**Лабораторная работа № 6.**

Тема: Детерминированные ЦВП с управлением по индексу.

Одномерные массивы.

Цель: научить работать с одномерными массивами в Lazarus

Используемое оборудование: ПК, среда программирования Lazarus, сайт draw.io

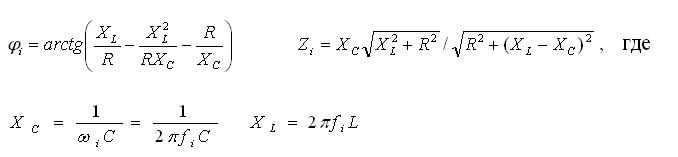
*Далее к каждому заданию будет начинаться с пункта №4(постановка задачи).*

**Задание 1.**

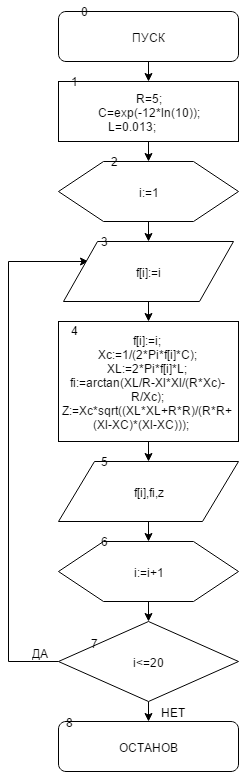
**Постановка задачи:**

Исследовать характер изменения фазового угла  j и реактивного сопротивления колебательного контура Z на различных частотах  f задана массивом значений

**Математическая модель:**



**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| R | integer | Сопротивление, константа |
| C | real | Ёмкость конденсатора, константа |
| L | real | Индуктивность, константа |
| Xc | real | Ёмкостное сопротивление |
| XL | real | Индуктивное сопротивление |
| fi | real | Фазовый угол |
| Z | real | Реактивное сопротивление |
| f | array of real | Массив значений частот |
| i | integer | Параметр цикла |

**Код программы:**

program zad1;

const

R=5;

C=exp(-12\*ln(10));

L=0.013;

var

Xc,XL, fi, Z:real;

f:array[1..20] of real;

i:byte;

begin

for i:=1 to 20 do begin

f[i]:=i;

Xc:=1/(2\*Pi\*f[i]\*C);

XL:=2\*Pi\*f[i]\*L;

fi:=arctan(XL/R-Xl\*Xl/(R\*Xc)-R/Xc);

Z:=Xc\*sqrt((XL\*XL+R\*R)/(R\*R+(Xl-XC)\*(Xl-XC)));

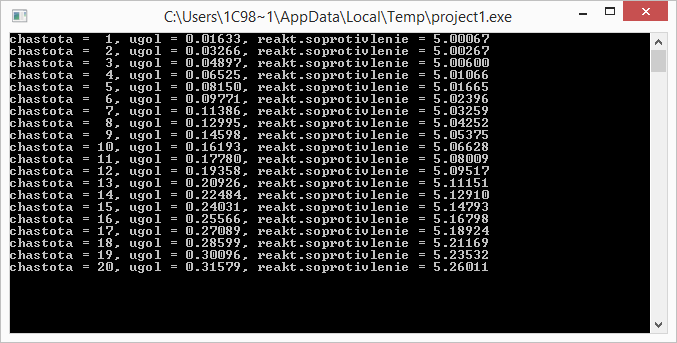
writeln('Pri ugle, ravnom ',fi:1:5,' gradusov, ', 'Z=',z:1:5);

end;

readln();

end.

**Результат работы программы:**



**Анализ:** Мы написали программу, которая выводит значения угла и реактивного сопротивления при частоте от 1 до 20 включительно.

**Вывод:** Чем больше частота, тем больше угол и реактивное сопротивление.

**Задание 2.**

**Постановка задачи:**

Одномерный массив задан случайным образом. Организовать вывод массива.

Найти сумму его элементов, вычислить факториал элементов массива, среднее арифметическое его элементов.

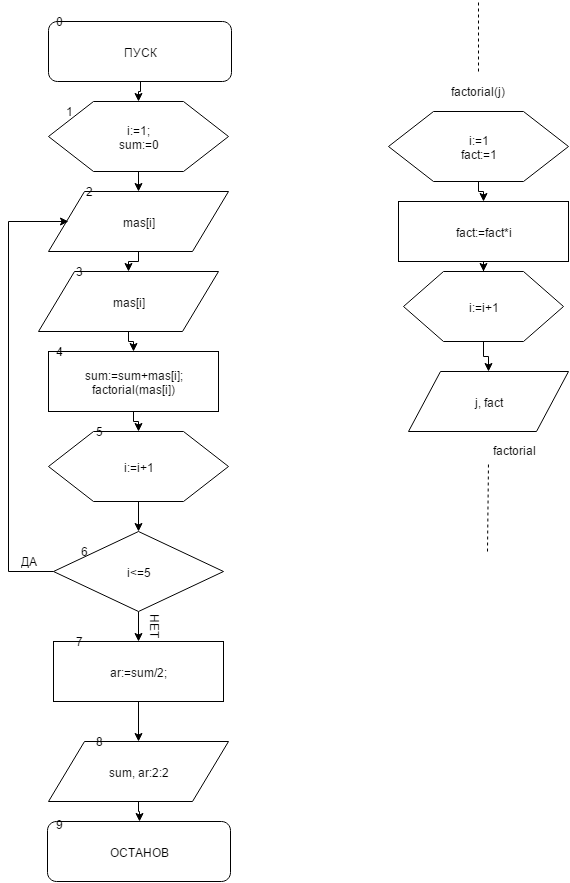
**Математическая модель:**

**fact=f\*n; fact=fact+1;**

**sum=sum+mas[i];**

**ar=sum/5;**

**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| mas | array of integer | Массив |
| ar | real | Среднее арифметическое |
| i | integer | Параметр цикла |
| sum | integer | Сумма |
| j | integer | Параметр цикла функции |
| factorial | longint | Пользовательская функция |
| fact | integer | Значение факториала |

**Код программы:**

program zadanie2;

var mas:array[1..5] of integer;

i,sum,j:integer;

ar:real;

function factorial(j:integer):longint;

var i,fact:integer ;

begin

fact:=1;

for i:=1 to j do

fact:=fact\*i;

Writeln('Factorial ',j,' raven ',fact)

end;

begin

Randomize;

sum:=0;

for i:=1 to 5 do begin

mas[i]:=Random(10);

write(mas[i],' ');

sum:=sum+mas[i];

factorial(mas[i]);

end;

ar:=sum/5;

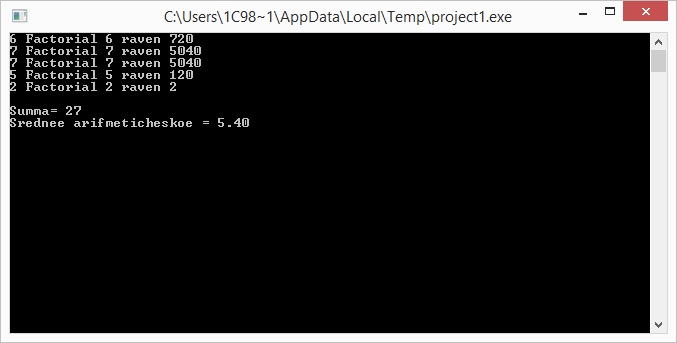
Writeln();

Writeln('Summa= ',sum);

Writeln('Srednee arifmeticheskoe = ',ar:2:2) ;

readln();

end.

**Результат работы программы:**

**Анализ:** Мы написали программу, которая генерирует одномерный массив с размерностью 5, считает считает факториал каждого элемента и вычисляет среднее арифметическое всех элементов массива.

**Вывод:** Мы написали программу, которая генерирует одномерный массив с размерностью 5, считает считает факториал каждого элемента и вычисляет среднее арифметическое всех элементов массива.

**Задание 3.**

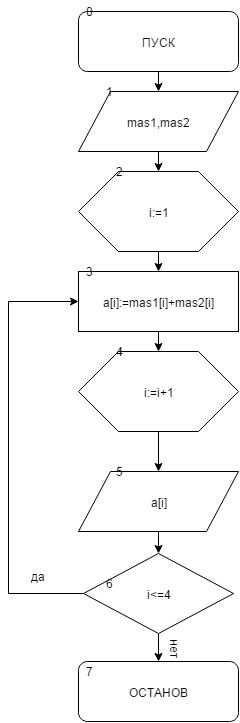
**Постановка задачи:**

Заданы два одномерных массива (задать явно внутри программы). Составить третий одномерный массив, элементы которого являются суммой соответствующих элементов первого и второго массивов.

**Математическая модель:**

**c[i]=a[i]+b[i]**

**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Смысл** |
| mas1 | array of integer | Массив №1, заданный явно внутри программы |
| mas2 | array of integer | Массив №2, заданный явно внутри программы |
| a | array of integer | Результирующий массив |
| i | integer | Параметр цикла |

**Код программы:**

program zad3;

const

mas1:array[1..4] of integer = (1,2,3,4);

mas2:array[1..4] of integer = (5,6,7,8);

var

i:integer;

a:array[1..4] of integer;

begin

for i:=1 to 4 do

a[i]:=mas1[i]+mas2[i];

Writeln(a);

end.

**Результат работы программы:**



**Анализ:** Мы написали программу, которая складывает поэлементно два массива и выводит результирующий массив a

**Вывод:** Мы научились писать программы, которые вычисляют сумму элементов одного массива с соответствующими элементами другого массива и выводить результирующий массив на экран.

**Задание 4.**

**Постановка задачи:**

Одномерный массив вводится пользователем с клавиатуры.  Переставить элементы массива, стоящие на четных и нечетных местах. задачу решить без проверки на четность индексов массива.

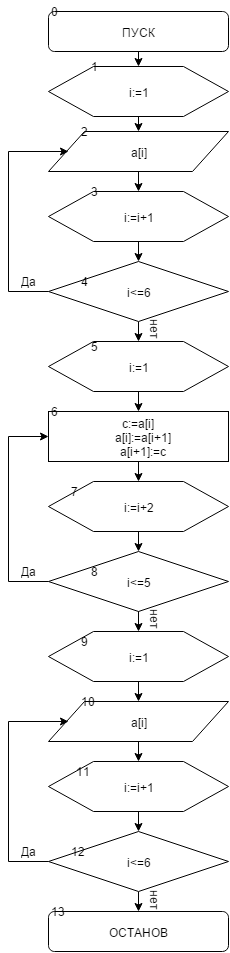
**Математическая модель:**

a[1],a[2],a[3],a[4],a[5],a[6] → a[2],a[1],a[4],a[3],a[6],a[5].

C:=a[i]

a[i]:=a[i+1]; a[i+1]:=C.

**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Смысл** |
| a | array of integer | Массив |
| i | integer | Параметр цикла |
| c | integer | Переменная, введённая для возможности хранения очередного значения a[i] |

**Код программы:**

program zad\_4;

var

a:array[1..6] of integer;

i,c:integer;

begin

for i:=1 to 6 do

readln(a[i]);

i:=1;

while i<=5 do

begin

c:=a[i];

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

a[i+1]:=c;

i:=i+2;

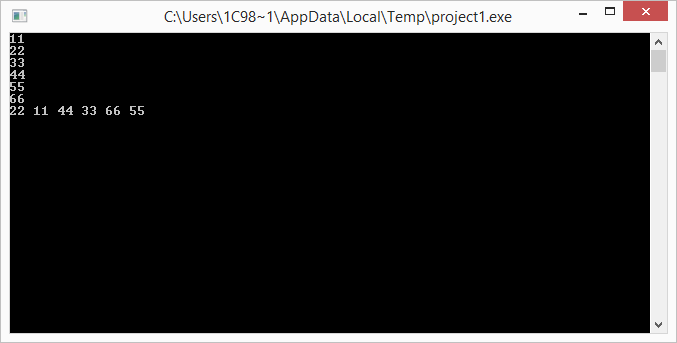
end;

for i:=1 to 6 do write(a[i], ' ');

readln();

end.

**Результат работы программы:**



**Анализ:**

Мы написали программу, которая меняет местами четные элементы массивов с нечетными поочередно, то есть порядок до изменение: a[1],a[2],a[3],a[4],a[5],a[6]. А порядок после:

a[2],a[1],a[4],a[3],a[6],a[5].

**Вывод:** Мы написали программу, которая меняет местами четные элементы массивов с нечетными без использования проверки индекса на чётность.

**Задание 5.**

**Постановка задачи:**

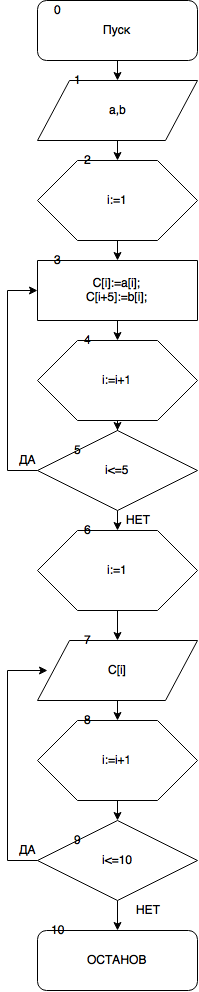
Заданы массивы A(5) и B(5). Получить массив C(10), расположив в начале его элементы массиваA, а затем – элементы массива B. Для формирования массива С использовать один цикл.

**Математическая модель:**

**C[i]:=a[i];**

**C[i+5]:=b[i]**

**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| a | array of integer | Первый заданный массив |
| b | array of integer | Второй заданный массив |
| c | array of integer | Массив, идуший на выход на экран, состоящий из массивов a и b |
| i | array of integer | Параметр цикла, индекс массива |

**Код программы**

program zad5;

const

a:array [1..5] of integer = (3,6,2,13,9);

b:array [1..5] of integer = (8,15,1,3,6);

var

i:integer;

c:array[1..10] of integer;

begin

for i:=1 to 5 do

begin

C[i]:=a[i];

C[i+5]:=b[i];

end;

for i:=1 to 10 do

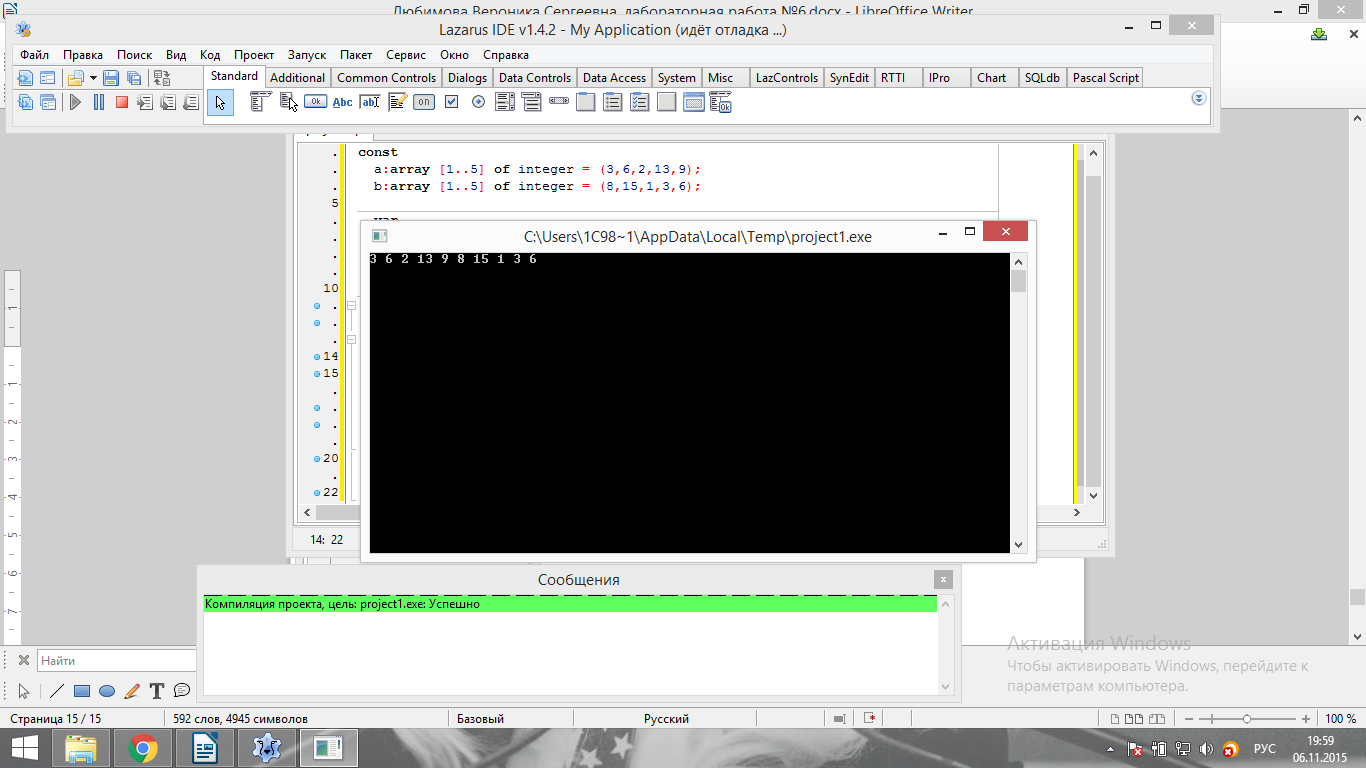
writeln(c[i]);

readln();

end.

.

**Результат работы программы:**



**Анализ:** Мы написали программу, которая заполняет массив C элементами из массивов a и b так, что C[1]-C[5]=a[1]-a[5], а c[6]-c[10]=b[1]-b[5]

**Вывод:**

Мы написали программу, в которой массив c заполняется сначала массивом а, а потом массивом b.