

Обработка квадратной матрицы

Квадратная матрица - это двумерный массив, в котором количество строк равно количеству столбцов. Обращение к элементу происходит также как и в обычном двумерном массиве $A[i,j]$.

Перечислим основные свойства квадратной матрицы (рис. 1.):

1. Квадратные матрицы имеют главную и побочную диагонали. Например, для матрицы A на главной диагонали лежат элементы 1,4,9, на побочной - 45,4,12.

1	31	45
10	4	18
12	10	9

Рис. 1.

Если $i = j$ - элементы расположены на главной диагонали;

$i > j$ - элементы расположены ниже главной диагонали;

$i < j$ - элементы расположены выше главной диагонали;

$i \geq j$ - элементы расположены на главной диагонали и ниже;

$i \leq j$ - элементы расположены на главной диагонали и выше;

$i + j = n + 1$ - элементы расположены на побочной диагонали;

$i + j < n + 1$ - элементы расположены над побочной диагональю;

$i + j > n + 1$ - элементы расположены под побочной диагональю.

2. Квадратная матрица, у которой все элементы, исключая элементы главной диагонали, равны нулю, называется диагональной матрицей (рис. 2.)

1	0	0
0	2	0
0	0	3

Рис. 2.

3. Диагональная матрица, у которой все элементы, стоящие на главной диагонали равны 1, называется единичной матрицей (рис. 3.):

1	0	0
0	1	0
0	0	1

Рис. 3.

Обращение к элементу главной диагонали выглядит как $A[i,i]$, к элементу побочной как $A[i,n-i+1]$, где n – количество строк матрицы.

При составлении программ с массивами выполняются несколько этапов:

1. Объявление массива.
2. Заполнение его одним из трех способов: с клавиатуры, по правилу, случайным образом.
3. В зависимости от условия задачи выполнение расчетов или сортировки элементов матрицы.
4. Печать массива.

Все эти этапы применимы и к квадратной матрице. Рассмотрим объявление матрицы. Под объявлением понимают выделение в памяти компьютера необходимого количества ячеек под элементы матрицы. Это количество называют размерностью и определяют как произведение количества строк на количество столбцов. Применительно к квадратной матрице размерность определяется как $N*N$, где N - количество строк матрицы.

Объявление квадратной матрицы в общем виде выглядит следующим образом:

<имя массива>: array [1..N,1..N] of <тип массива>;

i,j,n:integer;

Например, объявляется целочисленная квадратная матрица из 5 строк

var A: array [1..5,1..5] of integer;

i,j: integer;

Для заполнения квадратной матрицы используется один из трех уже известных нам стандартных блоков:

Блок заполнения массива с клавиатуры

for i:=1 to N do

for j:=1 to N do begin

write ('A[',i,',',j,']=');

readln (a[i,j]);

writeln

end;

Блок заполнения массива случайным образом

randomize;

for i:=1 to N do

for j:=1 to N do A[i,j]:= random(50);

Блок заполнения массива по правилу

for i:=1 to N do

for j:=1 to N do A[i,j]:= <правило>;

Рассмотрим следующую задачу: сформировать матрицу вида (рис 4.):

program zadacha;

var a: array [1..5, 1..5] of integer;

i,j,n: integer;

begin

n=5;

2	2	2	2	2
2	2	2	2	2
2	2	2	2	2
2	2	2	2	2
2	2	2	2	2

Рис. 4.

Блок заполнения массива по правилу

for i:=1 to n do

for j:=1 to n do A[i,j]:= 2;

Пусть элементы главной диагонали равны 5 (рис 5.):

Блок заполнения элементов главной диагонали массива

for i:=1 to n do

A[i,i]:= 5;

5	2	2	2	2
2	5	2	2	2
2	2	5	2	2
2	2	2	5	2
2	2	2	2	5

Рис. 5.

Что изменится в этой программе, если матрица будет выглядеть так (рис 6.):

Добавится блок заполнения элементов побочной диагонали.

Блок заполнения элементов побочной диагонали массива

```
for i:=1 to n do
```

```
A[i,n-i+1]:= 4;
```

5	2	2	2	4
2	5	2	4	2
2	2	4	2	2
2	4	2	5	2
4	2	2	2	5

Рис. 6.

Распечатаем полученную матрицу:

Блок печати элементов массива

Вывод элементов матрицы осуществляется последовательно, необходимо напечатать элементы каждой строки и каждого столбца. При этом хотелось бы, чтобы элементы, стоящие в одной строке, печатались рядом, т.е. в строку, а элементы столбца располагались один под другим.

```
for i:=1 to n do begin
```

```
for j:=1 to n do write ('A[',i,',',j,']= ',A[i,j]);
```

```
writeln
```

```
end;
```

```
writeln
```

```
end.
```

Рассмотрим матрицу вида (рис. 7.).

Особенностью этой матрицы является то, что ненулевыми являются элементы главной диагонали и элементы, расположенные над главной диагональю.

5	2	2	2	2
0	5	2	2	2
0	0	5	2	2
0	0	0	5	2
0	0	0	0	5

Рис. 7.

Как будет выглядеть блок заполнения матрицы?

При объявлении числовой матрицы, все ее элементы заполняются нулями, значит в нашем случае, для заполнения матрицы нам достаточно указать блоки заполнения ее главной диагонали и блок заполнения элементов над главной диагональю, т.е.:

Блок заполнения элементов главной диагонали массива

for i:=1 to n do A[i,i]:= 5;

Блок заполнения элементов над главной диагональю массива

for i:=1 to n-1 do

for j:=i+1 to n do A[i,j]:= 2;

В предложенной матрице заполнить следует элементы на и под главной диагональю (рис. 8).

5	0	0	0	0
3	5	0	0	0
3	3	5	0	0
3	3	3	5	0
3	3	3	3	5

Рис. 8.

Блок заполнения элементов главной диагонали массива

for i:=1 to n do A[i,i]:= 5;

Блок заполнения элементов под главной диагональю массива

for i:=2 to n do for j:=1 to i-1 do A[i,j]:= 3;

Заполнение матрицы вида (рис. 9).

1	1	1	1	7
1	1	1	7	0
1	1	7	0	0
1	7	0	0	0
7	0	0	0	0

Рис. 9.

Блок заполнения элементов побочной диагонали массива

for i:=1 to n do A[i,n-i+1]:= 7;

Блок заполнения элементов над побочной диагональю массива

for i:=1 to n-1 do

for j:=1 to n-i do A[i,j]:= 1;

0	0	0	0	7
0	0	0	7	1
0	0	7	1	1
0	7	1	1	1
7	1	1	1	1

Рис. 10.

Заполнение матрицы вида (рис. 10).

Блок заполнения элементов побочной диагонали массива

```
for i:=1 to n do A[i,n-i+1]:= 7;
```

Блок заполнения элементов под побочной диагональю массива

```
for i:=n downto 2 to n do
```

```
for j:= n-i+2 to n do A[i,j]:= 1;
```

Мы познакомились с новым видом массивов – квадратной матрицей, узнали, как индексируются элементы квадратной матрицы, что такое главная, побочная диагонали, как они формируются по правилу, с клавиатуры, случайным образом, а также научились распечатывать значения элементов массивов по строкам и по столбцам.