# Лабораторная работа № 10

# Итерационные ЦВП с управлением по индексу и функции. Вариационный ряд.

Цель работы: Научиться реализовывать алгоритмы разветвляющихся вычислительных процессов и оператор выбора с помощью FreePascal.

## Оборудование: PC, Lazarus

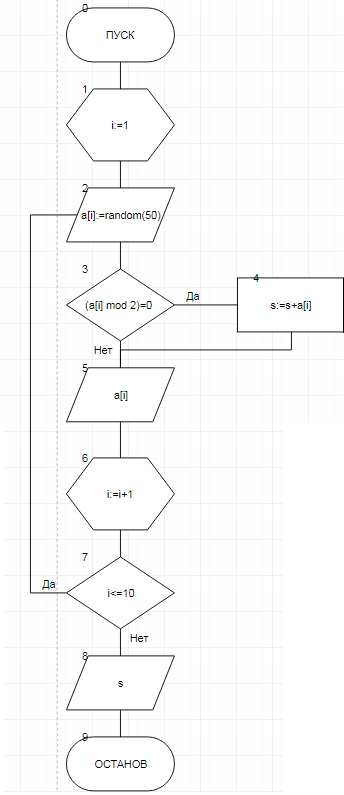
# Задача № 1

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

## Математическая модель:

Проверяем значение каждого элемента массива на четность. Если элемент четный, прибавляем его значение к переменной суммы.

## Блок-схема



## Список идентификаторов (обозначение переменных):

*Таблица 1*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Массив | array of integer |
| i | Параметр цикла | integer |
| s | Сумма четных элементов | integer |

Код программы:

program zadacha1;

var

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

i,s:integer;

begin

randomize;

for i:=1 to 10 do begin

a[i]:=random(99);

if (a[i] mod 2)=0 then

s:=s+a[i];

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

end;

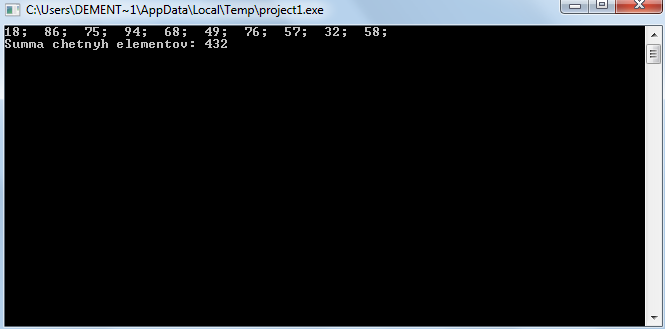
writeln();

writeln('Summa chetnyh elementov: ',s);

readln();

end.

## Результаты выполненной работы:



## Анализ результатов вычисления: Программа определяет, является ли каждый элемент массива четным, вычисляет сумму всех четных элементов массива, и выводит ответ на экран.

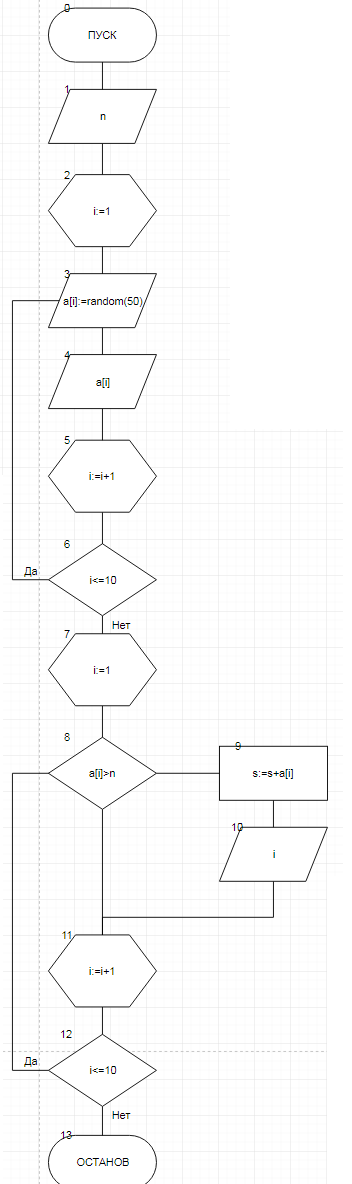
# Задача №2

Постановка задачи: Дан одномерный массив. С клавиатуры вводится число. Найти сумму всех элементов массива, значения которых больше числа, введенного с клавиатуры и вывести их индексы.

## Математическая модель:

Необходимо сравнить каждый элемент массива с введенным числом, и вычислить и вывести сумму и индексы тех элементов, которые больше введенного числа.

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 2*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Массив | array of integer |
| i | Параметр цикла | integer |
| n | Вводимое число | integer |
| s | Сумма элементов, превышающих вводимое число | integer |

## Код программы:

program zadacha2;

var a:array [1..10] of integer;

i,n,s:integer;

begin

randomize;

writeln('Vvedite chislo');

readln(n);

writeln('Massive:');

for i:=1 to 10 do begin

a[i]:=random(50);

write(a[i],' ');

end;

writeln(' ');

writeln('Indeksi:');

for i:=1 to 10 do

if a[i]>n then

begin

s:=s+a[i];

write(i,' ');

end;

writeln();

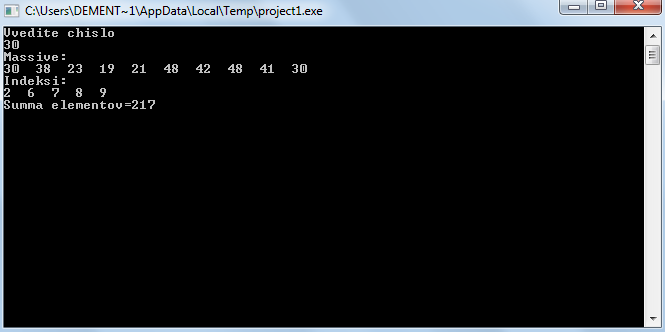
writeln('Summa elementov=',s);

readln();

end.

## 

## Результаты выполненной работы:



Анализ результатов вычисления: Программа сравнивает каждый элемент массива с введенным числом, вычисляет и выводит сумму и индексы тех элементов, которые оказались больше введенного числа.

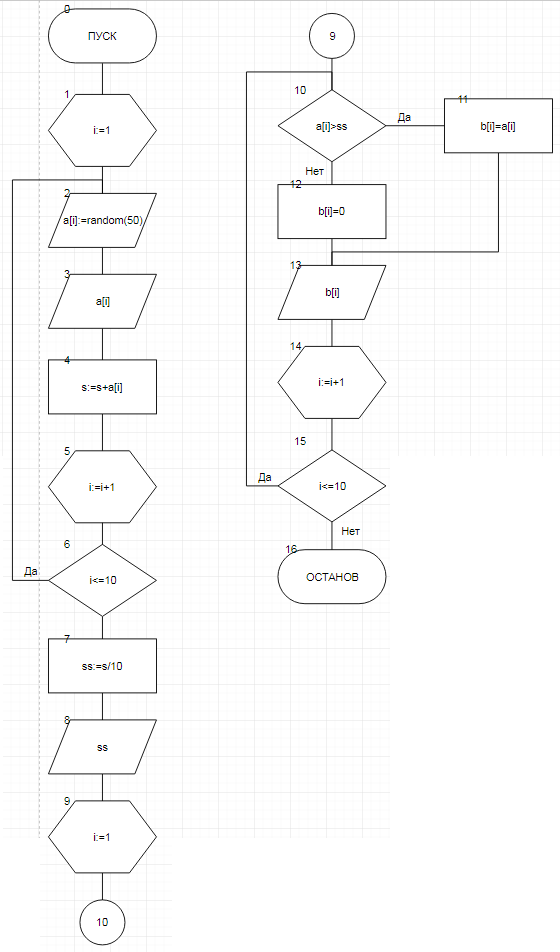
# Задача №3

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.

## Математическая модель:

**

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 3*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Массив | array of integer |
| b | Измененный массив | array of integer |
| i | Параметр цикла | integer |
| s | Сумма | integer |
| ss | Среднее арифметическое | real |

## Код программы:

program zadacha3;

var a:array [1..10] of integer;

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

i,s:integer;

ss:real;

begin

randomize;

for i:=1 to 10 do begin

a[i]:=random(50);

write(a[i],' ');

s:=s+a[i];

end;

ss:=s/10;

writeln();

writeln('Srenee arifmeticheskoe=',ss:2:0);

writeln('Izmeneniy massiv:');

for i := 1 to 10 do begin

if a[i]>ss then

b[i]:=a[i] else

b[i]:=0;

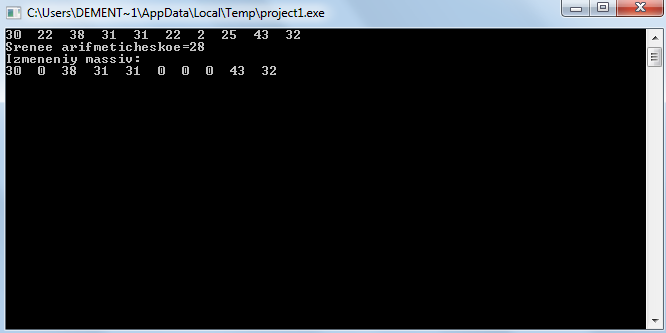
write(b[i],' ')

end;

readln();

end.

## Результаты выполненной работы:



## Анализ результатов вычисления: Программа случайным образом заполняет одномерный массив и находит его среднее арифметическое, после чего составляет второй массив, элементами которого являются элементы первого массива, которые больше среднего арифметического. Остальные элементы заменяет нулями.

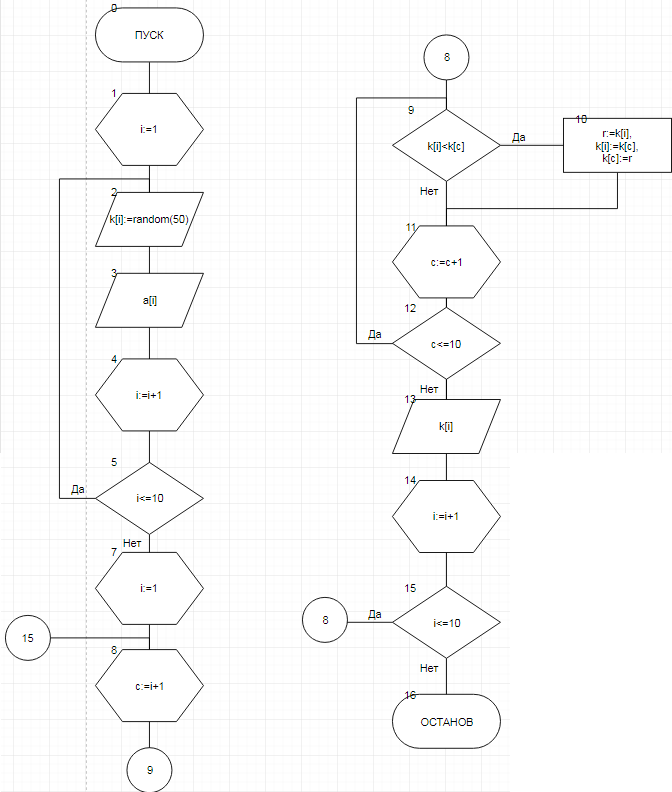
# Задача №4

Постановка задачи: Дан одномерный массив. Упорядочить его по убыванию.

## Математическая модель:

Чтобы решить поставленную задачу, нам необходимо находить максимальный по значению элемент массива и сдвигать его влево.

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 3*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| k | Массив | array of integer |
| i | Параметр цикла | integer |
| c | Параметр цикла | integer |
| r | Переменная для перестановки | integer |

## Код программы:

program zadacha4;

var r,i,c:integer;

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

begin

randomize;

for i:=1 to 10 do begin

k[i]:=random(50);

write(k[i],' ');

end;

writeln();

writeln('Sortirovka:');

for i:=1 to 10 do begin

for c:=i+1 to 10 do begin

if (k[i]<k[c]) then begin

r:=k[i];

k[i]:=k[c];

k[c]:=r;

end;

end;

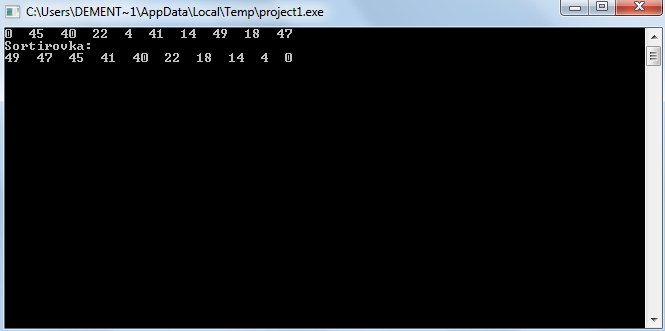
write(k[i],' ');

end;

readln;

end.

## Результаты выполненной работы:



Анализ результатов вычисления: Программа случайным образом заполняет одномерный массив, после чего меняет элементы массива местами, упорядочивая по убыванию.

# Вывод.

Таким образом, были изучены методы реализации итерационных ЦВП с управлением по индексу и функции, а также вариационный ряд средствами Free Pascal.