# Лабораторная работа № 12

# Многоступенчатые циклические вычислительные процессы. Двумерные массивы

Цель работы: Научиться реализовывать алгоритмы многоступенчатых циклических вычислительных процессов и двумерных массивов с помощью FreePascal.

## Оборудование: PC, Lazarus

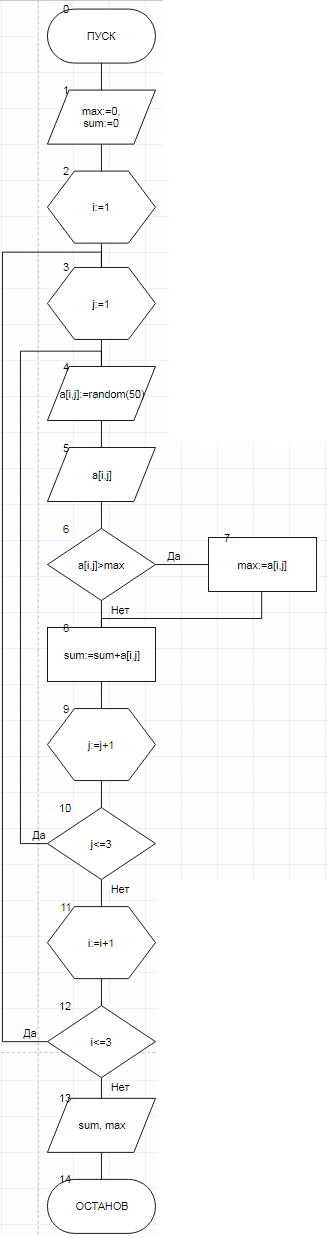
# Задача № 1

Постановка задачи: Найти сумму всех элементов массива 3x3. Массив задается явно внутри программы.

## Математическая модель:

Необходимо прибавлять к переменной суммы значение каждого элемента массива. Перебирая все элементы массива.

## Блок-схема



## Список идентификаторов (обозначение переменных):

*Таблица 1*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Массив | array of integer |
| i | Параметр цикла/индекс элементов массива | integer |
| j | Параметр цикла/ индекс элементов массива | integer |
| max | Максимальный элемент | integer |
| sum | Сумма элементов | integer |

Код программы:

program zadacha1;

var a:array [1..3,1..3] of integer;

i,j,max,sum:integer;

begin

max:=0;

sum:=0;

randomize;

for i:= 1 to 3 do begin

for j:= 1 to 3 do begin

a[i,j]:=random(50);

writeln('a[',i,',',j,']=',a[i,j]);

if a[i,j]>max then max:=a[i,j];

sum:=sum+a[i,j];

end;

end;

writeln();

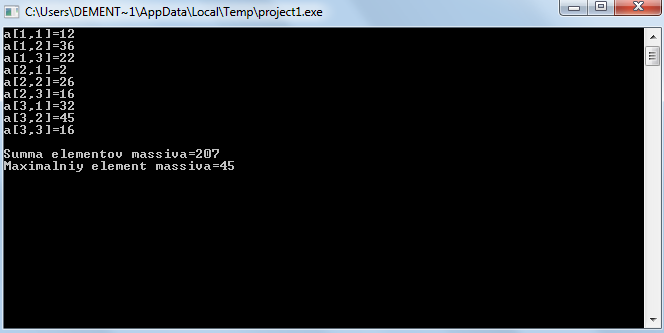
writeln('Summa elementov massiva=',sum);

writeln('Maximalniy element massiva=',max);

readln();

end.

## Результаты выполненной работы:



## Анализ результатов вычисления: Программа рассчитывает и выводит на экран максимальный элемент и сумму элементов массива.

# Задача №2

Постановка задачи: Дан массив 3x3. Найти сумму элементов на главной диагонали и сумму элементов побочной диагонали.

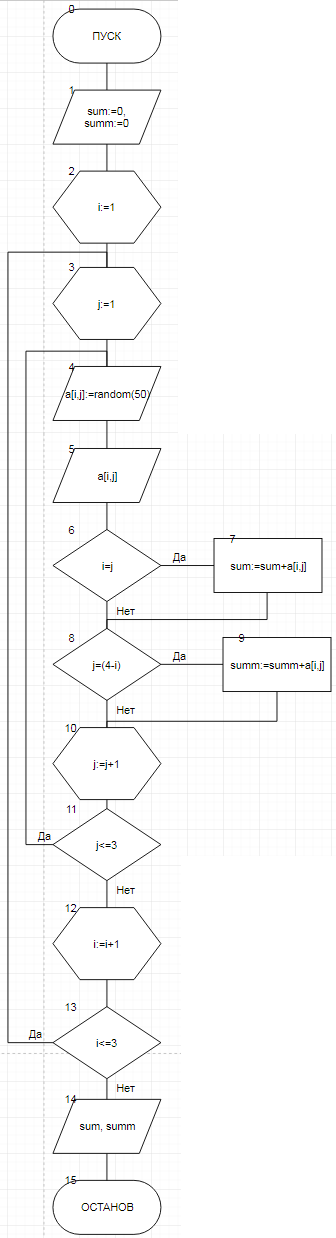
## Математическая модель:

На главной диагонали номер строки равен номеру столбца. На побочной номер столбца можно найти по формуле:

n-i+1,

где n – размерность массива, I – номер строки.

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 2*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Массив | array of integer |
| i | Параметр цикла/ индекс элементов массива | integer |
| j | Параметр цикла/ индекс элементов массива | integer |
| sum | Сумма на главной диаг. | integer |
| summ | Сумма на побочной диаг. | integer |

## Код программы:

program zadacha2;

var a:array [1..3,1..3] of integer;

i,j,sum,summ:integer;

begin

sum:=0;

summ:=0;

randomize;

for i:=1 to 3 do begin

for j:=1 to 3 do begin

a[i,j]:=random(50);

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

if i=j then sum:=sum+a[i,j];

if j=(4-i) then summ:=summ+a[i,j];

end;

writeln();

end;

writeln();

writeln('Summa elementov glavnoy diagonali=',sum);

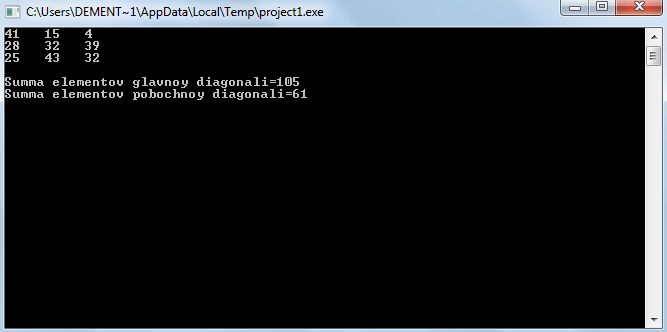
writeln('Summa elementov pobochnoy diagonali=',summ);

readln();

end.

## 

## Результаты выполненной работы:



## Анализ результатов вычисления: Программа вычисляет и выводит на экран суммы элементов главной и побочной диагоналей.

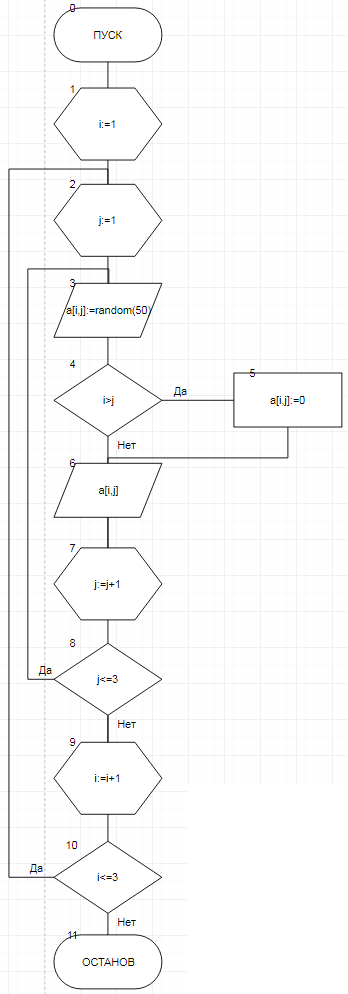
# Задача №3

Постановка задачи: Дан массив 3x3. Заменить элементы, стоящие ниже главной диагонали нулями.

## Математическая модель:

Присваиваем значение 0 всем элементам массива, чьи номера строки больше номера столбца.

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 3*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Массив | array of integer |
| i | Параметр цикла/ индекс элементов массива | integer |
| j | Параметр цикла/ индекс элементов массива | integer |

## Код программы:

program zadacha3;

var

a:array [1..3,1..3] of integer;

i,j:integer;

begin

randomize;

for i:=1 to 3 do begin

for j:=1 to 3 do begin

a[i,j]:=random(50);

if i>j then a[i,j] := 0;

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

end;

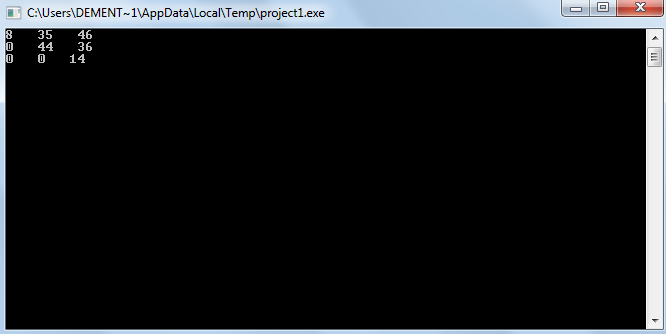
writeln();

end;

readln();

end.

## Результаты выполненной работы:



Анализ результатов вычисления: Программа случайным образом задает массив, после чего заменяет все элементы ниже главной диагонали на 0 и выводит результат на экран.

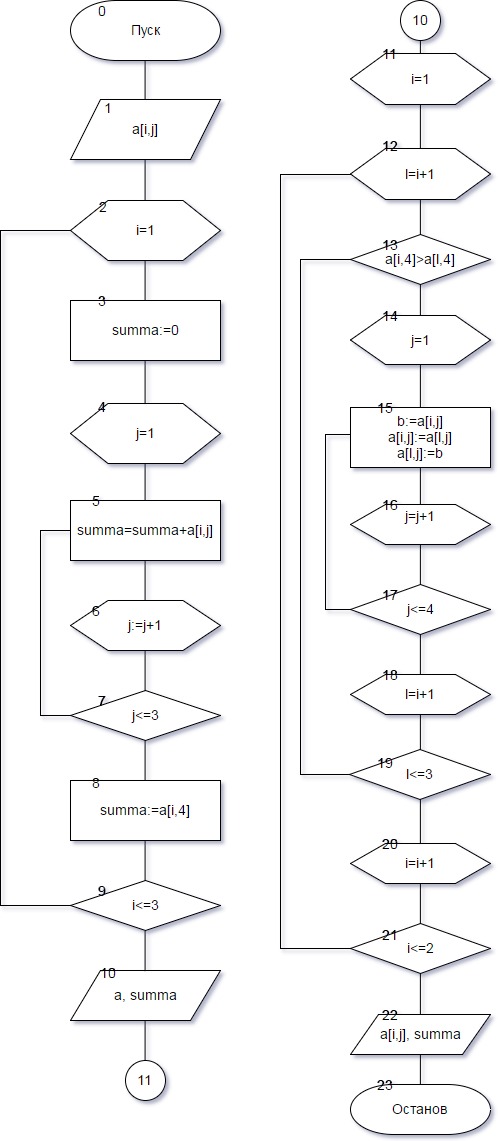
# Задача №4

Постановка задачи: Дана матрица 3x3. Найти суммы элементов каждой строки и упорядочить строки по возрастанию согласно их суммам.

## Математическая модель:

Находим сумму строк и сортируем их по возрастанию.

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 3*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Массив | array of real |
| i | Параметр цикла/ индекс элементов массива | real |
| j | Параметр цикла/ индекс элементов массива | integer |
| l | Параметр цикла/ индекс элементов массива | integer |
| summa | Сумма элементов в строке | real |
| b | Промежуточная переменная для сортировки | real |

## Код программы:

## program zadacha4;

## var a:array[1..4,1..4] of real;

## i,j,l:integer;

## summa,b:real;

## begin

## randomize;

## for i:=1 to 3 do

## begin

## summa:=0;

## for j:=1 to 3 do

## begin

## a[i,j]:=10\*random;

## summa:=summa+a[i,j];

## end;

## a[i,4]:=summa;

## end;

## writeln('Ishodny massiv:');

## writeln('Summa':(23));

## for i:=1 to 3 do

## begin

## for j:=1 to 4 do

## if j=4 then write(a[i,j]:8:1)

## else write(a[i,j]:5:1);

## writeln;

## end;

## for i:=1 to 2 do

## for l:=i+1 to 3 do

## if a[i,4]>a[l,4] then

## for j:=1 to 4 do

## begin

## b:=a[i,j];

## a[i,j]:=a[l,j];

## a[l,j]:=b;

## end;

## writeln('Stroky po vozrastaniu:');

## writeln('Summa':(23));

## for i:=1 to 3 do

## begin

## for j:=1 to 4 do

## if j=4 then write(a[i,j]:8:1)

## else write(a[i,j]:5:1);

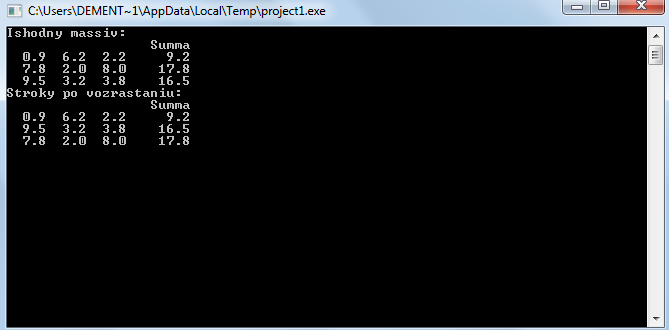
## writeln;

## end;

## readln

## end.

## Результаты выполненной работы:



Анализ результатов вычисления: Программа случайным образом задает и выводит на экран массив, а также сумму элементов в каждой строке, после чего сортирует строки в массиве по сумме элементов в каждой строке и выводит на экран “обновленный” массив.

# Вывод.

Таким образом, были изучены методы реализации многоступенчатых вычислительных процессов и двумерных массивов средствами Free Pascal.