**Лабораторная работа № 10**

**Тема:** Итерационные ЦВП с управлением по индексу и функции. Построение вариационного ряда.

**Цель:** Научиться реализовывать алгоритм итерационных ЦВП с управлением по индексу и функции средством при Pascal.

**Оборудование:**

* ПК
* Среда программирования Lazarus
* Интернет

**Задание 1**

**Постановка задачи**

Дан одномерный массив. Упорядочить его по убыванию.

**Математическая модель**

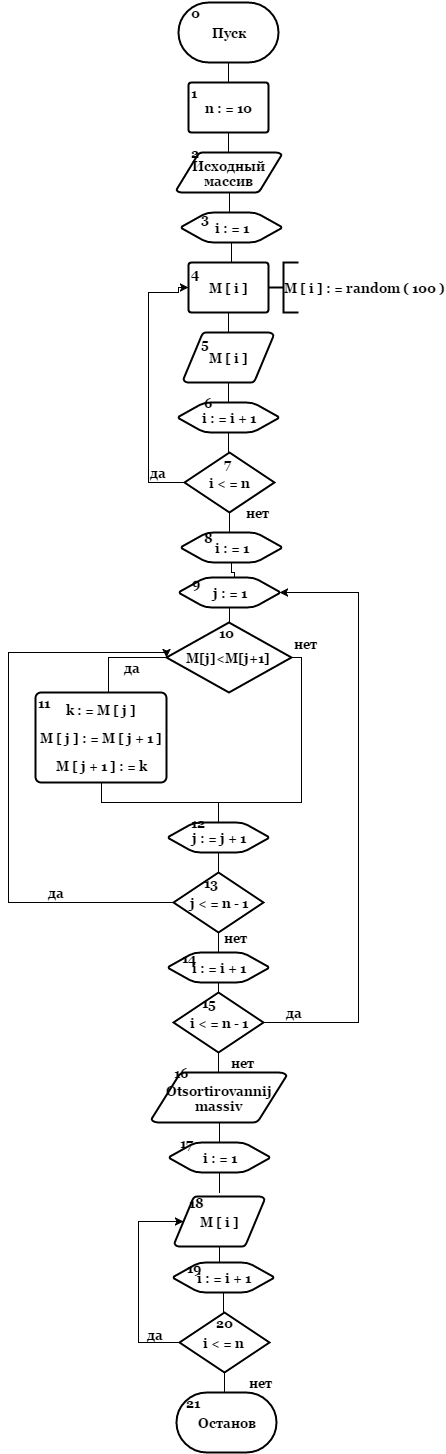
if M[j]<M[j+1] then

k:=M[j];

M[j]:=M[j+1];

M[j+1]:=k;

**Блок – схема**

****

**Список идентификаторов**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| M | array of integer | исходный массив |
| i | integer | параметр цикла, индекс элемента массива |
| n | const | 10 |
| j | integer | счётчик |
| k | integer | счетчик |

**Код программы**

program z1;

const

n=10;

var

M:array [1..n] of integer;

i,j,k:integer;

begin

randomize;

writeln('Ishodnij massiv');

for i:=1 to n do

begin

M[i]:=random(100);

writeln(M[i]);

end;

for i:=1 to n-1 do

for j:=1 to n-1 do

if M[j]<M[j+1] then

begin

k:=M[j];

M[j]:=M[j+1];

M[j+1]:=k;

end;

writeln('Otsortirovannij massiv');

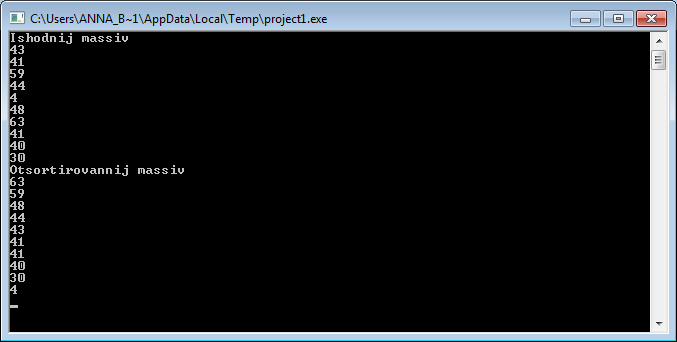
for i:=1 to n do

writeln (M[i]);

readln;

end.

**Результат выполнения программы**

****

**Анализ**

Программа выводит на экран массив в порядке убывания.

**Задание 2**

**Постановка задачи**

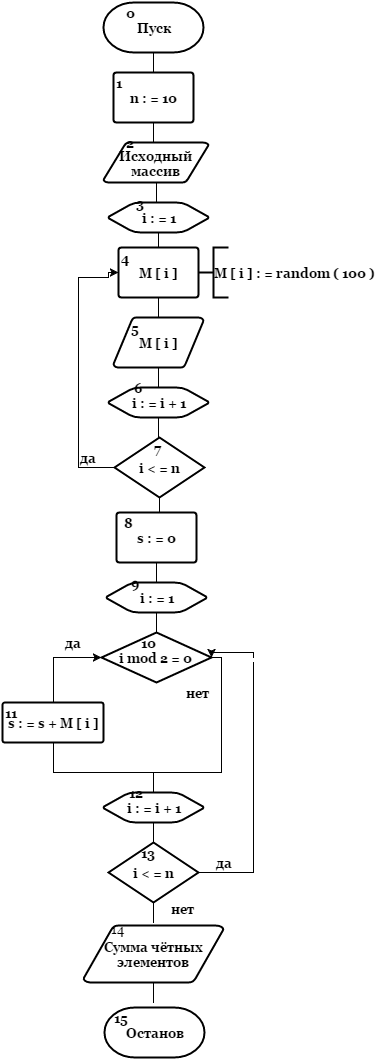
Дан одномерный массив. Найти сумму четных (по значению) элементов массива.

**Математическая модель**

if i mod 2 = 0 then

s:=s+M[i];

**Блок – схема**

****

**Список идентификаторов**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| n | const | 10 |
| M | array of integer | массив |
| i | integer | параметр цикла, индекс элемента массива |
| s | integer | сумма |

**Код программы**

program z2;

const

n=10;

var

M:array[1..n] of integer;

i,s:integer;

begin

randomize;

writeln('Ishodnij massiv');

for i:=1 to n do

begin

M[i]:=random(100);

writeln(M[i]);

end;

s:=0;

for i:=1 to n do

begin if i mod 2 = 0 then

s:=s+M[i];

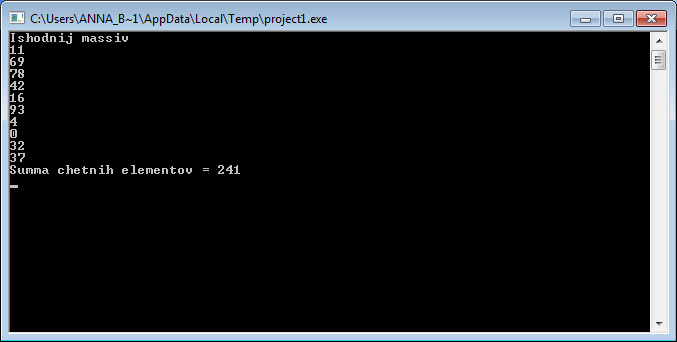
end;

writeln('Summa chetnih elementov = ',s);

readln;

end.

**Результат выполнения программы**

****

**Анализ**

Программа выводит на экран сумму чётных элементов массива.

**Задание 3**

**Постановка задачи**

Дан одномерный массив. С клавиатуры вводится число. Если такое число есть в массиве, то выводится его номер.

**Математическая модель**

if M[i]=a then

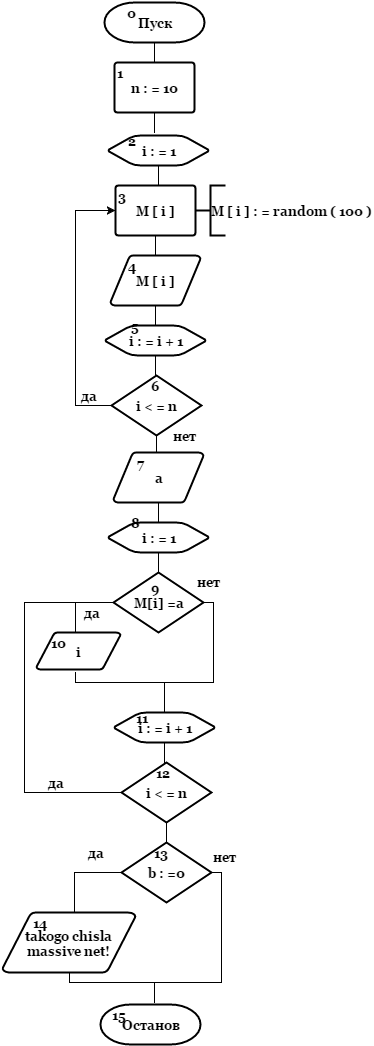
b:=1;

writeln('element pod nomerom ',i);

if b=0 then

writeln('takogo chisla massive net!');

**Блок – схема**

****

**Список идентификаторов**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| M | array of integer | массив |
| i | integer | параметр цикла, индекс элементов массива |
| b | integer | вспомогательная переменная |
| a | integer | число, вводимое с клавиатуры |

**Код программы**

program lab10\_z3;

const

n=10;

var

M:array[1..n] of integer;

i,a,b:integer;

begin

Randomize;

writeln('Massiv');

for i:=1 to n do

begin

M[i]:=random(100);

writeln(' ',M[i]);

end;

writeln('Vvedite chislo');

readln(a);

for i:=1 to n do

begin

if M[i]=a then

begin

b:=1;

writeln('element pod nomerom ',i);

end

end;

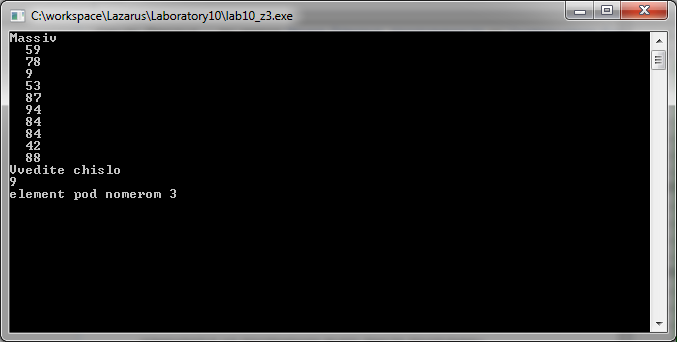
if b=0 then

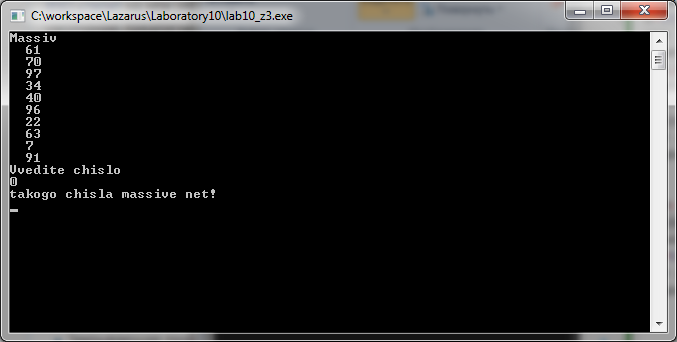
writeln('takogo chisla massive net!');

readln;

end.

**Результат выполнения программы**

****

****

**Анализ**

Программа выводит на экран номер элемента массива, если число, которое вводится с клавиатуры присутствует в массиве.

**Задание 4**

**Постановка задачи**

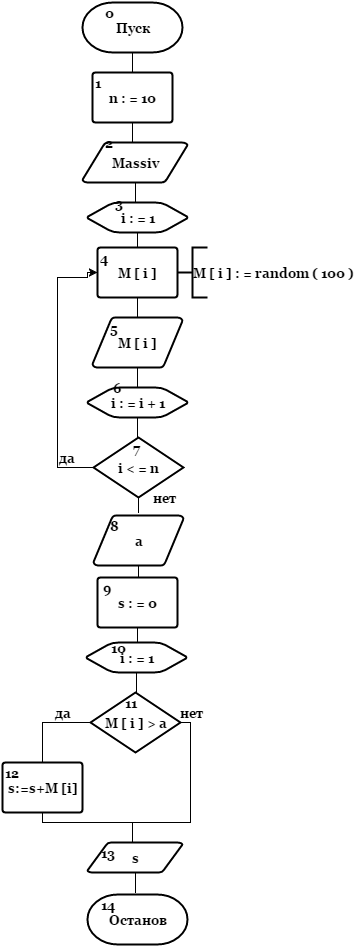
Дан одномерный массив. С клавиатуры вводится число. Найти сумму всех элементов массива, значения которых больше числа, введенного с клавиатуры.

**Математическая модель**

If M[i]>a then

s:=s+M[i]

**Блок – схема**

****

**Список идентификаторов**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| M | array of integer | массив |
| i | integer | параметр цикла, индекс элементов массива |
| a | integer | вводимое число |
| s | integer | сумма |
| n | const | 10 |

**Код программы**

program lab10\_z4;

const

n=10;

var

M:array[1..n] of integer;

i,a,s:integer;

begin

Randomize;

writeln('Massiv');

for i:=1 to n do

begin

M[i]:=random(100);

writeln(' ',M[i]);

end;

writeln('Vvedite chislo');

readln(a);

s:=0;

for i:=1 to n do

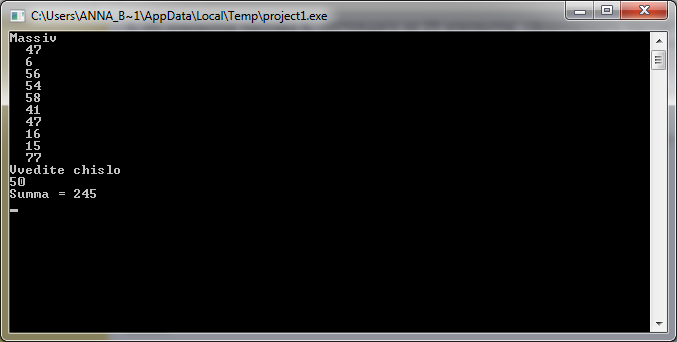
if M[i]>a then s:=s+M[i];

writeln('Summa = ', s);

readln;

end.

**Результат выполнения программы**

****

**Анализ**

Программа выводит на экран

**Задание 5**

**Постановка задачи**

Дан одномерный массив. Заменить все отрицательные элементы нулями.

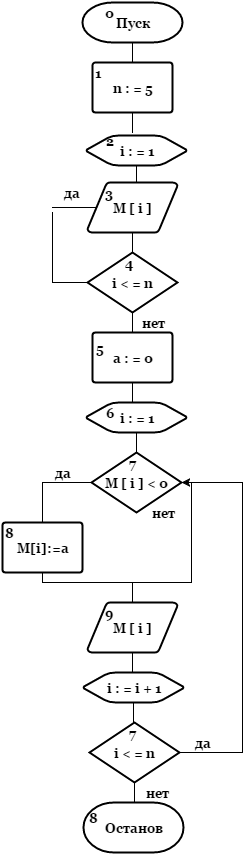
**Математическая модель**

a:=0

if M[i]<0 then

M[i]:=a

**Блок – схема**

****

**Список идентификаторов**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| M | array of integer | массив |
| i | integer | параметр цикла, индекс элементов массива |
| a | integer | вспомогательная переменная |
| n | const | 5 |

**Код программы**

program lab10\_z5;

const

n=5;

var

M:array[1..n] of integer;

i,a:integer;

begin

for i:=1 to n do

begin

readln(M[i]);

end;

a:=0;

for i:=1 to n do

begin

if M[i]<0 then

begin

M[i]:=a;

end;

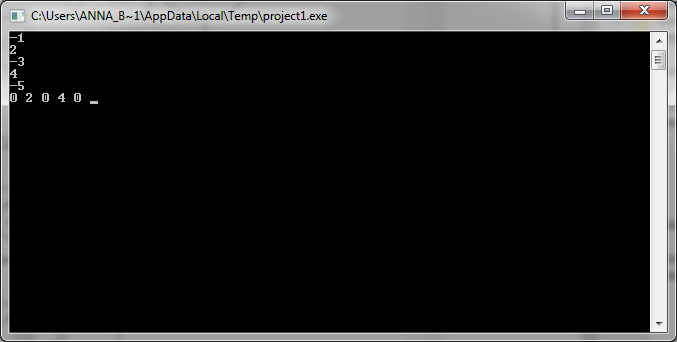
write(M[i],' ');

end;

readln;

end.

**Результат выполнения программы**

****

**Анализ**

Программа заменяет отрицательные значения массива на нули.

**Задание 6**

**Постановка задачи**

Дан одномерный массив. Найти его среднее арифметическое. Составить второй массив, элементами которого будут элементы первого массива, которые больше среднего арифметического. Остальные элементы заменить нулями. Массив отсортировать по убыванию.

**Математическая модель**

s:=s+M[i]

sr:=s/n

if M[i]>sr then

M1[i]:=M[i];

if M1[j]<M1[j+1] then

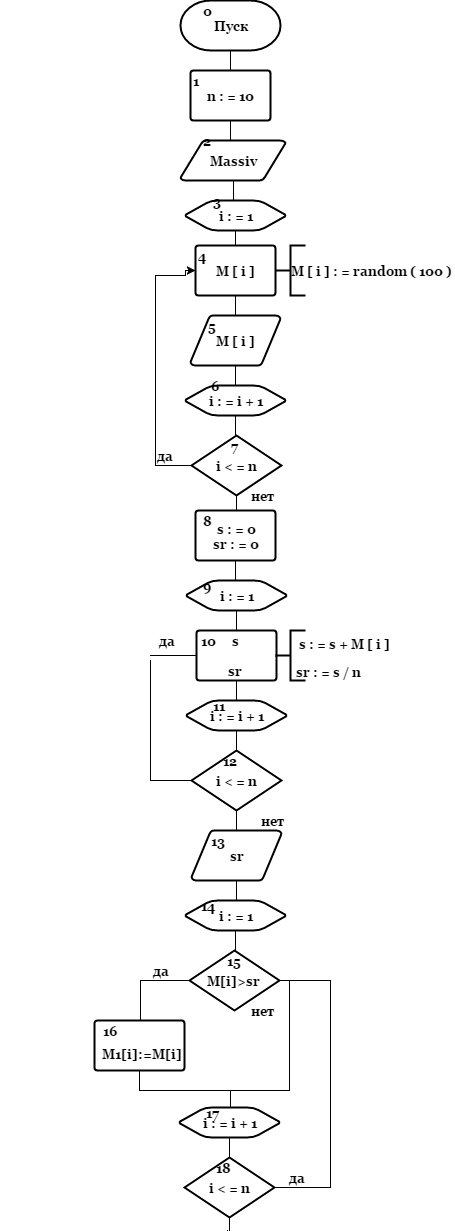
begin

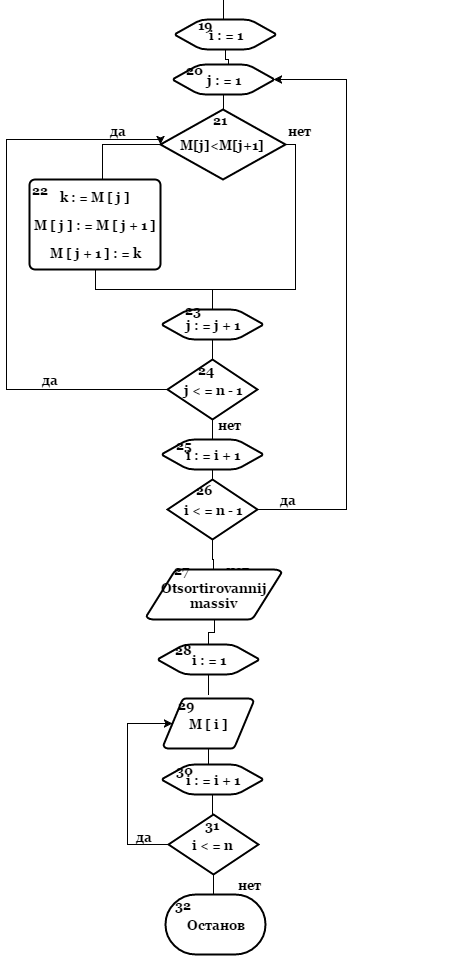
k:=M1[j];

M1[j]:=M1[j+1];

M1[j+1]:=k;

**Блок – схема**

****

****

**Список идентификаторов**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| M | array of integer | исходный массив |
| M1 | array of integer | получившийся массив |
| s | integer | сумма элементов массива |
| sr | real | среднее арифметическое массива |
| i | integer | параметр цикла, индекс элементов массива |
| k | integer | счётчик |
| j | integer | счётчик |

**Код программы**

program lab10\_z6;

const

n=10;

var

M:array [1..n] of integer;

M1:array[1..n] of integer;

i,s,k,j:integer;

sr:real;

begin

Randomize;

writeln('Massiv');

for i:=1 to n do

begin

M[i]:=random(100);

writeln(' ',M[i]);

end;

s:=0;

sr:=0;

for i:=1 to n do

begin

s:=s+M[i];

sr:=s/n;

end;

writeln('sr - ',sr:2:2);

for i:=1 to n do

begin

if M[i]>sr then

M1[i]:=M[i];

end;

for i:=1 to n-1 do

for j:=1 to n-1 do

if M1[j]<M1[j+1] then

begin

k:=M1[j];

M1[j]:=M1[j+1];

M1[j+1]:=k;

end;

writeln('Otsortirovannij massiv');

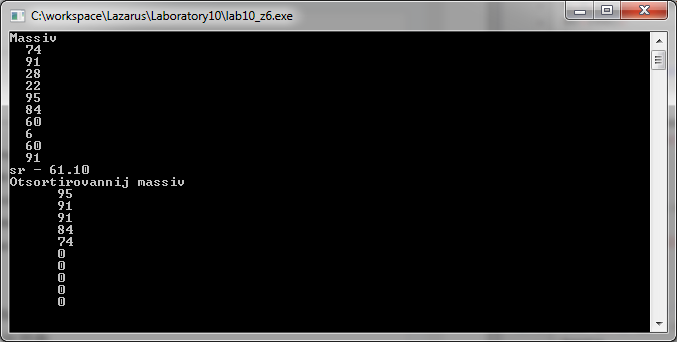
for i:=1 to n do

writeln(' ',M1[i]);

readln;

end.

**Результат выполнения программы**

****

**Анализ**

Программа выводит на экран новый массив, отсортированный в порядке убывание. Он состоит из элементов первого массива, которые больше его среднего арифметического, а элементы, которые меньше него заменены на нули.