Лабораторная работа №1

**Тема 1:** Стандартные функции. Арифметические выражения.

*(в лекциях см. п. 1.4, 2.9 )*

**Тема 2:** Логические выражения. Условный оператор.

*(в лекциях см. п. 2.4, 3.3 )*

**Пример задания:**

1. Вычислить значение арифметического выражения следующего вида:

|  |  |
| --- | --- |
|  | 2y при 0 < y < = 20 |
| 2. Вычислить значение функции f = | y+2 при y = 0 |
|  | у/2 при у < 0 или y > 25 |

3. Вывести на экран значения y и f.

**Варианты программы:**

**1.**

Var A, B, C, D, Y ,f: Real;

Begin

A:=1.5; B:=4.3; C:=10; D:=7.1; {ввод данных}

Y:=(A\*Sqrt(Abs(Sin(A+B\*C)))-Exp(-A\*C))/Sqrt(Abs(2\*B+D));

WriteLn (‘Y=’, Y:8:3); {вывод результата}

f:=0; {перед началом вычисления значению f присваивается 0 (можно любое

другое значение) для того, чтобы в случае, если ни одно из условий не

выполнится, на экран вывелось f = 0, т.е. чтобы значение f не осталось

неопределенным }

if (y>0) and (y<=20) then f:=2\*y;

if (y=0) then f:=y+2;

if (y<0) or (y>25) then f:=y/2;

WriteLn ('f=', f:8:3);

End.

**2.**

Const D=10;

Var A, B, C, Y ,f: Real;

Begin

randomize;

A:=random;

Write('введите числa B и C типа real:');

Read(B,C);

Y:=(A\*Sqrt(Abs(Sin(A+B\*C)))-Exp(-A\*C))/Sqrt(Abs(2\*B+D));

WriteLn ('Y=', Y:8:3); { вывод результата}

if (y>0) and (y<=20) then f:=2\*y

else if (y=0) then

f:=y+2

else

if (y<0) or (y>25) then f:=y/2

else f:=0;

WriteLn ('f=', f:8:3);

End.

***Замечание:*** В первом варианте программы выполняется три проверки условия (т.е. три оператора if ) даже если например уже первое условие будет истинным (хотя понятно, что в этом случае другие условия точно будут ложными и нет необходимости в их проверке). Во втором варианте используется один вложенный оператор if, что позволяет избежать лишних проверок.

**Задание:**

Разработать программу для вычисления:

1) значения заданного арифметического выражения (**см. вариант по таблице 1**);

2) значения заданной функции (**см. вариант по таблице 2**);

и вывода на экран полученных результатов.

Значения исходных данных выбираются произвольно. Ввод исходных данных организовать любым известным вам способом (использовать не менее двух способов).

**Таблица 1 – Варианты арифметических выражений**

|  |  |  |
| --- | --- | --- |
| Номер варианта | Выражение | Исходные данные |
| 3 |  | b, x, y |

**Таблица 2 – Варианты функций**

|  |  |  |
| --- | --- | --- |
| 3 | f = | |x|-|y|, при 0.5<=xy<1.0; |
| x+y, при 0.1<xy<0.5;2x, при 1.0<=xy<2.5. |

Лабораторная работа №2

**Тема 1:** Операторы циклов с условиями: While и Repeat

*(в лекциях см. п. 3.6, 3.7)*

**Задание:**

Написать программу для вычисления заданного выражения и вывода на экран полученного значения, используя заданный циклический оператор (**см. вариант по таблице 1**);

**Таблица 1.**

|  |  |  |
| --- | --- | --- |
| Номер варианта | Выражение | Оператор цикла |
| 3 | 0,15 - 0,17 + 0,19 -...- 0,33 | Repeat |

Лабораторная работа №3

**Тема 1:** Работа с одномерными массивами.

*(в лекциях см. п. 4.1 )*

**Тема 2:** Подпрограммы: процедуры и функции.

*(в лекциях см. п. 5 )*

**Пример задания:**

Массив T=(6, –3, 14, 7, –2, –4, 12, 0) преобразован к виду T=(–1, –10, 7, 0, –9, –11, 5, –7). Определите правило преобразования и напишите программу, которая:

1) по такому же алгоритму преобразовывает произвольный массив T из 38 элементов и выводит на экран элементы преобразованного массива.

2) вычисляет сумму тех элементов преобразованного массива, которые находятся в диапазоне [–14, 2], и выводит результат на экран.

*Указания*:

- сформировать массив T из 38 случайных целых чисел в диапазоне [–30,30] и вывести элементы массива на экран;

- преобразование массива оформить в виде процедуры;- вычисление суммы оформить в виде функции;

- все выводы на экран осуществлять в основной программе.

**Варианты программы:**

uses crt; {подключение модуля CRT, чтобы можно было использовать

содержащиеся в нем стандартные процедуры clrscr и readln}

const kol\_el=38; {задаем количество элементов массива;

при отладке программы удобнее задать меньшее количество элементов }

type mas=array [1..kol\_el] of integer; {создаем новый тип - массив}

var T:mas;

Procedure Preobr(var a:mas); {процедура, преобразующая переданный в качестве

параметра массив по заданному правилу}

var i:integer; {Здесь i – локальная переменная, она доступна только в теле данной процедуры.

Переменная i, описанная в основной программе –глобальная и доступна в любом месте программы, но только если

нет локальной переменной с тем же именем. В этой процедуре можно было бы и не описывать локальную i,

а использовать глобальную, но хороший стиль программирования предполагает,

что все переменные, которые используются только для выполнения каких-либо действий внутри подпрограммы

должны быть в ней и описаны, чтобы избежать различных ошибок}

begin

for i:=1 to kol\_el do

a[i]:=a[i]-7;

end;

Function Sum(b:mas):integer;

var i,s:integer;

begin

s:=0;

for i:=1 to kol\_el do

s:=s+b[i];

sum:=s; {обязательное присвоение имени функции полученного результата}

end;

var i:integer; {описание i:integer; и Т:mas можно было бы расположить в одном разделе Var }

Begin

clrscr;

Randomize;

for i:=1 to kol\_el do begin

T[i]:=Random(61)-30;

write(T[i]:4);

end;

Preobr(T);{при вызове процедуры происходит следующее:

- в памяти создается копия массива T – массив a (формальный параметр процедуры),

- выполняется тело процедуры, в котором преобразуется массив a;

- возврат в основную программу на оператор, следующий за вызовом процедуры;

При возврате в массив T переносятся значения преобразованного массива a }

writeln;

writeln;

for i:=1 to kol\_el do

write(T[i]:4); {вывод преобразованного массива}

{вызов функции Sum, выполнение тела функции,

возврат в программу, вывод результата на экран: }

write(#13,#10,#10,'Summa elementov = ',Sum(T));

readln;

End.

Еще некоторые пояснения к программе:

Оператор write(#13,#10,#10,'Summa elementov = ',Sum(T)); можно было бы заменить последовательностью из трех следующих операторов:

writeln;

writeln;

write('Summa elementov = ',Sum(T));

Результат один и тот же: вывод текста Summa elementov = … будет осуществлен в начале новой строки через строчку от предыдущего вывода.

#10 - перевод строки

#13 - переход к началу строки (*см. п.2.3 лекций*)

Т.е. оператор write(#13,#10,#10,'Summa elementov = ',Sum(T)); выполняется следующим образом: сначала курсор перемещается в начало текущей строки, затем переводится на следующую строку в ту же позицию, снова переводится на следующую строку в ту же позицию, и только потом выводится текст.

**Общий текст задания для всех вариантов:**

Задана последовательность значений элементов некоторого массива до и после преобразования по некоторому правилу. Определите алгоритм преобразования и напишите программу, которая:

1) формирует массив из заданного количества случайных целых чисел в заданном диапазоне и выводит элементы массива на экран;

2) по определенному вами алгоритму преобразовывает этот массив и выводит на экран элементы преобразованного массива.

3) производит заданные вычисления и выводит результат на экран.

*Указания*:

- преобразование массива оформить в виде процедуры;

- заданные вычисления оформить в виде функции;

- все выводы на экран осуществлять в основной программе.

**Варианты 3:**

Массив C = (–4, 9, 5, –8, –5, 4, 0) преобразован к виду C = (–8, 0, 0, –16, –10, 0, 0). Размер массива C - 40 элементов из диапазона [–23,23]. Вычислить сумму нечетных элементов преобразованного массива.