Лабораторная работа №12. Многоступенчатые циклические вычислительные процессы. Двумерные массивы.

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

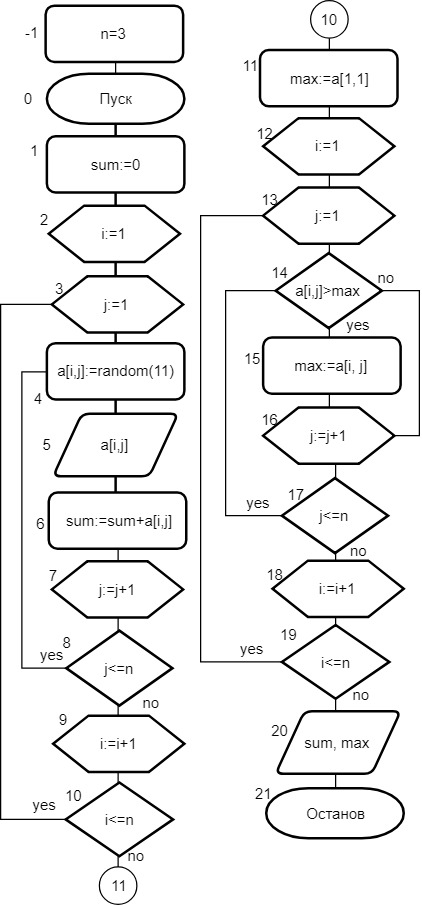
**Цель:** Изучить условия использования многоступенчатых циклических вычислительных процессов и двумерных массивов.

**Оборудование:** ПК, материалы лекций, компилятор Pascal ABC.

Задание №1.

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

**Блок-схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| ИМЯ | СМЫСЛ | ТИП |
| a | Массив/Выходные данные | integer |
| sum | Сумма | integer |
| max | Максимальный элемент | integer |
| j, i | Счетчики циклов/Индексы | integer |
| n | Граница массива | const |

**Код программы:**   
program z1;

const n=3;

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

j, i, max, sum: integer;

begin

sum:=0;

for i:=1 to n do begin

writeln;

for j:=1 to n do begin a[i,j]:=random(11);

write(a[i,j]:2, ' ');

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

end;

end;

max:=a[1,1];

for i:=1 to n do

for j:=1 to n do

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

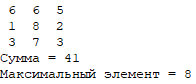
writeln;

writeln('Сумма = ', sum);

writeln('Максимальный элемент = ', max);

end.

**Результат выполнения работы:**

****

**Анализ результатов вычисления:**

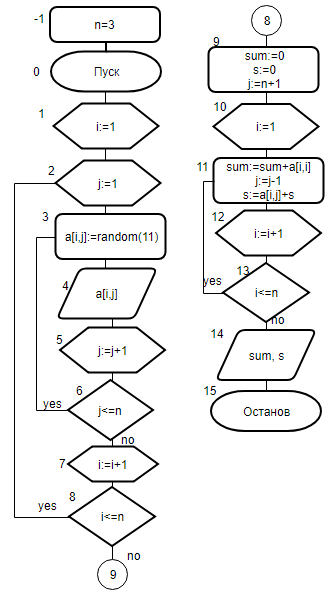
В ходе выполнения задания было изучено использование многоступенчатых циклических вычислительных процессов и двумерных массивов, и основные операции: отбор по определенному качеству, а также суммирование.

Задание №2.

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

сумму элементов побочной диагонали.

**Блок-схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| ИМЯ | СМЫСЛ | ТИП |
| a | Массив/Выходные данные | integer |
| sum, s | Суммы | integer |
| j, i | Счетчики циклов/Индексы | integer |
| n | Граница массива | const |

**Код программы:**   
 program z2;

const n=3;

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

j, i, s, sum: integer;

begin

for i:=1 to n do begin

writeln;

for j:=1 to n do begin a[i,j]:=random(11);

write(a[i,j]:2, ' ');

end;

end;

writeln;

sum:=0;

s:=0;

j:=n+1;

for i:=1 to n do begin

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

j:=j-1;

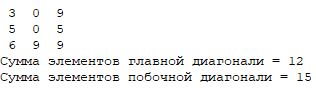
s:=a[i,j]+s;

end;

writeln('Сумма элементов главной диагонали = ', sum);

writeln('Сумма элементов побочной диагонали = ', s);

end.

**Результат выполнения работы:   
**

**Анализ результатов вычисления:**

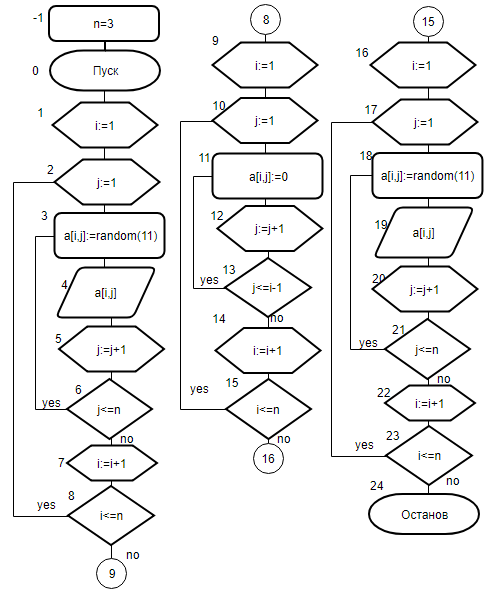
В ходе выполнения задания было изучено использование многоступенчатых циклических вычислительных процессов и двумерных массивов, а также отбор элементов для суммирования по индексу.

Задание №3.

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

диагонали нулями.

**Блок-схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| ИМЯ | СМЫСЛ | ТИП |
| a | Массив/Выходные данные | integer |
| j, i | Счетчики циклов/Индексы | integer |
| n | Граница массива | const |

**Код программы:**   
program z3;

const n=3;

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

j, i: integer;

begin

for i:=1 to n do begin

writeln;

for j:=1 to n do begin a[i,j]:=random(11);

write(a[i,j]:2, ' ');

end;

end;

for i:=1 to n do begin

for j:=1 to i-1 do begin a[i,j]:=0;

end;

end;

writeln;

for i:=1 to n do begin

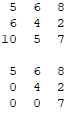
writeln;

for j:=1 to n do

write(a[i,j]:2, ' ');

end;

end.

**Результат выполнения работы:   
**

**Анализ результатов вычисления:**

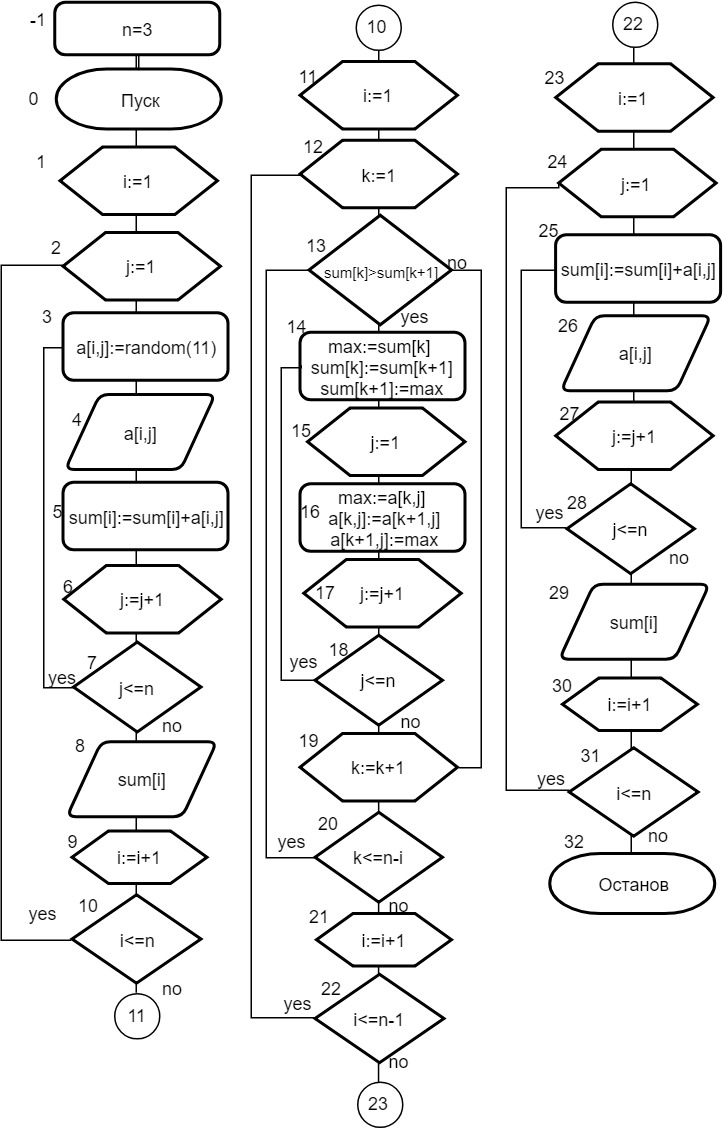
В ходе выполнения задания было изучено использование многоступенчатых циклических вычислительных процессов и двумерных массивов, а также замена элементов по определнному параметру.

Задание №4.

**Постановка задачи:** Дана матрица 3x3. Найти суммы элементов каждой строки и

упорядочить строки по возрастанию согласно их суммам

**Блок-схема:**

****

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| ИМЯ | СМЫСЛ | ТИП |
| a | Массив/Выходные данные | integer |
| sum | Массив с суммами | integer |
| k, max | Переменные для сохранения | integer |
| j, i | Счетчики циклов/Индексы | integer |
| n | Граница массива | const |

**Код программы:**   
program z4;

const n=3;

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

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

j, i, max, k: integer;

begin

for i:=1 to n do begin

writeln;

sum[i]:=0;

for j:=1 to n do begin

a[i,j]:=random(11);

write(a[i,j]:2, ' ');

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

end;

write('| ', sum[i]);

end;

writeln;

for i:=1 to n-1 do

for k:=1 to n-i do begin

if sum[k]>sum[k+1] then begin

max:=sum[k];

sum[k]:=sum[k+1];

sum[k+1]:=max;

for j:=1 to n do

begin

max:=a[k,j];

a[k,j]:=a[k+1,j];

a[k+1,j]:=max;

end;

end;

end;

for i:=1 to n do begin

writeln;

sum[i]:=0;

for j:=1 to n do begin

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

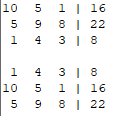
write(a[i,j]:2, ' ');

end;

write('| ', sum[i]);

end;

end.

**Результат выполнения работы:   
**

**Анализ результатов вычисления:**

В ходе выполнения задания было изучено использование многоступенчатых циклических вычислительных процессов и двумерных массивов, перестановка элементов внутри массива, а также использование вложенных циклов и ветвления

Вывод:

МЦВП и двумерные массивы удобно использовать при решении задач с матрицами, а также при работе с большим объемом информации. Но организация вычисления требует большого количества подпрограмм и циклов, что не всегда удобно и иногда вызывает трудности, особенно, когда речь заходит о сортировке элементов массива по определнному параметру.