

A group of four students are sitting around a table in a library, looking at a laptop and papers. The background is filled with bookshelves. The image has a blue and purple color overlay.

# ИНФОРМАТИКА

**Преподаватель:  
доц. Меретуков Шумаф Туркубиевич**



# Структурированные типы данных

••••

**Двумерный массив -  
Матрица**



## Двумерный массив

**Двумерный массив** или **матрица** в Паскале трактуется как одномерный массив, тип элементов которого также является массивом (массив массивов).

Положение элементов в двумерных массивах Паскаля описывается двумя индексами.

Их можно представить в виде прямоугольной таблицы или матрицы.

Рассмотрим двумерный массив Паскаля размерностью 3\*3, то есть в ней будет три строки, а в каждой строке по три элемента:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

Каждый элемент имеет свой номер, как у одномерных массивов, но сейчас номер уже состоит из двух чисел – номера строки, в которой находится элемент, и номера столбца.

Таким образом, **номер элемента определяется пересечением строки и столбца.**

Например,  $a_{21}$  – это элемент, стоящий во **второй строке** и в **первом столбце.**



# Матрица

**Матрица** – это прямоугольная таблица чисел (или других элементов одного типа).

**Матрица** – это массив, в котором каждый элемент имеет два индекса (номер строки и номер столбца).

**A**

	1	2	3	4	5
1	1	4	7	3	6
2	2	-5	0	15	10
3	8	9	11	12	20

столбец 3

строка 2

ячейка **A**[3, 4]





## Описание двумерного массива Паскаля

Существует несколько способов объявления двумерного массива Паскаля.

Мы уже умеем описывать одномерные массивы, элементы которых могут иметь любой тип, а, следовательно, и сами элементы могут быть массивами.

Рассмотрим варианты описания двумерных массивов:

### 1. Объявления в разделе переменных **var**

```
const N = 3;  
      M = 4;  
var A: array[1..N,1..M] of integer;  
    B: array[-3..0,-8..M] of integer;  
    Q: array['a'..'d',False..True] of real;
```



## 2. Объявления в разделе типов **type**

```
type  
  Vector = array [1..5] of <тип_элементов>;  
  Matrix = array [1..10] of vector;  
Var  
  m: matrix;
```

Мы объявили двумерный массив **m**, состоящий из 10 строк, в каждой из которых 5 столбцов.

При этом к каждой *i*-й строке можно обращаться **m [ i ]**, а каждому *j*-му элементу внутри *i*-й строки – **m [ i , j ]**.



## Описание двумерного массива Паскаля

Определение типов для двумерных массивов можно задавать и в одной строке:

```
type
  matrix= array [1..5] of array [1..10] of < тип элементов >;
Var
  m: matrix;
```

или еще проще: :

```
type
  matrix = array [1..5, 1..10] of <тип элементов>;;
Var
  m: matrix;
```



## 3. Объявления в разделе констант **const**

**const**

```
a: array[1..10] of integer=(31,546,61,9876,-35,654,1,0,90,-429);  
mass:array = ((2,3,1,0), (1,9,1,3), (3,5,7,0));
```

**ИЛИ**

**const**

```
row=3;col=3;  
matrix_1:array[1..row,1..col] of word=((1,2,3), (4,5,6), (7,8,9));  
vector_1:array[1..row] of word=(1,2,3);
```





## Объявление:

```
const
  N = 3; M = 4;
var
  A: array[1..N,1..M] of integer;
```

## Ввод с клавиатуры:

```
for j:=1 to M do
  for i:=1 to N do begin
    write('A[' , i , ' , ' , j , ']=');
    read ( A[i,j] );
  end;
```

<i>i</i>	<i>j</i>	
A[1,1]		2
A[F,2]		<del>5</del>
A[F,3]		<del>4</del>
=		4
A[3,4]		5
=		4



## Объявление:

```
const
  N = 3; M = 4;
var
  A: array[1..N,1..M] of integer;
  i, j:integer;
```

## Заполнение случайными числами:

```
randomize;
for i:=1 to N do
  for j:=1 to M do
    A[i,j] := random(25) - 10;
```

цикл по строкам

цикл по столбцам



## Вывод на экран:

```
for i:=1 to N do begin  
  for j:=1 to M do  
    write ( A[i,j]:5 );  
  writeln;  
end;
```

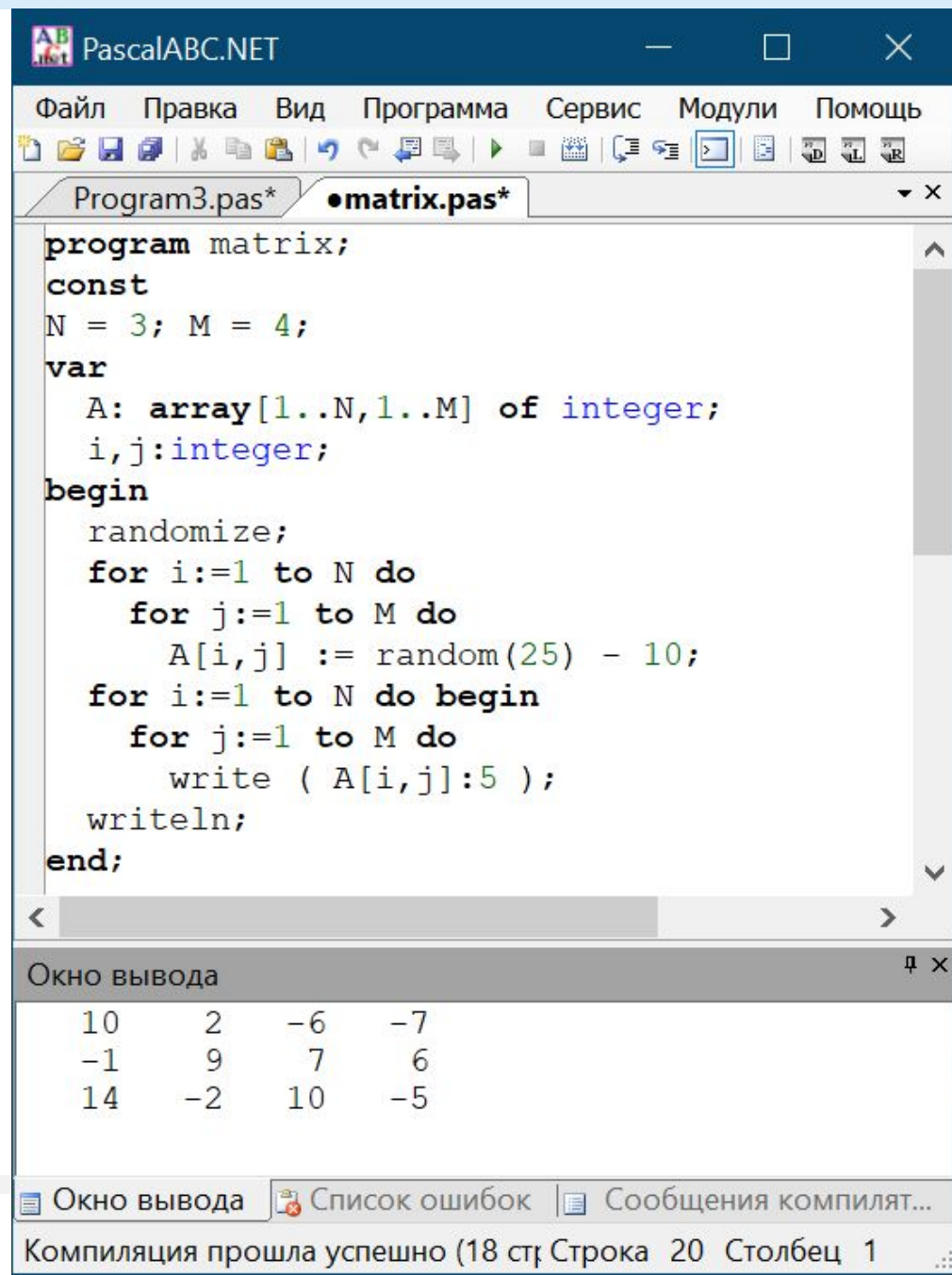
ВЫВОД СТРОКИ

12	25	1	13
156	1	12	447
1	456	222	23

в той же строке

перейти на  
новую строку

# Ввод и вывод данных матрицы



The screenshot shows the PascalABC.NET IDE with a Pascal program named 'matrix.pas'. The program generates a 3x4 matrix of random integers. The output window displays the resulting matrix:

```
program matrix;
const
N = 3; M = 4;
var
  A: array[1..N,1..M] of integer;
  i,j:integer;
begin
  randomize;
  for i:=1 to N do
    for j:=1 to M do
      A[i,j] := random(25) - 10;
  for i:=1 to N do begin
    for j:=1 to M do
      write ( A[i,j]:5 );
    writeln;
  end;
```

Row	Col 1	Col 2	Col 3	Col 4
1	10	2	-6	-7
2	-1	9	7	6
3	14	-2	10	-5

Окно вывода

Окно вывода | Список ошибок | Сообщения компилят...  
Компиляция прошла успешно (18 стр; Строка 20 Столбец 1



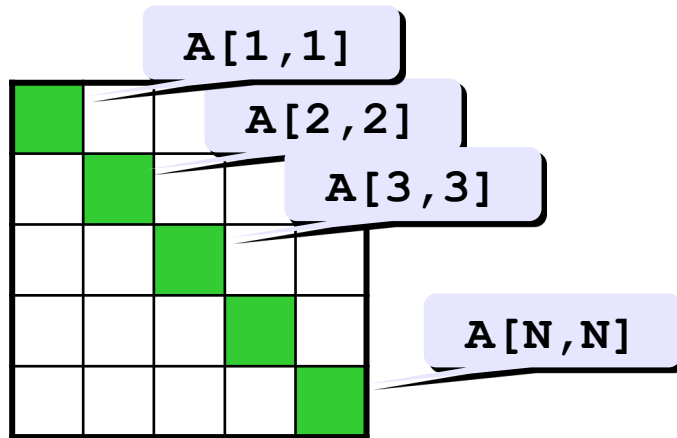
**Задача:** заполнить матрицу из 3 строк и 4 столбцов случайными числами и вывести ее на экран.  
Найти сумму элементов матрицы.

```
program qq;  
const N = 3; M = 4;  
var A: array[1..N,1..M] of integer;  
    i, j, S: integer;  
begin  
    { заполнение матрицы и вывод на экран}  
    S := 0;  
    for i:=1 to N do  
        for j:=1 to M do  
            S := S + A[i,j];  
        writeln('Сумма элементов матрицы ', S);  
    end.
```



## Обработка элементов матрицы

**Задача 2.** Вывести на экран главную диагональ квадратной матрицы из  $N$  строк и  $N$  столбцов.



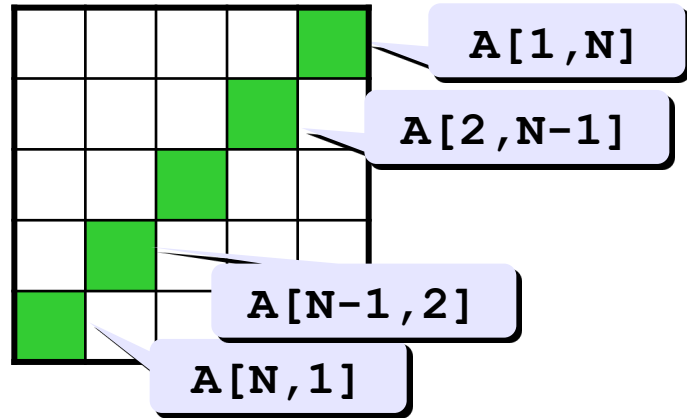
```
for i:=1 to N do  
  write ( A[i,i]:5 );
```

**Задача 2а.** Найти сумму элементов главной диагонали квадратной матрицы из  $N$  строк и  $N$  столбцов.





**Задача 3.** Вывести на экран вторую диагональ.



сумма номеров строки и столбца  $N+1$

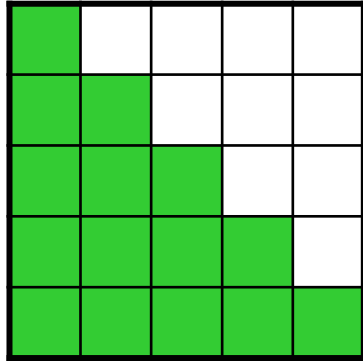
```
for i:=1 to N do  
  write ( A[i, N+1-i] : 5 );
```

**Задача 3а.** Найти сумму элементов побочной диагонали квадратной матрицы из  $N$  строк и  $N$  столбцов.



## Обработка элементов матрицы

**Задача 4.** Найти сумму элементов, стоящих на главной диагонали и ниже ее.



Одиночный цикл или вложенный?

**строка 1:**  $A[1,1]$

**строка 2:**  $A[2,1]+A[2,2]$

...

**строка N:**  $A[N,1]+A[N,2]+...+A[N,N]$

```
S := 0;  
for i:=1 to N do  
  for j:=1 to i do  
    S := S + A[i,j];
```

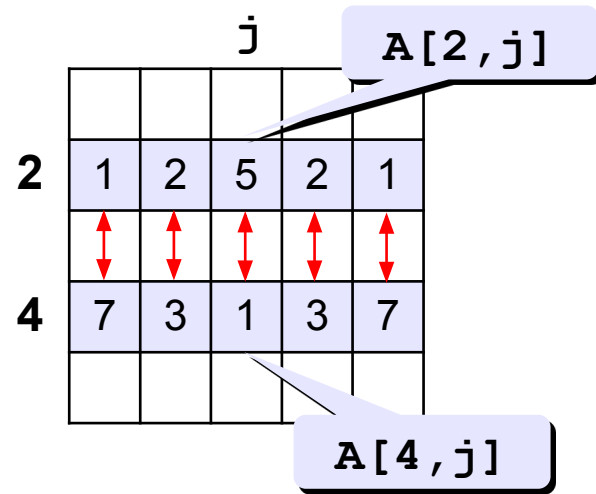
цикл по всем строкам

складываем нужные  
элементы строки  $i$



**Задача 5.** Перестановка строк или столбцов.

В матрице из  $N$  строк и  $M$  столбцов переставить 2-ую и 4-ую строки.



```
for j:=1 to M do begin
  c := A[2,j];
  A[2,j] := A[4,j];
  A[4,j] := c;
end;
```

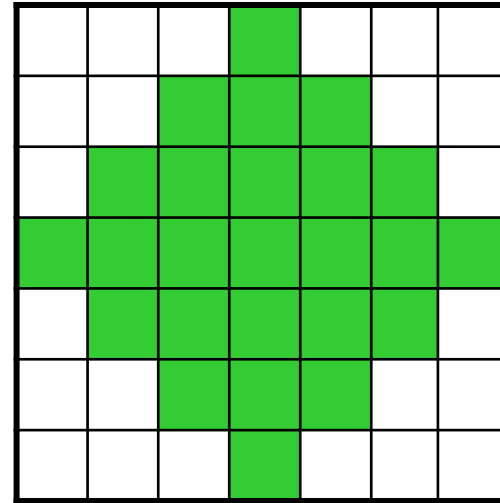
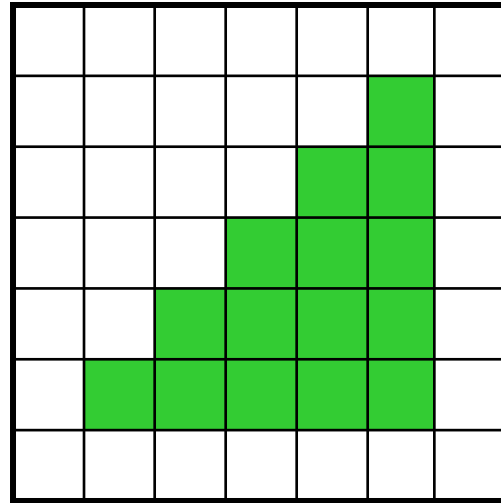
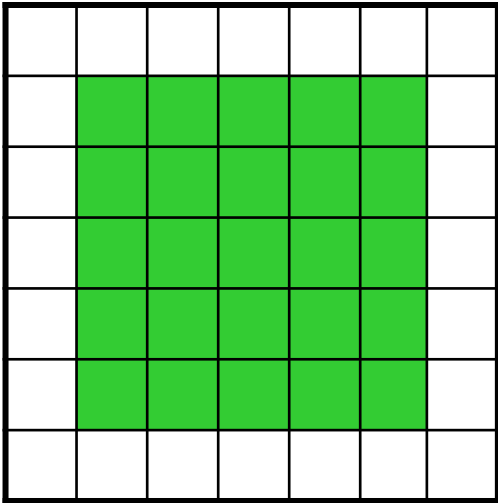
**Задача 6.** К третьему столбцу добавить шестой.

```
for i:=1 to N do
  A[i,3] := A[i,3] + A[i,6];
```



## Задачи для самостоятельного решения

1. Заполнить матрицу из 7 строк и 7 столбцов случайными числами в интервале  $[10,90]$  и вывести ее на экран.
2. Заполнить элементы, отмеченные зеленым фоном, числами 99, и вывести полученную матрицу на экран.





## Задачи для самостоятельного решения

3. Заполнить матрицу из 8 строк и 5 столбцов случайными числами в интервале  $[-10,10]$  и вывести ее на экран.

4. Удвоить все элементы матрицы и вывести её на экран.

5. Найти минимальный и максимальный элементы в матрице их номера.

**Формат вывода:**

**Минимальный элемент  $A[3,4]=-6$**

**Максимальный элемент  $A[2,2]=10$**

6. Вывести на экран строку, сумма элементов которой максимальна.

**Формат вывода:**

**Строка 2: 3 5 8 9 8**