# Лабораторная работа № 7

# Итерационные циклические вычислительные процессы.

## Цель работы: научиться реализовывать итерационные ЦВП средствами Free Pascal.

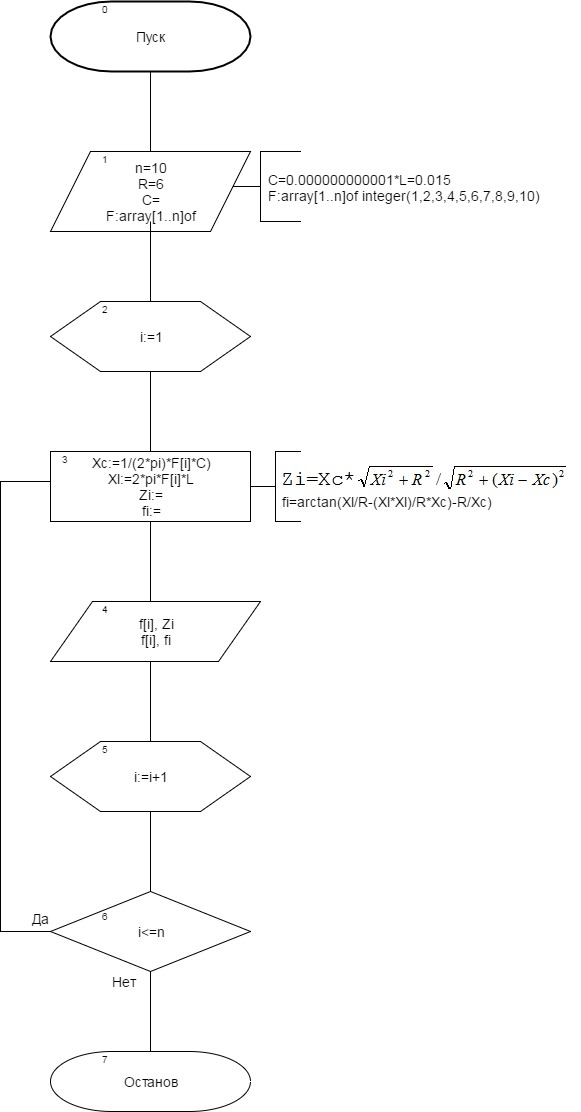
## Оборудование: PC, Lazarus

# Задача № 1

Постановка задачи: Вычислить 2 в степени n и при этом определить первое значение степени, при котором результат будет превышать значение 1000. Для данной задачи написать максимально возможное количество вариантов программ, используя разные виды циклов.

## Математическая модель:

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 1*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| n | Кол-во элементов массива, частота | Integer |
| R | Сопротивление | Integer |
| С | Емкость конденсатора | Real |
| L | Индуктивность | Real |
| Zi | Реактивное сопротивление | Real |
| Xc | Емкостное сопротивление | Real |
| Xl | Индуктивное сопротивление | Real |
| fi | Фазовый угол | Real |
| F | Массив | Integer |

Код программы:

Program Zadanie1;

const n=10;

R=6;

C=0.000000000001;

L=0.015;

F:array[1..n] of integer=(1,2,3,4,5,6,7,8,9,10);

var i: integer;

Zi,Xc,Xl,fi: real;

begin

for i:=1 to n do

begin

Xc:=1/(2\*pi\*F[i]\*C);

Xl:=2\*pi\*F[i]\*L;

Zi:=Xc\*sqrt(Xl\*Xl+R\*R)/(sqrt(R\*R+(Xl-Xc)\*(Xl-Xc)));

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

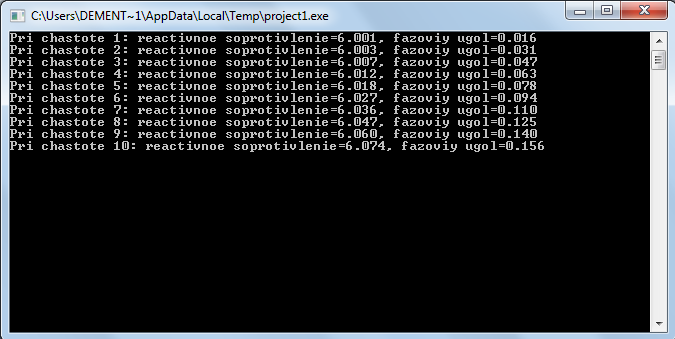
writeln('Pri chastote ', F[i], ': reactivnoe soprotivlenie=', Zi:1:3, ', fazoviy ugol=', fi:1:3);

end;

readln

end.

## Результаты выполненной работы:



## Анализ результатов вычисления: Программа вычисляет и выводит на экран изменения фазового угла и реактивного сопротивления колебательного контура на различных частотах, которые заданы массивом значений.

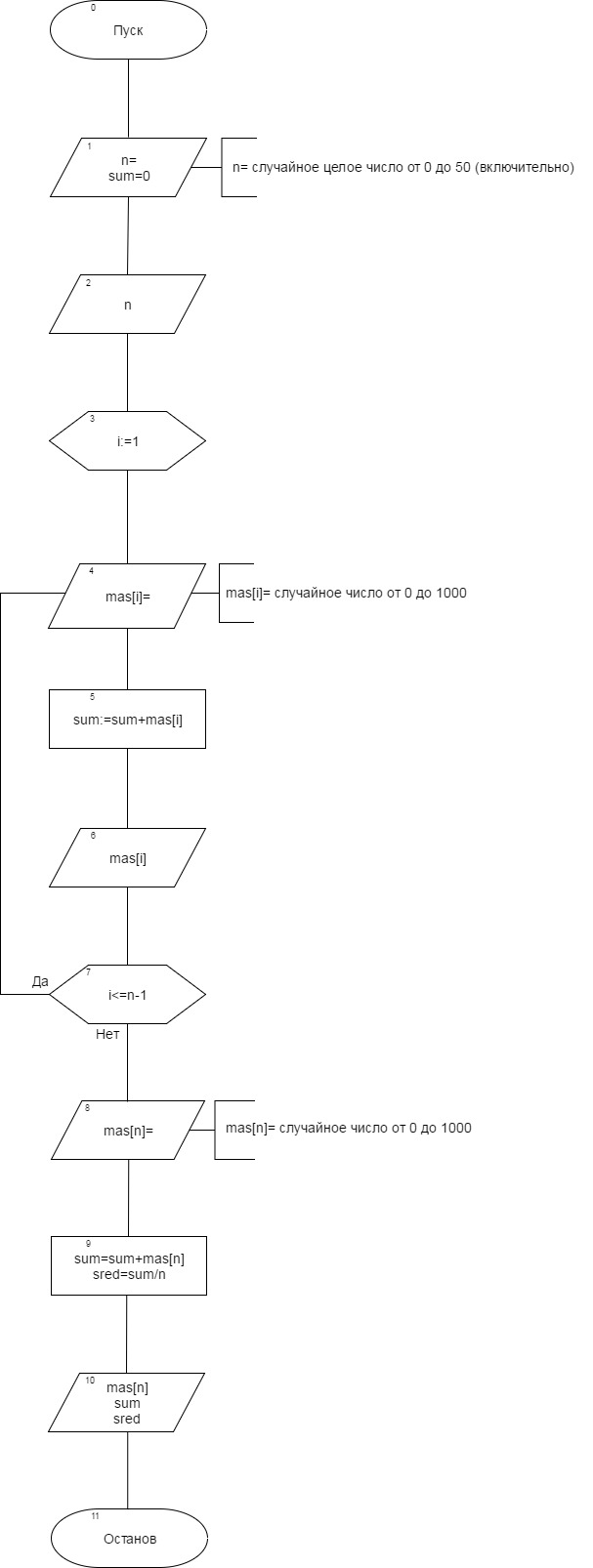
# Задача №2

## Постановка задачи: Одномерный массив задан случайным образом. Организовать вывод массива. Найти сумму его элементов, вычислить среднее арифметическое его элементов..

## Математическая модель:

1. Sred = Sum/K, где Sred – среднее арифметическое, Sum – сумма элементов массива, K – количество элементов массива.

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 2*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| *i* | Параметр цикла | integer |
| *n* | Количество элементов массива | Integer |
| *sum* | Сумма элементов массива | real |
| *sred* | Среднее арифметическое элементов массива | real |
| *mas* | Массив элементов | real |

## Код программы:

## program zadacha2;

## var mas: array of real;

## i, n: integer;

## sum, sred: real;

## begin

## Randomize;

## n:=random(51);

## SetLength(mas,n);

## sum:=0;

## writeln('Pust kolichestvo elementov massiva=',n);

## write('Massiv: ');

## for i:=1 to n-1 do

## begin

## mas[i]:= random + random(1000);

## sum:=sum+mas[i];

## write(mas[i]:3:3, ', ');

## end;

## mas[n]:=random + random(1000); // выношу последний элемент массива из цикла, чтобы после его вывода поставить точку и перенести курсор на следующую строку;

## sum:=sum+mas[n];

## sred:=sum/n;

## writeln(mas[n]:3:3, '. ');

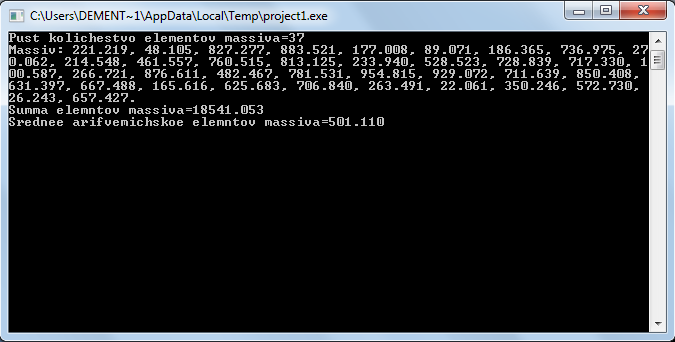
## writeln ('Summa elemntov massiva=', sum:3:3);

## writeln ('Srednee arifvemichskoe elemntov massiva=', sred:3:3);

## readln

## end.

## Результаты выполненной работы:



Анализ результатов вычисления: программа вычисляет и выводит на экран одномерный массив, состоящий из случайного количества элементов, которые заданы случайным образом, сумму и среднее арифметическое элементов этого массива. Для наглядности максимальное количество элементов массива равно 50 (включительно), а максимальное значение каждого элемента ограничено до 1000.

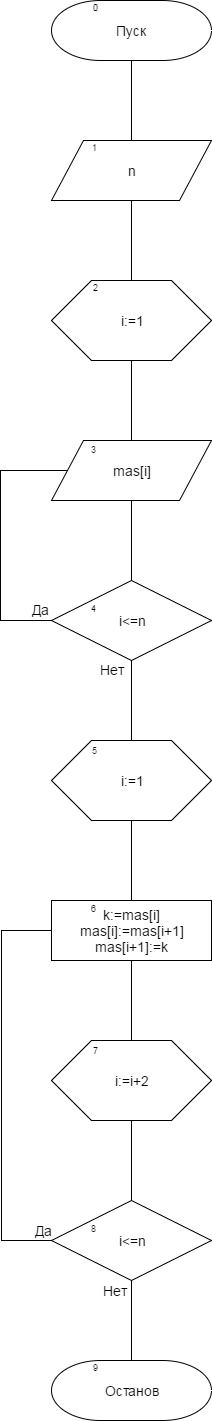
# Задача №3

## 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:

1. k:=mas[i];
2. mas[i]:=mas[i+1];
3. mas[i+1]:=k;

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 3*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| i | Параметр цикла | Integer |
| k | Промежуточная переменная | Integer |
| n | Количество элементов массива | Integer |
| *mas* | Массив | Integer |

## Код программы:

program Zadanie3;

var mas:array of integer;

n, i,k:integer;

begin

writeln('Vvedite kolichestvo elementov massiva');

readln (n);

SetLength (mas,n);

for i:=1 to n do begin

writeln ('Vvedite ', i, ' element massiva');

readln (mas[i]);

end;

i:=1;

write ('peredelanny massiv:');

while i<n do begin

k:=mas[i];

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

mas[i+1]:=k;

i:=i+2;

end;

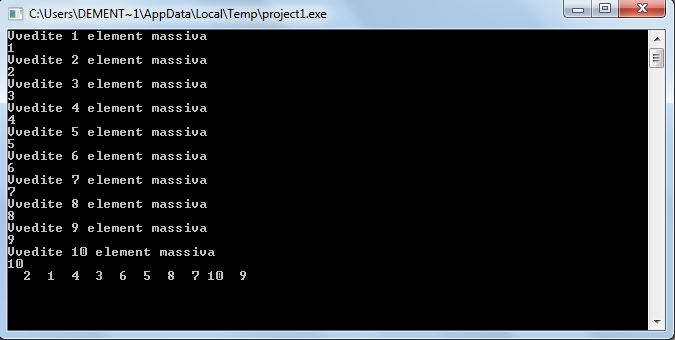
for i:=1 to n do

write(mas[i]:3);

readln

end.

## Результаты выполненной работы:



## Анализ результатов вычислений: Программа выводит на экран одномерный массив введенный пользователем с клавиатуры с переставленными элементами массива, которые стояли на четных и нечетных местах.

# Задача №4

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

## Блок-схема:

## Задача 4-1

## Список идентификаторов (обозначение переменных):

*Таблица 4*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| *i* | Параметр цикла | integer |
| *a* | Массив A | integer |
| *b* | Массив B | integer |
| *c* | Массив, состоящий из массивов A и B | integer |

## Код программы:

program Zadacha4;

const a:array[1..5] of integer = (1,2,3,4,5);

b:array[1..5] of integer = (6,7,8,9,10);

var

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

i:integer;

begin

for i:=1 to 5 do

begin

C[i]:=a[i];

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

end;

write('Massiv C: ');

for i:= 1 to 10 do

write(c[i], '; ');

readln

end.

## Результат выполненной работы:

## 2017-11-09_20-11-04

## Анализ результатов вычислений: Программа вычисляет и выводит на экран с массив, состоящий из массивов A и B, которые вводились пользователем. В начале массива С расположены элементы массива А, после которых следуют элементы массива В.

# Вывод.

Таким образом, были изучены методы реализации детерминированных ЦВП с управлением по индексу и одномерные массивы средствами Free Pascal.