form1.Close;
```

```
end;
```

6. Запустить программу на выполнение.

В работающей программе нажать на кнопку **Завершить**.

7. Аналогично записать процедуру для кнопки **Вычислить**

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
dist : integer;
```

```
t: real;
```

```
min : integer;
```

```
sek : integer;
```

```
v: real
```

```
begin
```

```
dist := StrToInt(Edit1.Text);
```

```
t := StrToFloat(Edit2.Text);
```

```
min := Trunc(t);
```

```
sek := Trunc(t*100) mod 100;
```

```
v := (dist/1000) / ((min*60 + sek)/3600);
```

```
label4.Caption := 'Дистанция: ' + Edit1.Text + ' м' + #13 + //
Обратный перевод чисел 'Время: ' + IntToStr(min) + ' мин ' +
// текстовое представление
IntToStr(sek) + ' сек ' + #13 + 'Скорость: ' +
FloatToStrF(v, ffFixed, 4, 2) + ' км/час';
end;
```

8. Еще раз запустить программу на выполнение. Если программа работает нормально, то можно поэкспериментировать с различными цветами формы и ее компонентов – свойство **Color**, со шрифтами – свойство **Font**, изменить размеры формы и ее компонентов.

9. Показать работу учителю.

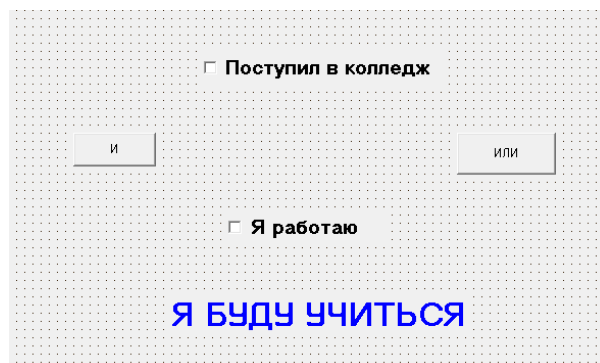
2

Задание 2. Аналогично 1 заданию, создайте программу, которая вычисляет стоимость покупки.

Практическая работа №33 Разработка приложения в среде BorlandDelphi с группирующими компонентами

Задание 1. Создать проект «Буду учиться» при условиях: «И» и «ИЛИ» - «Поступил в колледж», «Я работаю».

Кнопки должны быть расположены по образцу:



- 1) На вкладке Standart выберите компонент CheckBox (контейнер для отметок) и разместите 2 таких компонента в форме : 1-вверху, 2- внизу.
- 2) Выделите 1-ый контейнер и перейдите на вкладку СЛЕВА, выберите свойство CAPTION – напишите : Поступил в колледж
- 3) Выделите 2-ой контейнер и перейдите на вкладку СПРАВА, выберите свойство CAPTION – напишите : Я работаю
- 4) На вкладке Standart выберите компонент Button (кнопка) и разместите 2 таких компонента в форме : 1-слева, 2-ой - справа.
- 5) Таким же образом через свойство CAPTION – напишите название для 1-ой кнопки – **И**, для второй – **ИЛИ**.
- 6) Разместите внизу формы компонент **Label1** и через свойство CAPTION – напишите **БУДУ УЧИТЬСЯ**
- 7) * Для красочного оформления надписи используйте свойство Font, Color.
- 8) Для компонента **Label1** найдите в Свойствах поле Visible (видимая). И установите значение false.
- 9) Затем 2 раза щелкните на кнопке И, в открывшемся окне Until 1 в заготовке процедуры TForm1.Button1.Click, наберите команды условий:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
if CheckBox1.Checked and CheckBox2.Checked
then Label1.Visible:=true
else Label1.Visible:=false;
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);
begin
if CheckBox1.Checked or CheckBox2.Checked
then Label1.Visible:=true
else Label1.Visible:=false;
end;
```

end.

- 10) Перейдите в окно форм и Запустите проект - нажмите F9.
- 11) Сохраните файлы в своей папке под именем Project1 – для этого выберите п.м. Файл – Сохранить все – выберите свою папку и назовите файл – нажмите Сохранить- еще одно название – нажмите Сохранить.
- 12) Создайте самостоятельно кнопку «Выход»

2 Задание. Создайте аналогичную программу (свой пример), произведите модификацию (улучшение) программы.

Цель работы - создать программу, выполняющую следующие действия:

1. После запуска программы в окне изображается три поля.
2. По щелчку мышью на кнопке "Случайный выбор" из трех слов составляется предложение случайным образом.

3. Для выхода из программы необходимо щелкнуть мышью на закрывающей кнопке в строке заголовка.

Описание плана разработки программы

1. Открыть новый проект.

2. Разместить на форме экземпляры компонентов: поле со списком ComboBox, командная кнопка Button.

3. Выполнить следующие действия:

Список для **ComboBox1**

Кошка Плавает
Змея Бегаёт
Кузнечик Летает
Дельфин Ползает
Черепашка Прыгает
Ласточка Прячется

Список для **ComboBox2**

Быстро
Высоко
Медленно
Сильно
Хорошо
Плохо

Список для **ComboBox3**

Плавает
Бегаёт
Летает
Ползает
Прыгает
Прячется

Листинг подпрограммы

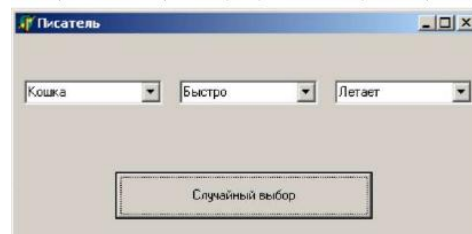
```
procedure TForm1.FormCreate
(Sender: TObject); begin
```

```
Randomize;
```

```
ComboBox1.ItemIndex :=0;
ComboBox2.ItemIndex :=0;
ComboBox3.ItemIndex :=0; end;
```

```
procedure TForm1.Button1Click
(Sender: TObject); begin
```

```
ComboBox1.ItemIndex :=
Random(ComboBox1.ItemIndex);
ComboBox2.ItemIndex := Random(ComboBox2.ItemIndex);
ComboBox3.ItemIndex := Random(ComboBox3.ItemIndex); end;
```



Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Form1	Properties	Caption	Установка имени формы "Сочинитель"
	Events	OnCreate	ComboBox1.ItemIndex :=0; ComboBox2.ItemIndex :=0; ComboBox3.ItemIndex :=0;
	Properties	Style	Выберите значение cSDropDownList из раскрывающегося списка
ComboBox1 (Вкладка Standard)		Items	Щелкните на кнопке построителя. Откроется окно String List Editor (Редактор списка строк). Ввести пункты списка по одному в каждую строчку, завершая ввод нажатием клавиши Enter. После того как список готов, щелкнуть на кнопке OK.
ComboBox2 (Вкладка Standard)	Properties	Style	Выберите значение cSDropDownList из раскрывающегося списка
		Items	Щелкните на кнопке построителя. Откроется окно String List Editor (Редактор списка строк). Ввести пункты списка по одному в каждую строчку, завершая ввод нажатием клавиши Enter. После того как список готов, щелкнуть на кнопке OK.
ComboBox3 (Вкладка Standard)	Properties	Style	Выберите значение cSDropDownList из раскрывающегося списка
		Items	Щелкните на кнопке построителя. Откроется окно String List Editor (Редактор списка строк). Ввести пункты списка по одному в каждую строчку, завершая ввод нажатием клавиши Enter. После того как список готов, щелкнуть на кнопке OK.
Button1 (Вкладка Standard)	Properties	Caption	Установка имени кнопки "Случайный выбор"
	Events	OnClick	ComboBox1.ItemIndex := Random(ComboBox1.ItemIndex.Count); ComboBox2.ItemIndex := Random(ComboBox2.ItemIndex.Count); ComboBox3.ItemIndex := Random(ComboBox3.ItemIndex.Count);

Практическая работа №34 Разработка приложения в среде BorlandDelphic добавлением компонента Ttimer

Цель работы - создать программу-игру, выполняющую следующие действия:

1. После запуска программы в окне изображается беспорядочно прыгающая кнопка.

2. Необходимо успеть щелкнуть по ней.

3. Кнопка перепрыгивает из одного места в другое по сигналу, полученному от таймера.

4. Для выхода из программы необходимо щелкнуть мышью на закрывающей кнопке в строке заголовка.

Описание плана разработки программы

1. Открыть новый проект.

2. Разместить на форме экземпляры компонентов: командная кнопка Button, таймер Timer.

3. Выполнить следующие действия:

Задание для самостоятельного выполнения

1. Измените игру так, чтобы скорость можно было настраивать в процессе игры.

2. Создайте две кнопки: Медленнее и Быстрее. Щелчок на одной из них будет увеличивать или уменьшать значение свойства Timer1.Interval на 100 миллисекунд.

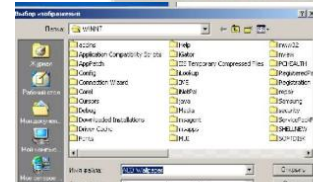
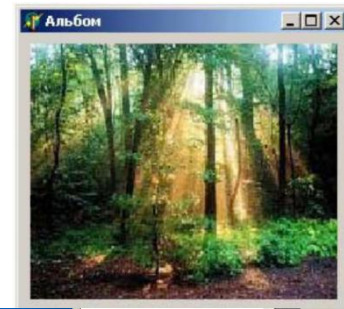
Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Form1	Properties	Caption	Установка имени формы "Прыгающая кнопка"
		ClientWidth (Внутренняя ширина)	Присвоить значение 300
		ClientHeight (Внутренняя высота)	Присвоить значение 200
		BorderStyle (тип границы)	Выбрать значение bsSingle (тонкая)
	Events	OnCreate	Randomize;
Button1 (Вкладка Standard)	Properties	Caption	Ввести надпись "Нажми меня"
		TabStop	Присвоить значение False. Это свойство позволяет выбрать данный элемент управления клавишей Tab. Клавиатурой пользоваться запрещается.
		Visible	Присвоить значение False. Сначала кнопка невидима.
		Height	Присвоить значение 30
		Width	Присвоить значение 80
	Events	OnClick	Button1.Caption := 'Готово'; Button1.Enabled := False; Timer1.Enabled := False;

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Button2 (Вкладка Standard)	Properties	Caption	Ввести надпись "Выход"
		Default (по умолчанию)	Выбрать значение True
		Left (слева)	Присвоить значение 110
		Top (сверху)	Присвоить значение 160
		Width (ширина)	Присвоить значение 80
		Height (высота)	Присвоить значение 30
	Events	OnClick	Close;
Timer1 (Вкладка System)	Properties	Interval (интервал)	Присвоить значение 500 (промежуток времени измеряется в миллисекундах)
	Events	Timer	var i: Integer; begin i:=Random(9); Button1.Visible := True; Button1.Top := 10 + 50 * (i div 3); Button1.Left := 10 + 100 * (i mod 3); end;

Практическая работа №35 Разработка приложения в среде BorlandDelphic компонентом Image

Цель работы - создать программу, выполняющую следующие действия:

1. После запуска программы в окне изображается рисунок.
2. По щелчку мышью на рисунке появляется диалоговое окно.
3. Выбрать в открывшемся диалоговом окне любой другой рисунок.
4. Для выхода из программы необходимо щелкнуть мышью на закрывающей кнопке в строке заголовка.



Описание плана разработки программы

1. Открыть новый проект.
2. Разместить на форме экземпляры компонентов: панель Panel, рисунок Image, диалоговое окно OpenFileDialog.
3. Выполнить следующие действия:
4. Сохраните проект, запустите и протестируйте его.

Листинг подпрограммы

procedure TForm1.Image1Click (Sender:
TObject); begin

OpenDialog1.Execute;

Image1.Picture.LoadFromFile
(OpenDialog1.FileName); end;

***Растровое изображение

(* .JPEG) | *.JPEG)

. JPEG

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Form1	Properties	Caption	Установка имени формы "Альбом"
Panell (Вкладка Standard)	Properties	Caption	Очистите значение свойства Caption
		BevelOuter	Выбрать в раскрывающемся списке значение bvLowered
		BevelInner	Выбрать в раскрывающемся списке значение bvNone
		BevelWidth	Присвоить значение 2
		Width	Присвоить значение 241
		Height	Присвоить значение 185
Image1 (Вкладка Additional)	Properties	Left	Присвоить значение 2
		Top	Присвоить значение 2
		Width	Присвоить значение 237
		Height	Присвоить значение 181
		Stretch	Включить свойство True
		Picture	С помощью кнопки-построителя открыть диалоговое окно Picture Editor (Редактор изображений). Щелкнуть на кнопке Load (Загрузить) - откроется диалоговое окно Load Picture (Загрузка рисунка). Открыть папку C:\Windows и выбрать файл Лес.bmp, щелкнуть на кнопке Открыть. Вернуться в окно Редактора изображений, щелкнуть на кнопке OK. Image1.Picture.LoadFromFile (OpenDialog1.FileName);
	Events	OnClick	OpenDialog1.Execute;
OpenDialog1 (Вкладка Dialogs)	Properties	Title	Ввести текст: "Выбор изображения"
		FileName	Ввести полный путь доступа к файлу: C:\Windows\YPec.bmp
		Filter	Ввести текст: Растровое изображение (*.bmp) *.bmp
		DefaultExt	Присвоить свойству значение: .bmp
		Options	Подсвойству ofFileMustExit (Файл должен существовать) присвоить значение True (Да)

Практическая работа №36 Разработка приложения в среде BorlandDelphic компонентом PaintBox для вывода графиков

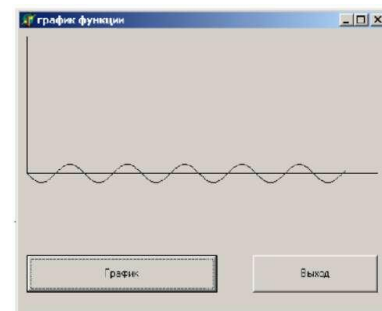
Тема: Построение графиков

Задание 1.

Цель работы - создать программу построения графика, в которой устанавливается масштаб, в цикле осуществляется построение графика функции, рисуются оси координат и печатаются на них числовые шкалы.

Описание плана разработки программы

1. Открыть новый проект.
2. Разместить на форме экземпляры компонентов: кнопка Button, область рисования PaintBox.
3. Выполнить следующие действия:



Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Form1	Properties	Caption	Установка имени формы "График функции"
Button1	Properties	Caption	Введите название "График"
	Events	OnClick	Написать процедуру, рисующую график по точкам
PaintBox1	Properties	Canvas	Соединение линиями получаемых точек
Button2	Properties	Caption	Введите название "Выход"
	Events	OnClick	Close;

Листинг программы

```
procedure TForm1.Button1Click(Sender: TObject);
function f(x:integer):integer; var c:integer; begin
c:=round(10*sin(0.1*x)); f:=c; end;
var x:integer; begin
PaintBox1.Canvas.LineTo(0,150);
PaintBox1.Canvas.LineTo(410,150);
x:=0;
for x:=0 to 350 do begin
PaintBox1.Canvas.LineTo(x, f(x)+150);
end; end; end.
```

Кнопку выхода пропишите самостоятельно!

4. Сохраните проект, запустите и протестируйте его

Практическая работа №37 Разработка приложения в среде BorlandDelphic компонентом Chart для вывода графиков

Задание 2. Создание графиков с помощью компонента Chart

Поместите компонент Chart с вкладки Additional на форму и запустите редактор диаграмм (щелчок пр. КН.мыши по компоненту Chart на форме – Edit Chart) .

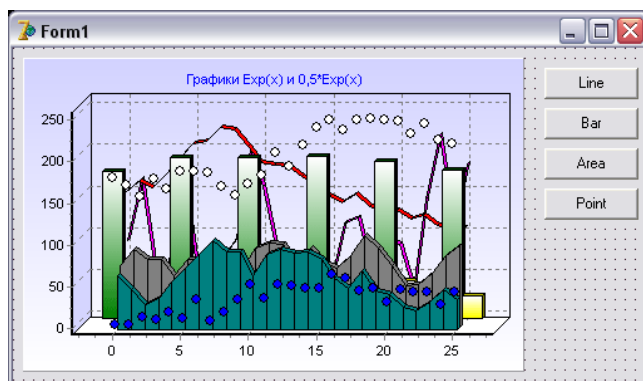
Перейдите на вкладку Chart – Add

Создайте восемь кривых следующих типов:

- Line – 2 серии;
- Bar – 2 серии;
- Area – 2 серии.
- Point – 2 серии;

В приложении будут отображаться две функции, но каждая из них может быть представлена различными способами: лентами, столбиками, плоскостями, точками, поэтому были созданы восемь серий (по четыре на каждую кривую).

- Перейдите к вкладке Legend и снимите флажок с пункта Visible. Перейдите к вкладке Titles и в текстовом поле наберите следующее: «Графики $\text{Exp}(x)$ и $0,5*\text{Exp}(x)$ ».
- На вкладке Panels установите флажок около пункта Visible для того, чтобы разрешить градиентную заливку заднему фону графика. Измените цвета градиента.
- На вкладке Walls измените цвет левой стенки графика. На вкладке 3D установите величину 3D равной 5%.



В редакторе диаграмм перейдите к вкладке Series, в раскрывающемся списке выберите пункт «Series3» (тип Bar), затем откройте вкладку Marks и снимите флажок с пункта Visible. Прodelайте то же самое с Series4. Если серии типа Bar у вас имеют другие имена, то нужно менять их свойства. Данное действие запрещает показывать на графике значения функций в каждой точке.

-Разместите на форме 4 кн. Button как на рис выше.

Пропишите для каждой кн соответственно код:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  N:=0;Grafik(N,M); M:=0;
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);
begin
  N:=2;Grafik(N,M); M:=2;
end;
```

```
procedure TForm1.Button3Click(Sender: TObject);
begin
  N:=4;Grafik(N,M); M:=4;
end;
```

```
procedure TForm1.Button4Click(Sender: TObject);
begin
  N:=6;Grafik(N,M);M:=6;
end;
```

Введите две общие переменные N и M типа integer. С помощью них будем обращаться к номеру серии компонента Chart1,

```
var
  Form1: TForm1;
  N,M:integer;
```

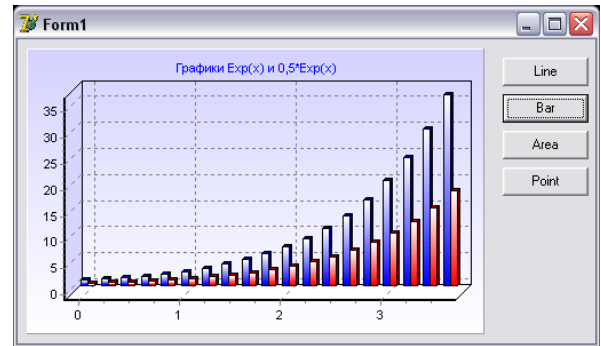
Для отображения линии напшм процедуру (в данном случае она названа Grafik), которая будет очищать содержимое серий и отображать линии на них.

implementation

```
{ $R *.dfm }
```

Напишете процедуру:

```
procedure Grafik(N,M:integer);
var
  x: real; i:integer;
begin
  begin
    Form1.Chart1.Series[M].Clear;
    Form1.Chart1.Series[M+1].Clear;
    x:=0;
    for i:=0 to 18 do begin
      Form1.Chart1.Series[N].AddXY(x, exp(x), ", clBlue);
      Form1.Chart1.Series[N+1].AddXY(x, exp(x)/2, ", clRed);
      x:=x+0.2;
    end;
  end;
end;
```



Практическая работа №38 Разработка приложения в среде BorlandDelphic компонентом Chart для вывода графиков на отдельной форме

Тема: Создание графика с помощью компонента Chart. Рисование на холсте Canvas.

Задание 1. Создать приложение с использованием компонента Chart. В данном приложении должен строиться график функции параболы. Рекомендуемый вид формы:

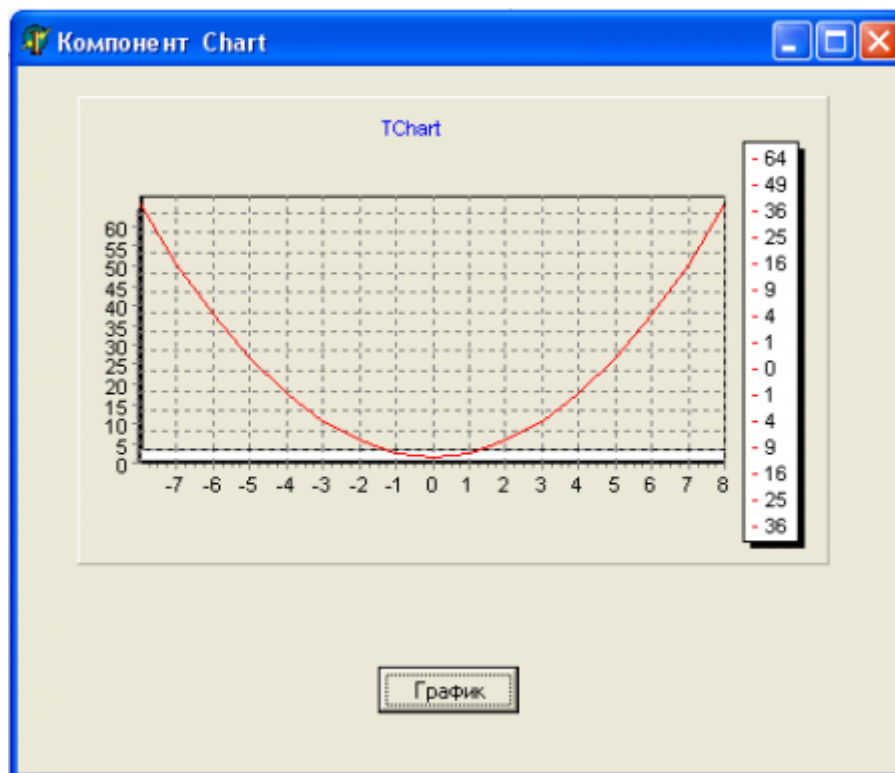


Рис 1. График функции параболы

Ход работы:

Разместите на форме компонент Chart, дважды по нему щелкните мышью. На вкладке Series нажмите кнопку Add и выберите график, который Вы хотите получить. Далее на форме располагаем кнопку и в ее обработчике пишем следующий код:

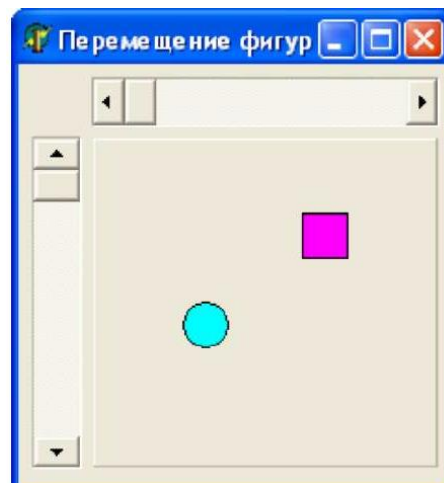
```
procedure TForm1.Button1Click(Sender: TObject);  
var  
  X,Y : Integer;  
begin  
  for X:=-8 to 8 do begin  
    Y:=X * X;  
    Series1.AddXY(X,Y);  
  end;  
end;
```

В результате мы получим график функции параболы.

Практическая работа №39 Разработка приложения в среде BorlandDelphic компонентом Shape

Цель работы - создать программу, выполняющую следующие действия:

1. После запуска программы в окне изображается две полосы прокрутки. Вертикальная полоса будет управлять движением по вертикали, горизонтальная - по горизонтали.
2. Наводя указатель мыши на одну из двух фигур, можно выбирать, какая из этих фигур связана с полосами прокрутки.
3. Требуется дополнительные объекты, с помощью которых ограничивается область движения фигур в окне.
4. Если полоса прокрутки активная, то она должна реагировать на клавиши ВВЕРХ, ВНИЗ, ВЛЕВО, ВПРАВО, PAGE UP, PAGE DOWN.
5. Для выхода из программы необходимо щелкнуть мышью на закрывающей кнопке в строке заголовка.



Описание плана разработки программы

1. Открыть новый проект.
2. Разместить на форме экземпляры компонентов: панель Panel, полоса прокрутки ScrollBar, фигура Shape.
3. Ввести дополнительную переменную логического типа num. Если она принимает значение True (Да), то текущей считается первая фигура. Значению False (Нет) соответствует вторая фигура. Эта переменная должна быть доступна во всех процедурах.
4. Выполнить следующие действия:

1)

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Form1	Properties	Caption	Установка имени формы "Перемещение фигур"
Panel (Вкладка Standard)	Properties	Height	Присвоить значение 161
		Width	Присвоить значение 161
		Caption	Оставить значение свойства пустым
ScrollBar1 (Вкладка Standard)	Properties	Min	Присвоить значение 5
		Max	Присвоить значение 145
		Position	Присвоить значение 76
		SmallChange	Присвоить значение 2

Продолжение таблицы 13.

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
	Events	LargeChange (Большое изменение)	Присвоить значение 20
		OnChange	if num then Shape1.Left := ScrollBar1.Position else Shape2.Left := ScrollBar1.Position
ScrollBar2 (Вкладка Standard)	Properties	Kind	Выбрать значение sbVertical. Горизонтальная полоса прокрутки станет вертикальной.
		Min	Присвоить значение 5
		Max	Присвоить значение 145
		Position	Присвоить значение 76
		SmallChange (Малое изменение)	Присвоить значение 2
		LargeChange	Присвоить значение 20
	Events	OnChange	if num then Shape1.Top := ScrollBar2.Position else Shape2.Top := ScrollBar2.Position

2) (продолжение таблицы)

Shapel (Вкладка Additional)	Properties	Height	Присвоить значение 11
		Width	Присвоить значение 11
		Left	Присвоить значение 76
		Top	Присвоить значение 76
		Shape (Форма)	Выбрать значение stCircle (Круг)
		Brush (Кисть)	Выбрать для подсвойства Color (Цвет кисти) значение cAqua (голубой цвет)
	Events	OnMouseMove (При движении мыши)	Shape1.Brush.Color := clAqua; Shape1.Brush.Color := clFuchsia; Num := True; ScrollBar1.Position := Shape1.Left; ScrollBar2.Position := Shape1.Top;
Shape2 (Вкладка Additional)	Properties Events	Height	Присвоить значение 11
		Width	Присвоить значение 11
		Left	Присвоить значение 76
		Top	Присвоить значение 76
		Shape	Выбрать значение stSquare (Квадрат)
		Brush	Выбрать для подсвойства Color (Цвет кисти) значение clFuchsia (фиолетовый цвет) Аналогично Shape2
	Events	OnMouseMove	

5. Сохраните проект, запустите и протестируйте его. Листинг подпрограммы

```

procedure TForm1.ScrollBar1Change (Sender: TObject); begin
if num then Shape1.Left := ScrollBar1.Position else Shape2.Left := ScrollBar1.Position
end;
procedure TForm1.ScrollBar2Change (Sender: TObject); begin
if num then Shape1.Top := ScrollBar2.Position else Shape2.Top := ScrollBar2.Position
end;
procedure TForm1.ShapelMouseMove
(Sender: TObject; Shift: TShiftState; X, Y: Integer);

```



```
begin  
Shapel.Brush.Color := clAgua; Shapel.Brush.Color := clFuchsia; Num := True;  
ScrollBar1.Position:= Shapel.Left; ScrollBar2.Position:= Shapel.Top; end;
```

procedure TForm1. Shape2MouseMove

```
(Sender: TObject; Shift: TShiftState; X, Y: Integer);  
begin  
Shape2.Brush.Color := clFuchsia; Shape2.Brush.Color := clAgua; Num := False;  
ScrollBar1.Position:= Shape2.Left; ScrollBar2.Position:= Shape2.Top; end;
```

```
procedure TForm1. FormCreate (Sender: TObject); begin  
num := True; end;
```

Практическая работа №40 Разработка приложения с группой компонентов типа TButton

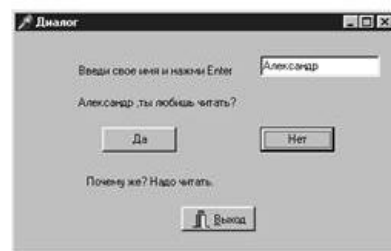
"Разработка проекта ДИАЛОГ"

1. Постановка задачи

Создать программу, с помощью которой можно выполнить: после запуска программы пользователь вводит свое имя, например, Александр, в прямоугольник с мигающим текстовым курсором и нажимает клавишу Enter (см. рисунок 1).

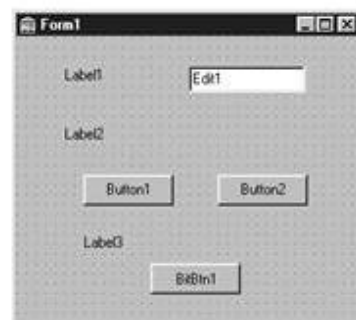
Появляется вопрос: "Александр, ты любишь читать?". Если пользователь нажмет на клавишу Да, то должна появиться реплика: "Молодец!", если он нажмет на клавишу Нет, то появится реплика: "Почему же? Надо читать".

Для выхода из программы необходимо щелкнуть мышью на кнопке Выход.



2. План разработки программы

1. Открыть новый проект. Дать название форме Диалог.
2. Разместить на форме компоненты в соответствии с рисунком 2
3. Сохранить проект под именем, например, Pr2.pas и Pr2.dpr
4. Запустить программу, затем закрыть окно проекта кнопкой Выход.
5. Выполнить следующие действия:



Выделенная компонента	Окно инспектора объектов	Имя свойства	Действие
BitBtn1	Properties	Caption	&Выход
		Kind	bkClose
Label1	Properties	Caption	Введи свое имя и нажми Enter
Edit1	Events	OnKeyPress	If key=#13 then begin Label2.Caption:=Edit1.Text + ' , ты любишь читать?'; end;
Button1	Properties	Caption	Да
	Events	OnClick	Label3.Caption:='Молодец!';
Button2	Properties	Caption	Заменить на Нет
	Events	OnClick	Label3.Caption:='Почему же? Надо читать.';
Label2	Properties	Caption	Удалить название
Label3	Properties	Caption	Удалить название

6. Сохранить проект, запустить и протестировать его.

Задание

- Выполнить сброс значений компонентов Edit1, Label2, Label3 для обеспечения возможности повторить диалог.

Подсказка. Разместить на форме еще одну кнопку **BitBtn**. Свойству Kind придать значение bkRetry, а свойству Caption - **&Повторить**. Ввести программно пустые строки в свойства указанных компонентов по нажатию кнопки

Повторить:

Label2.Caption := "";

Label3.Caption := "";

Edit1.Text := "";

Практическая работа №41 Разработка приложения с компонентами типа TEdit

1. Рассчитать зарплату рабочему, если известно, что он отработал а-дней, а f-плата за 1 день

«подсказка» - объявите переменные после слова **implementation**

```
{ $R *.dfm }
```

```
VAR f,t,a : Integer;
```

Создайте клик на кн. «Расчитать»»

```
procedure
```

```
TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
f:= StrToInt(Edit1.Text);
```

```
t:= StrToInt(Edit2.Text);
```

```
a:=f*t;
```

```
label3.Caption :=IntToStr(a); ;
```

```
end;
```

```
end.
```

Протестируйте приложение.

1.1Добавьте кн. Выход. – напишите для нее процедуру.

1.2Добавьте строку для ввода Фамилии.

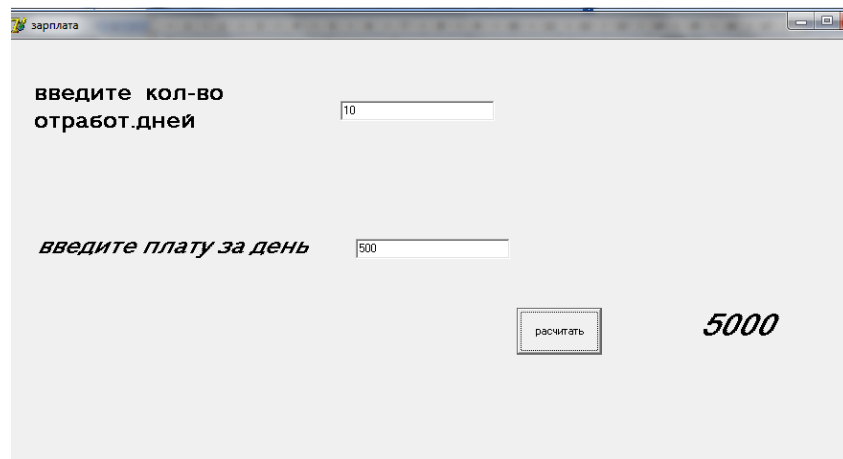
Сохраните проект.

2.Создайте самостоятельно программу расчета платы за детский сад.

Приложение должно содержать информацию об имени и фамилии ребенка, о вводе названия месяца, кол-ве дней, которые посетил ребенок (с), плату за 1 день (к), расчет итоговой суммы (s), кнопки выхода и кнопки сброса (очистки данных).

3.Создайте приложение расчета стоимости магазинной покупки, состоящей из любых 3-ех вещей.

4. Рассчитайте уравнение $y=x^2+c^2$ -в.



Практическая работа №42 Разработка приложения с переключателями и флажками

Цель работы - создать компьютерную версию одной из головоломок Самуэля Ллойда: из заданного набора чисел надо выбрать те, сумма которых составит 50. Числа, которые избрал Ллойд для своей головоломки: 25, 27, 3, 12, 6, 15, 9, 30, 21, 19.

Выполнить следующие действия:

1. После запуска программы в окне изображается список чисел Ллойда.
2. Выбираем с помощью флажков числа и помещаем в правое окно.
3. Сумма выбранных чисел представлена в виде надписи.
4. Для выхода из программы необходимо щелкнуть мышью на закрывающей кнопке в строке заголовка.

Описание плана разработки программы

1. Открыть новый проект.
2. Разместить на форме экземпляры компонентов: список флажков CheckListBox, надпись Label, список ListBox.
3. Выполнить следующие действия:



Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Form1	Properties	Caption	Установка имени формы "Головоломка"
		BorderStyle	Задайте значение bsSingle
CheckListBox1 (Вкладка Additional)	Properties	Items	Задаем состав списка. Щелкнуть на кнопке построителя. Откроется окно String List editor (Редактор списка строк). Введите в список заданные числа через Enter. Нажмите кнопку OK.
		Height	Подобрать значение так, что все числа поместились в список (без полос прокруток).
	Events	OnClickCheck	Описать очистку списка. Проверить, установлен флажок или нет. После обновления списка необходимо подсчитать сумму выбранных чисел. Элементы списка выглядят как числа, но являются текстовыми строками (воспользоваться функцией StrToInt).

Продолжение таблицы 20.

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
ListBox1 (Вкладка Standard)	Properties	Height	Подобрать значение так, что все числа поместились в список (без полос прокруток). Первоначально список пуст. Заполняться он будет при работе программы.
Label1 (Вкладка Standard)	Properties	Caption	Введите текст: "Сумма: 0" Autosize
		Alignment	Задайте значение taCenter

4. проект, запустите и протестируйте его.

Сохраните

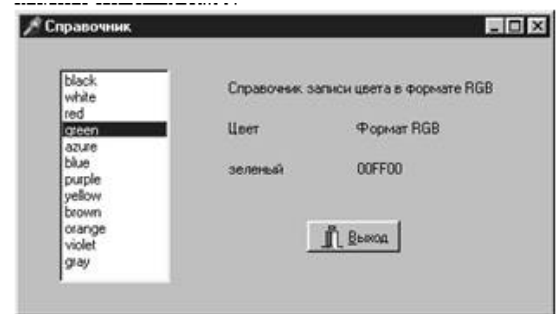
Листинг подпрограммы

```
procedure TForm1.CheckListBox1ClickCheck (Sender: TObject);
var i, s: Integer;
begin
  ListBox1.Clear;
  for i := 0 to CheckListBox1.Items.Count-1 do if CheckListBox1.Checked[i] then ListBox1.Items.Add (CheckListBox1.Items[i]);
  s := 0;
  for i := 0 to ListBox1.Items.Count -1 do
    s := s + StrToInt(ListBox1.Items[i]); Label1.Caption := 'Сумма: ' + IntToStr(s); if s = 50 then
      begin
        Label1.Caption := 'Сумма:' + IntToStr (s); CheckListBox1.Enabled := False; ListBox1.Enabled := False; end;
      end;
```

Практическая работа №43 Разработка приложения с простыми и комбинированными списками

"Разработка проекта СПРАВОЧНИК"

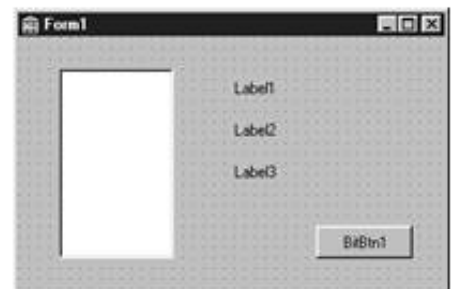
1. Постановка задачи: Создать программу, с помощью которой можно выполнить: после запуска программы пользователь выбирает с помощью мышки или стрелок название цвета и нажимает клавишу Enter. На экране появляется название цвета на русском языке и код в формате RGB. Программа заканчивает свою работу по нажатию клавиши Выход.



Пояснения Новыми в этой работе являются использование компоненты ListBox (список) на странице палитры компонентов Standard, использование встроенного редактора для ввода информации и алгоритм выбора (оператор Case).

2. План разработки программы

1. Открыть новый проект. Дать название форме Справочник.
2. Разместить в форме компоненты в соответствии с рисунком
3. Сохранить проект под именем, например, Pr3.pas и Pr.dpr
4. Запустить программу, затем закрыть окно проекта кнопкой Выход.
5. Выполнить следующие действия:

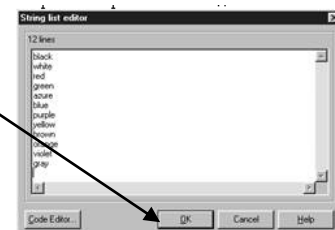


Выделенная компонента	Окно инспектора объектов	Имя свойства	Действие
BitBtn1	Properties	Caption	&Выход
		Kind	bkClose
Label1	Properties	Caption	Справочник: записи цвета в формате RGB
Label2	Properties	Caption	Цвет
Label3	Properties	Caption	Формат RGB
			Удалить название

6. Выделить ListBox1, в Object Inspector найти свойство Items, справа от него нажать на кнопку с тремя точками. В появившемся окне встроенного редактора (см. рисунок 9) ввести названия цветов, каждый на новой строке.

Сохраните набранный текст под именем Color.txt. Для этого нажмите правую клавишу мыши и выберите режим Save. Для выхода из встроенного редактора нажмите клавишу OK.

7. Выполнить следующие действия:



8. Сохранить проект окончательно, запустить и протестировать его.

. Задание

1. Изменить интерфейс программы: шрифты, цвет экрана и букв.

Подсказка. Возможно, придется в коде подкорректировать количество пробелов.

2. Сделать так, чтобы когда курсор мыши находится в поле ListBox1, появлялась подсказка о том, что надо сделать.

Подсказка. Воспользуйтесь свойствами Hint (текст сообщения), Showhint (показывать ли сообщение) компонента ListBox1.

Выделенная компонента	Окно инспектора объектов	Имя свойства	Действие
ListBox1	Events	OnKeyPress	<pre>if key=#13 then case ListBox1.ItemIndex of 0: Label3.Caption:='черный 000000'; 1: Label3.Caption:='белый FFFFFF'; 2: Label3.Caption:='красный FF0000'; 3: Label3.Caption:='зеленый 00FF00'; 4: Label3.Caption:='бирюзовый 00FFFF'; 5: Label3.Caption:='синий 0000FF'; 6: Label3.Caption:='фиолетовый FF00FF'; 7: Label3.Caption:='желтый FFFF00'; 8: Label3.Caption:='коричневый 996633'; 9: Label3.Caption:='оранжевый FF8000'; 10: Label3.Caption:='лиловый 8000FF'; 11: Label3.Caption:='серый A0A0A0'; end;</pre>

Практическая работа №44 Разработка приложения с меню, многостраничным блокнотом и статистическим арифметическим массивом

Тема Массивы

Цель работы Изучить различные алгоритмы сортировки списка. Закрепить навыки использования массивов данных и массивов объектов.

Пояснения к работе

Наиболее распространенным видом сортировки данных является упорядочивание массива – расположение записей сортируемого массива данных в порядке монотонного изменения некоторого ключевого признака. Сортировка данных позволяет сократить во много раз продолжительность решения задач, которые связаны с обработкой больших массивов информации. Смысл любой сортировки заключается в перестановке элементов таблицы в определенном заданном порядке. Упорядочивание осуществляется в процессе многократного просмотра исходного массива.

Сортировка простым выбором

1. Начинаем сортировку с первого элемента $i=1$;
2. Найти максимальный элемент и его номер в массиве A_1, A_2, \dots, A_n
3. Поменять местами A_1 и максимальный элемент A_k ;
4. перейти к следующему элементу $i=i+1$;
5. если рассмотрены не все $N-1$ элементы, то повторить с п. 2.

Пузырьковая сортировка (простой обмен)

1. $i=1$;
2. Сравнить попарно и, при нарушении упорядоченности, поменять местами элементы A_i и A_{i+1} , A_{i+1} и A_{i+2}, \dots, A_{n-1} и A_n
3. $i=i+1$
4. Если $i \leq N$, то повторить с п.2.

Простые вставки (добавление нового элемента в массив, не нарушая его упорядоченности)

1. Задать значение X ;
2. $i=1$;
3. Найти место для X , когда выполняется $A_i \leq X \leq A_{i+1}$;
4. Выполнить сдвиг элементов $A_{i+1} A_{i+2} \dots A_n$ вправо;
5. Поставить значение X на нужное место ($A_{i+1} = X$).

Постановка задачи: Разработать приложение, реализующее пузырьковую сортировку элементов одномерного массива. Предусмотреть визуализацию алгоритма сортировки через использование массива объектов. Объекты массива имеют различную высоту, каждая из которых соответствует значению элемента массива данных.

Порядок выполнения

1. Создать новый проект и установить на форме компоненты и изменить значения свойств в соответствии с таблицей:

Button1	Caption	Изменить
	Top	220
	Left	15
Button1	Caption	Сортировать
	Top	220
	Left	150
Timer1	Interval	0
Label1	Top	25
	Left	30
	Color	clBlue
	Width	15
	Height	100
	Name	Palka
	Visible	False

2. Объявить массив данных и массив объектов:

```
mas: Array[1..15] of Integer;  
zabor: Array[1..15] of TLabel;  
a,b,i: Integer;
```

3. Для формы выбрать событие **Create** и ввести код:

```
For i:=1 to 15 Do Begin  
zabor[i] := TLabel.Create(palka);
```

```

zabor[i].Parent := Form1;
zabor[i].Left := i*30;
zabor[i].Top:=palka.Top;
zabor[i].Visible :=true;
zabor[i].Width :=palka.Width ;
zabor[i].Height :=palka.Height ;
zabor[i].Color :=palka.Color ;
end;
b:=0; //количество упорядоченных элементов
a:=1 ;//номер элемента, с которого начинается сравнение пар

```

4. Для кнопки **Изменить** ввести код:

```

Randomize;
For i:=1 To 15 Do
Begin
zabor[i].Height :=random(200)+10;
mas[i]:=zabor[i].Height ;
end;

```

5. Для кнопки **Сортировать** ввести код

```

timer1.Interval :=1000;

```

6. Для таймера ввести код:

```

Var
c:integer;
Begin
For i:=a to 14 do
If mas[i]>mas[i+1] then
Begin
c:=mas[i];
mas[i]:=mas[i+1];
mas[i+1]:=c;
zabor[i].Height :=mas[i];
zabor[i+1].Height :=mas[i+1];
a:=i+1;
exit;
end;
a:=1;
b:=b+1;
if b>14 then timer1.Interval :=0;
End;

```

7. Проверить работу приложения.
8. Модернизировать приложение:
- вывести сообщение об окончании сортировки;
 - создать еще один массив меток для вывода значений элементов массива;
 - добавить две радиокнопки и внести изменения в программу для выбора режима сортировки: по возрастанию и по убыванию.
9. Результат показать преподавателю.

Задания для самостоятельного выполнения

Задание №1

Модернизировать разработанное приложение: первую половину массива сортировать по возрастанию, а вторую половину массива по убыванию.

Практическая работа №45 Разработка приложения со статистическим массивом записей

Тема Проектирование многооконных приложений

Цель работы: Получить навыки добавления новых форм к проекту.

Пояснения к работе

Проект приложения, включающий несколько окон, создается поэтапно. Шаги подключения новой формы к проекту:

- создание новой формы в действующем проекте
File / New/ Form
- сохранение формы, как файла модуля *.pas
- подключение нового файла к проекту
Project / Add to Project..., открыть
- создание ссылки на новый модуль, с той формы, которая будет его использовать
File / UseUnit,
в открывшемся диалоговом окне выбрать нужный модуль и нажать **Ок**.

Методы, используемые при работе с формой

имя_формы.**Show** – отображение на экране окна формы, до этого невидимого;
имя_формы.**Hide** – скрыть форму, вернуться на предыдущую;
имя_формы.**Update** принудительная прорисовка формы;
имя_формы.**Free** удаление из памяти формы.

Компонент StringGrid

Компонент **StringGrid** расположен на вкладке **Additional** и представляет собой таблицу, ячейки которой содержат строки символов. Свойства компонента приведены в таблице:

ColCount	Количество столбцов таблицы
RowCount	Количество строк таблицы
DefaultColWidth	Ширина столбцов
DefaultRowHeight	Высота строк
FixedCols, FixedRows	Количество зафиксированных слева столбцов, сверху строк
Cells	Обращение к ячейке таблицы
Options.goEditing	Признак допустимости редактирования содержимого ячеек

Постановка задачи

Разработать приложение «Тригонометрические функции». Приложение должно работать следующим образом:

- после загрузки на экран выводится окно-заставка, которое отображается 5 сек;
- через 5 секунд заставка пропадает, на экран выводится основное окно;
- в основном окне отображается таблица со значениями тригонометрических функций с заданным шагом.

Порядок выполнения

- Загрузить Delphi, сохранить проект.
- Изменить свойства главной формы в соответствии с таблицей:

BorderIcons	biMaximize=False
BorderStyle	bsSingle
Position	poScreenCenter

- Добавить на форму компонент **StringGrid** (Additional) и изменить свойства в соответствии с таблицей:

Align	alClient
ColCount	5
RowCount	2
DefaultColWidth	120
DefaultRowHeight	18
ScrollBars	ssVertical

Options.goRangeSelect	False
-----------------------	-------

4. Для события **OnCreate** ввести код:

```

Var
t:longint; i,n:integer; step,x,sx,cx: double;
Begin
step:=0.01;
n:=round(90/step)+1;
With stringgrid1 do
  Begin
    cells[0,0]:='x'; cells[1,0]:='sin(x)';
    cells[2,0]:='cos(x)'; cells[3,0]:='tg(x)';
    cells[4,0]:='ctg(x)'; colwidths[0]:=colwidths[0] div 2;
    rowcount:=n+1;
    For i:=1 to n do
      Begin
        x:=(i-1)*step;
        sx:=sin(x*pi/180); cx:=cos(x*pi/180);
        cells[0,i]:=floattostr(x); cells[1,i]:=floattostr(sx);
        cells[2,i]:=floattostr(cx);
        if cx <> 0 then cells[3,i]:=floattostr(sx/cx)
        else cells[3,i]:= не существует;
        if sx<>0 then cells[4,i]:=floattostr(cx/sx)
        else cells[4,i]:= 'не существует';
      end;
    end;
  end;

```

5. Проверить работу приложения.

6. Добавить новую форму (заставка), для этого:

- выбрать в меню **File / New/ Form**;
- сохранить новую форму с именем **Unit2.pas**;
- подключить новый файл к проекту, выбрав в меню **Project / Add to Project...**, открыть **Unit2.pas**;
- создать ссылку на новый модуль, с той формы, которая будет его использовать:
 - выбрать в меню **File / Use Unit**,
 - в открывшемся диалоговом окне выбрать модуль **Unit2** и нажать **Ок**.

7. Изменить свойства **Form2** в соответствии с таблицей:

BorderIcons	biSystemMenu=False biMinimize=False biMaximize=False
BorderStyle	bsNone
Position	poScreenCenter
FormStyle	fsStayOnTop

8. На форму **Form2** добавить метку, ввести для нее заголовок **ТРИГОНОМЕТРИЧЕСКИЕ ФУНКЦИИ**

9. Загрузить в редактор файл проект, для этого выбрать в меню **Project/ View Source**.

10. Откорректировать файл проекта следующим образом:

```
program Project1;
uses
  Forms,
  Unit1 in 'Unit1.pas' {Form1},
  Unit2 in 'Unit2.pas' {Form2};
{$R *.res}
Begin
  Application.Initialize;
  form2:=tform2.create(application); form2.Show ;
form2.Update ; form2.Label1.Update ;
  Application.CreateForm(TForm1, Form1);
form2.Hide ; form2.Free;
  Application.Run;
end.
```

11. Для того, чтобы заставка задерживалась на экране 5 секунд в начало процедуры **Form1Create** добавить:

```
t:=gettickcount div 1000;
while (gettickcount div 1000)<t+5 do
```

12. Проверить работу приложения

Практическая работа №46 Разработка приложения с компонентами главного меню

Цель работы -Создать программу, выполняющую следующие действия:

1. После запуска программы в окне изображается строка меню (Файл, Выход).
2. При выборе пункта меню Файл появляются пункты меню (Рисунки, Выход).
3. При выборе пункта меню Рисунки появляется вложенное меню, состоящее из двух пунктов (Облака, Лес).
4. По щелчку правой кнопки мыши появляется контекстное меню.
5. Выбрать по пункту другой рисунок
6. Для выхода из программы необходимо щелкнуть мышью на закрывающей кнопке в строке заголовка.
7. Если выбрать любой из пунктов Выход, работа программы завершается.



Описание плана разработки программы

1. Открыть новый проект.
2. Разместить на форме экземпляры компонентов: панель Panel, рисунок Image, диалоговое окно OpenFileDialog.
3. Выполнить следующие действия:

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
Form1	Properties	Caption	Установка имени формы "Мое меню"
	Events	OnMouseDown	<pre>var p:TPoint; begin p.X :=X; p.Y :=Y; p := ClientToScreen (p); PopupMenu1.Popup (p.X, p.Y); end;</pre>
Запустить редактор меню (дважды щелкнуть на значке меню на форме)			
Form1.MainMenu1	Properties (в окне Object Inspector не выбран никакой объект)	Caption	Ввести текст пункта меню -Файл, и нажать Enter. Система присвоит ему имя N1
Между существующими и будущими пунктами меню можно переключаться с помощью щелчка мыши или курсорных клавиш.			
Form1.MainMenu1	Properties	Caption	Ввести текст пункта меню -Выход, и нажать Enter. Система присвоит ему имя N2.
	Events (щелкнуть на пункте Выход в строке меню)	N2Click	Close;
Щелкните на пункте Файл. Редактор меню создал еще одну заготовку под этим пунктом. Это заготовка для меню, которое откроется при выборе пункта Файл в работающей программе. Используя заготовки, создайте в этом меню два пункта: Рисунки (система присвоит ему имя N3) и Выход (N4). Выберите в редакторе меню пункт Рисунки и нажмите комбинацию клавиш Ctrl + Вправо.			

N4: TMenuItem	Events	OnClick	Выберем из раскрывающегося списка уже существующую процедуру-обработчик N2Click
Form1.MainMenu1	Properties	Caption	Ввести текст пункта меню -Облака, и нажать Enter. Система присвоит ему имя N5.
N5: TMenuItem	Events (выбрать в строке меню на форме пункт Облака)	OnClick	Image1.Picture.LoadFromFile ('C:\Windows\Облака.bmp');
Form1.MainMenu1	Properties	Caption	Ввести текст пункта меню - Лес, и нажать Enter. Система присвоит ему имя N6.
N6: TMenuItem	Events (выбрать в строке меню на форме пункт Лес)	OnClick	Image1.Picture.LoadFromFile ('C:\Windows\Лес.bmp');
Закройте окно редактора меню и убедитесь, что теперь строка меню появилась в основной форме программы.			

Продолжение таблицы 7.

Выделенный объект	Вкладка окна Object Inspector	Имя свойства/ имя события	Действие
PopupMenu (Вкладка Standard)	Properties	Caption	Ввести текст пункта меню -Облака, и нажать Enter. Система присвоит ему имя N7.
		Caption	Ввести текст пункта меню - Лес, и нажать Enter. Система присвоит ему имя N8.
N7	Events	OnClick	Выберем из раскрывающегося списка уже существующую процедуру-обработчик N5Click
N8	Events	OnClick	Выберем из раскрывающегося списка уже существующую процедуру-обработчик N6Click
Image (Вкладка Additional)	Properties	Stretch	Присвоить значение True

4. Сохраните проект, запустите и протестируйте его.

Листинг подпрограммы

```

procedure TForm1.N2Click (Sender: TObject);
begin
Close; end;
procedure TForm1.N5Click (Sender: TObject); begin
Image1.Picture.LoadFromFile ('C:\Windows\Облака.bmp'); end;
procedure TForm1.N6Click (Sender: TObject); begin
Image1.Picture.LoadFromFile ('C:\Windows\Лес.bmp'); end;
procedure TForm1.FormMouseDown
(Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer); var p:TPoint; begin
p.X :=X; p.Y :=Y;
p := ClientToScreen (p); PopupMenu1.Popup (p.X, p.Y); end;

```

Практическая работа №47 Разработка приложения с таблицей для выбора допустимых значений.

Тема: Разработка тестового приложения.

Цель работы: Закрепить навыки программирования в Delphi.

Постановка задачи:

Текстовый файл содержит несколько вопросов и 4 варианта ответа, из которых только один ответ верный(помечен символом +). Файл имеет следующую структуру:

Тема теста (дисциплина)

/вопрос 1

- ответ 1

+ ответ 2

- ответ 3

- ответ 4

/вопрос 2

+ вопрос 1

- вопрос 2

- вопрос 3

- вопрос 4

...

Создать текстовый файл с тестом из 5 вопросов в соответствии с заданной структурой.

Разработать приложение, которое позволяет выбирать файл с тестовыми заданиями и проводить последовательное тестирование.

Предусмотреть вывод результат тестирования на экран

Порядок выполнения:

1. Загрузить Delphi;
2. Установить на форму следующие объекты и изменить им свойства (см. таблицу), объявить используемые переменные как глобальные.

объект	свойство	значение
Label	Name	Label1
	WordWrap	True
RadioButton	Name	RadioButton1
Button	Name	Button1
	Caption	Далее
OpenDialog	Name	OpenFile

3. Для формы выбрать событие **Create** и ввести код для создания массива радиокнопок и объявить необходимые переменные:

```
For i:=1 to 4 Do
  Begin
    otv[i] := TRadioButton.Create(RadioButton1);
    otv[i].Parent := Form1;
    otv[i].Left := RadioButton1.Left;
    otv[i].Top:= RadioButton1.Top+i*40;
    otv[i].Visible :=true;
    otv[i].Width := RadioButton1.Width;
    otv[i].Height := RadioButton1.Height;
    otv[i].Color := RadioButton1.Color;
    otv[i].Checked :=false;
    otv[i].Font := RadioButton1.font;
  end;
RadioButton1.Visible :=false;
```

4. Добавить на форму меню из двух пунктов **Файл / Открыть и Выход**
5. Для пункта**Открыть** ввести код:

```
If openfile.Execute Then
  Begin
    assignfile(f1,openfile.FileName);
    reset(f1);
    readln(f1,s);
    form1.Caption :=s; { тематеста }
```

```

verno:=0;//кол-во верных ответов
n:=0; //номер текущего вопроса
button1.Enabled :=true;
label1.Caption :=";
for i:=1 to 4 do
begin
otv[i].Caption :=";
otv[i].Checked :=false;
end;
end;

```

6. Для кнопки **Далее** ввести код:

```

if n>0 then {проверим ответ на предыдущий вопрос}
for i:=1 to 4 do
if (otv[i].Checked =true) and (otv[i].tag=1) then verno:=verno+1;
ifeof(f1) then {если тест закончен}
begin
showmessage(' кол-во верных ответов='+inttostr(verno));
closefile(f1);
button1.Enabled :=false;
exit;
end;
n:=n+1;
{÷читаем следующий вопрос}
readln(f1,s);
delete(s,1,1);
label1.Caption :=s;
for i:=1 to 4 do {варианты ответов}
begin
readln(f1,s);
otv[i].Checked :=false;
otv[i].Tag:=0;
if s[1]='+' then otv[i].Tag:=1;{запомним верный ответ}
delete(s,1,1);
otv[i].Caption :=s;
end;;

```

7. Проверить работу приложения.
8. Модернизировать приложение: вывод каждого вопроса должен сопровождаться отображением картинки, имя графического файла для каждого вопроса записать в строке, следующей за вопросом.
9. Результат показать преподавателю.

Критерии оценки по выполнению практических работ

Практическая работа считается выполненной по следующим критериям:

Оценка «отлично» выставляется, если студент обстоятельно, с достаточной полнотой излагает программный материал, дает правильные формулировки, точные определения ключевых понятий, обнаруживает полное понимание материала и может обосновать свой ответ, привести примеры, демонстрирует самостоятельность мышления, правильно отвечает на дополнительные вопросы, качественно оформляет электронную документацию.

Оценка «хорошо» выставляется, если студент дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает единичные ошибки, которые сам же исправляет после замечаний преподавателя, качественно оформляет электронную документацию.

Оценка «удовлетворительно» выставляется, если студент демонстрирует знание и понимание основных положений программного материала, но при этом допускает неточности в формулировке правил или определений, излагает материал недостаточно связно и последовательно, имеются небольшие недочеты в оформлении электронной документации (несоответствие размера или типа шрифта, наличие лишних пустых строк и т.д.).

Оценка «неудовлетворительно» выставляется, если студент обнаруживает незнание большей части программного материала, допускает ошибки в формулировке правил и определений, искажающие их смысл, беспорядочно и неуверенно излагает материал, сопровождая изложение частыми запинками, перерывами, имеются серьезные недочеты в оформлении электронной документации (практическая работа выполнена не полностью).