**Лабораторная работа № 12**

**Тема:** Многоступенчатые вычислительные процессы. Двумерные массивы.

**Цель:** Научиться реализовывать алгоритм многоступенчатых вычислительных процессов средством при Pascal.

**Оборудование:**

* ПК
* Среда программирования Lazarus
* Интернет

**Задание 1**

**Постановка задачи:**

Найти сумму всех элементов массива 3x3. Массив задается явно внутри программы. Найти максимальный элемент.

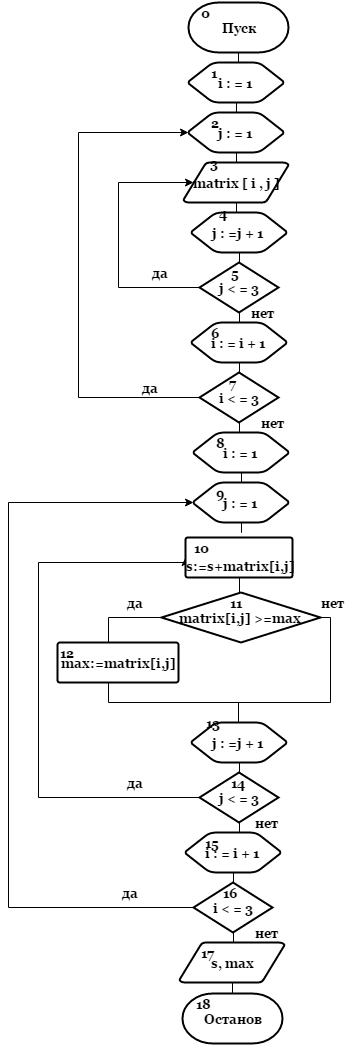
**Математическая модель:**

s:=s+matrix[i,j];

if matrix[i,j] >=max then

max:=matrix[i,j];

**Блок – схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Смысл** |
| matrix | array of integer | двумерный  массив |
| i | integer | индекс  строки |
| j | integer | индекс  столбца |
| s | integer | сумма элементов  массива |
| max | integer | Максимальный  элемент массива |

**Код программы:**

program lab12\_z1;

var

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

i,j,s,max:integer;

begin

for i:=1 to 3 do

for j:=1 to 3 do

read(matrix[i,j]);

for i:=1 to 3 do

for j:=1 to 3 do

begin

s:=s+matrix[i,j];

if matrix[i,j] >=max then

max:=matrix[i,j];

end;

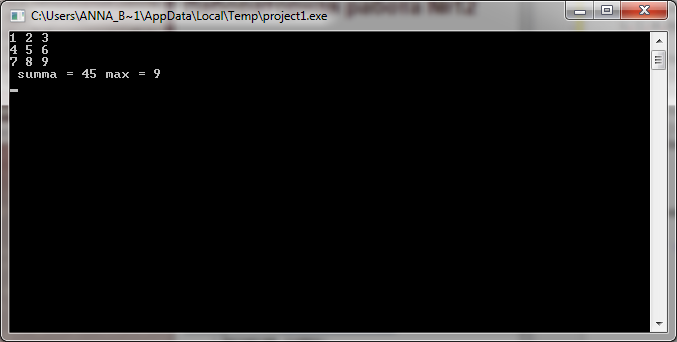
readln;

writeln(' summa = ',s,' max = ',max);

readln;

end.

**Результат выполнения программы:**

****

**Анализ:**

Программа выводит на экран сумму всех элементов и максимальный элемент массива.

**Задание 2**

**Постановка задачи:**

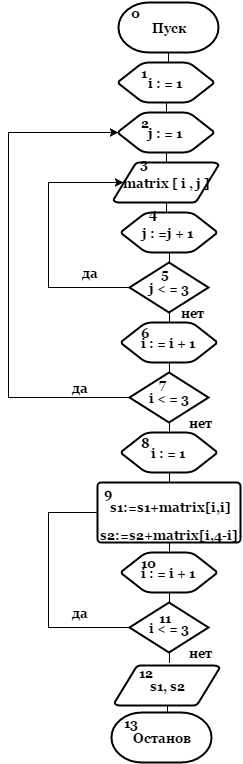
Дан массив 3x3. Найти сумму элементов на главной диагонали и сумму элементов побочной диагонали.

**Математическая модель :**

s1:=s1+matrix[i,i];

s2:=s2+matrix[i,4-i];

**Блок – схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Смысл** |
| matrix | array of integer | двумерный массив |
| i | integer | индекс строки |
| j | integer | индекс столбца |
| s1 | integer | сумма элементов  главной диагонали |
| s2 | integer | сумма элементов  побочной  диагонали |

**Код программы:**

program lab12\_z2;

var

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

i,j,s1,s2:integer;

begin

for i:=1 to 3 do

for j:=1 to 3 do

read(matrix[i,j]);

for i:=1 to 3 do

begin

s1:=s1+matrix[i,i];

s2:=s2+matrix[i,4-i];

end;

readln;

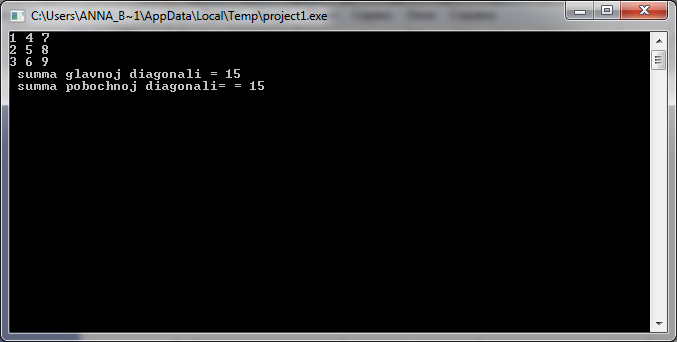
writeln(' summa glavnoj diagonali = ',s1);

writeln(' summa pobochnoj diagonali= = ',s2);

readln;

end.

**Результат выполнения программы:**

****

**Анализ:**

Программа выводит на экран результаты суммы элементов главной и побочной диагонали.

**Задание 3**

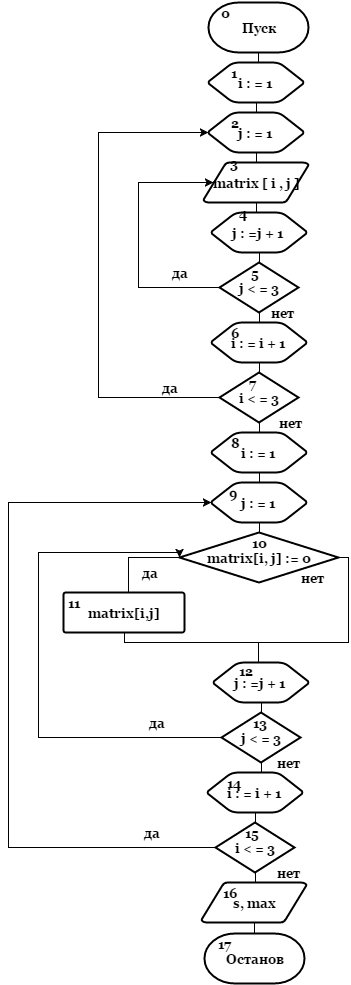
**Постановка задачи:**

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

**Математическая модель :**

if i > j then matrix[i, j] := 0

**Блок – схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Смысл** |
| matrix | array of integer | двумерный массив |
| i | integer | индекс строки |
| j | integer | индекс столбца |

**Код программы:**

program lab12\_z3;

var

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

i,j:integer;

begin

for i:=1 to 3 do

for j:=1 to 3 do

read(matrix[i,j]);

for i := 1 to 3 do

begin

writeln();

for j := 1 to 3 do

begin

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

write(matrix[i, j]:3);

end;

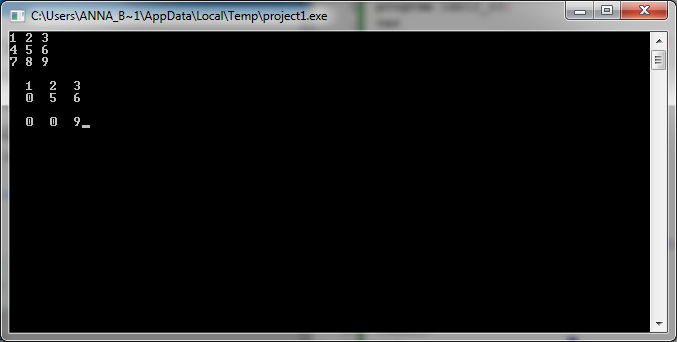
readln;

end;

readln;

end

**Результат выполнения программы:**

****

**Анализ:**

Программа выводит на экран новую матрицу, в которой элементы, находящиеся ниже главной диагонали, меняются на нули.

**Задание 4**

**Постановка задачи:**

Дана матрица 3x3. Поменять местами первую и последнюю строки.

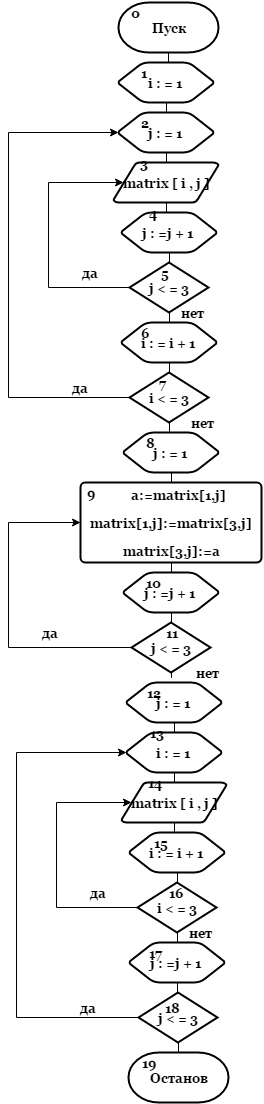
**Математическая модель :**

a:=matrix[1,j];

matrix[1,j]:=matrix[3,j];

matrix[3,j]:=a;

**Блок – схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| matrix | array of integer | двумерный массив |
| i | integer | индекс строки |
| j | integer | индекс столбца |
| a | integer | вспомогательная переменная |

**Код программы:**

program lab12\_z4;

var

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

i,j,a:integer;

begin

for i:=1 to 3 do

for j:=1 to 3 do

read(matrix[i,j]);

for j:=1 to 3 do

begin

a:=matrix[1,j];

matrix[1,j]:=matrix[3,j];

matrix[3,j]:=a;

end;

writeln('Otsortirovannaya matriza');

for j:=1 to 3 do

begin

for i:=1 to 3 do

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

writeln(' ');

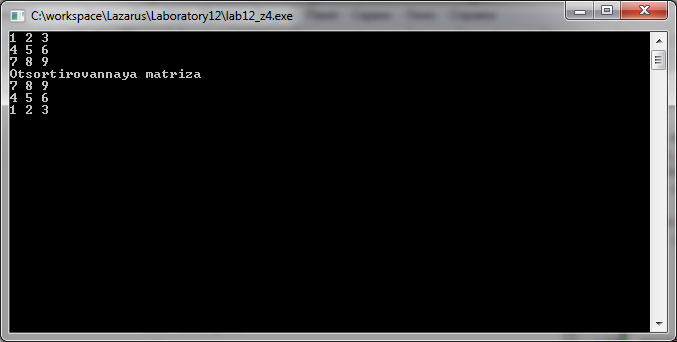
end;

readln;

readln;

end.

**Результат выполнения программы:**

****

**Анализ:**

Программа выводит на экран матрицы, в которой поменяны местами первая и последняя строчка.

**Задание 5**

**Постановка задачи:**

Дана матрица 3x3. Упорядочить строки по возрастанию.

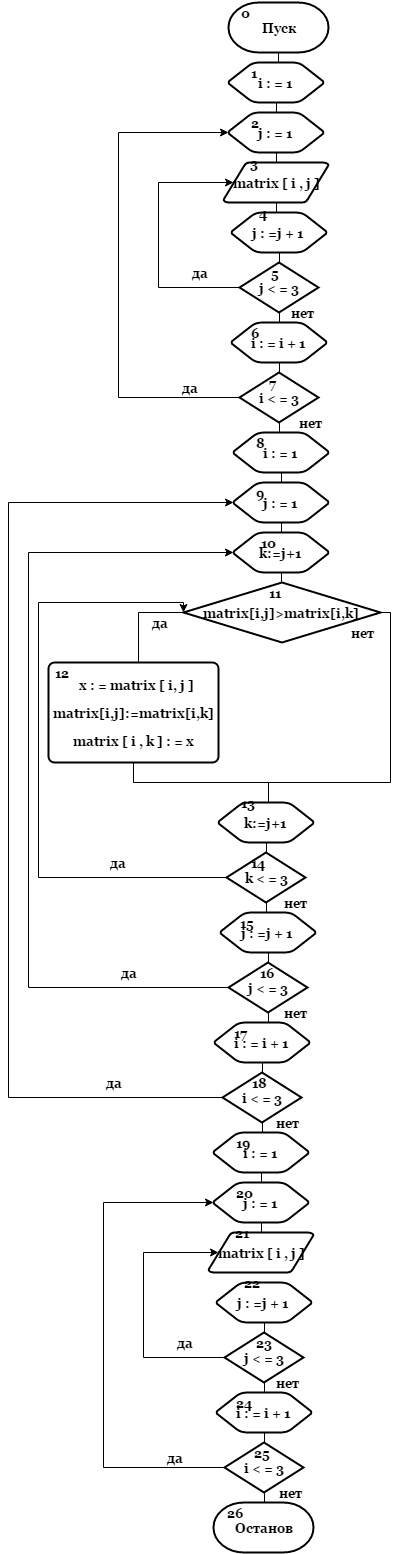
**Математическая модель:**

x:=matrix[i,j];

matrix[i,j]:=matrix[i,k];

matrix[i,k]:=x;

**Блок – схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Смысл** |
| matrix | array of integer | двумерный массив |
| i | integer | индекс строки |
| j | integer | индекс столбца |
| k | integer | вспомогательная переменная |
| x | integer | вспомогательная переменная |

**Код программы:**

program lab12\_z5;

var

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

i,j,k,x:integer;

begin

for i:=1 to 3 do

for j:=1 to 3 do

read(matrix[i,j]);

for i:=1 to 3 do

for j:=1 to 2 do

for k:=j+1 to 3 do

if matrix[i,j]>matrix[i,k] then

begin

x:=matrix[i,j];

matrix[i,j]:=matrix[i,k];

matrix[i,k]:=x;

end;

writeln('Otsortirovannij massiv');

for i:=1 to 3 do

begin

for j:=1 to 3 do

write(matrix[i,j]:3);

writeln();

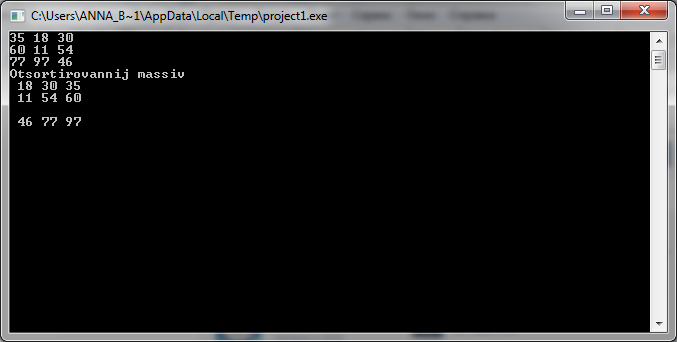
readln;

end;

readln

end

**Результат выполнения программы:**

****

**Анализ:**

Программа выводит на экран отсортированные в порядке возрастания строки.

**Задание 6**

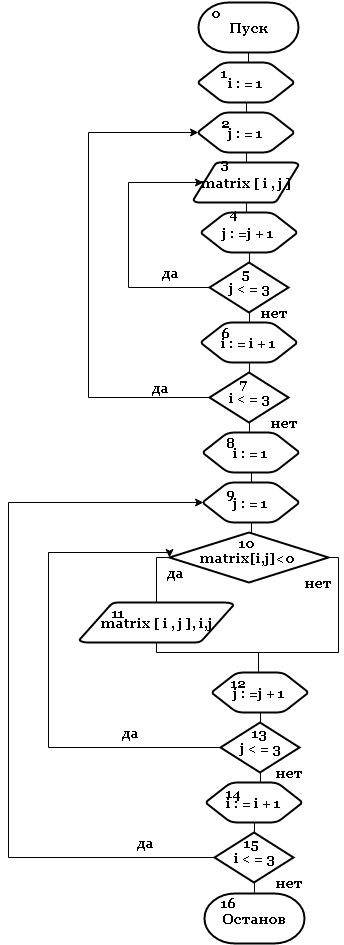
**Постановка задачи:**

Дана матрица 3x3. Выдать на экран все элементы, значение которых меньше 0, номер этого элемента в матрице 2

**Математическая модель :**

matrix[i,j]<0

**Блок – схема**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **имя** | **тип** | **смысл** |
| matrix | array of integer | двумерный массив |
| i | integer | индекс строки |
| j | integer | индекс столбца |

**Код программы:**

program lab12\_z6;

var

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

i,j:integer;

begin

for i:=1 to 3 do

for j:=1 to 3 do

read(matrix[i,j]);

writeln();

for i:=1 to 3 do

for j:=1 to 3 do

begin

if matrix[i,j]<0 then

writeln (' ',matrix[i,j]:2,' (',i,' ; ',j,' )');

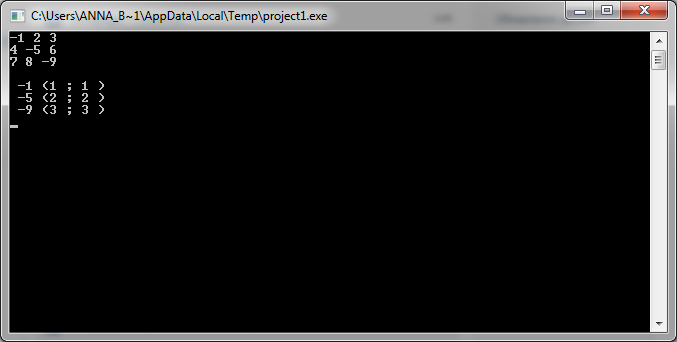
end;

readln;

readln;

end.

**Результат выполнения программы:**

****

**Анализ:**

Программа выводит на экран отрицательные элементы и их индексы.