

Таблицы и массивы

Представление таблицы в языках программирования называется **массивом**.

Примером массива:

матрица (таблица чисел, состоящая из строк и столбцов)

$$A=(1 \ 24 \ -3,5 \ 0 \ 6 \ -7)$$

Такую таблицу называют **линейной**.

Для обозначения чисел массива используют **индексированные имена**.

Пример :

$A[1]=1$ - это 1й элемент массива

$A[2]=24$ - это 2й элемент массива и т.д.

В программировании линейная таблица называется **одномерным массивом**.

В нашем примере А - это имя массива.

Элементы массива пронумерованы.

Порядковый номер элемента называется **индексом**.

Каждый элемент массива обозначается индексированным именем:

<имя массива> [<индекс>]

Пример :

A[2]

A - имя массива

2 - индекс

Все элементы массива должны иметь одинаковый тип.

Если массив состоит только из целых чисел, то тип массива – целый.

В нашем примере тип массива – вещественный.

Массив – это пронумерованная конечная последовательность однотипных величин.

Решение задач по обработке массива связано с перебором элементов массива (такой перебор можно построить в цикле).

В алгоритмическом языке массив называется таблицей и записывается **вещ таб** A[1:6], что означает массив вещественного типа с именем A и количеством элементов равным 6

алг ввод массива

вещ таб A[1:6]

цел n

нач

n:=1

вывод A=

пока n<=6

повторять

нц

ввод A[n]

n:=n+1

кц

кон

алг ввод массива

вещ таб A[1:6]

цел n

нач

вывод A=

для n **от** 1 **до** 6

повторять

нц

ввод A[n]

кц

кон

Описание и ввод массива на АЯ

Общая форма описания одномерного массива на Паскале имеет вид:

```
var <имя массива>: array [ <нижняя  
граница индекса .. верхняя граница  
индекса>] of <тип массива>
```

слово «array» переводится как «массив»

Пример :

```
var A:array [1..6] of real
```

Массивы в паскале

```
алг ввод массива
вещ таб A[1:6]
цел n
нач
вывод A=
для n от 1 до 6
    повторять
        ввод A[n]
    нц
кц
кон
```

```
Program massiv;
var A:array [1..6] of real;
    n: integer;
Begin
writeln( 'A=' );
for n:=1 to 6
do
    read(A[n]);
End.
```

Ввод массива на Паскале

Составить алгоритм и программу для умножения матрицы на число.

Решение :

Пусть матрица **B** состоит из *12*
вещественных чисел

Найдем матрицу **5B**

Пример 1


```
алг массива
вещ таб B[1:12]
цел n
нач
вывод B=
для n от 1 до 12
    повторять
        нц
            ввод B[n]
        кц
вывод 5B=
для n от 1 до 12
    повторять
        нц
            B[n]:=5*B[n]
            вывод B[n]
        кц
кон
```

```
Program massiv;
var B:array [1..12] of real;
    n: integer;
Begin
writeln( 'B=' );
for n:=1 to 12
    do
        read(B[n]);

writeln( '5B=' );
for n:=1 to 12
    do
        begin
            B[n]:=5*B[n];
            write(B[n]);
        end
End.
```

Составить алгоритм и программу для нахождения суммы двух матриц.

Решение:

Пусть 1 матрица - A состоит из 7 элементов

2 матрица - B состоит из 7 элементов

Искомая матрица $C=A+B$

Пример 2

```
алг сумма массивов
вещ таб A[1:7]
      таб B[1:7]
      таб C[1:7]
цел n
нач
  вывод A=
  для n от 1 до 7
    повторять
      нц
        ввод A[n]
      кц
  вывод B=
  для n от 1 до 7
    повторять
      нц
        ввод B[n]
      кц
  вывод C=
  для n от 1 до 7
    повторять
      нц
        C[n]:= A[n] + B[n]
      кц
  вывод C[n]
кц
```

кон

```
Program symma massivov;
var A:array [1..7] of real;
    B:array [1..7] of real;
    C:array [1..7] of real;
    n: integer;
Begin
  {ввод массива A}
  writeln( 'A=' );
  for n:=1 to 7
    do
      read(A[n]);
  {ввод массива B}
  writeln( 'B=' );
  for n:=1 to 7
    do
      read(B[n]);
  {вычисление суммы C=A+B}
  writeln( 'C=' );
  for n:=1 to 7
    do
      begin
        C[n]:= A[n]+B[n];
        write(C[n]);
      end
  End.
```

Составить алгоритм и программу нахождения положительных элемента массива.

Решение:

Пусть матрица V состоит из 12 вещественных чисел.

Для выбора положительных элементов добавим в цикл условие выбора элемента матрицы V большего нуля.

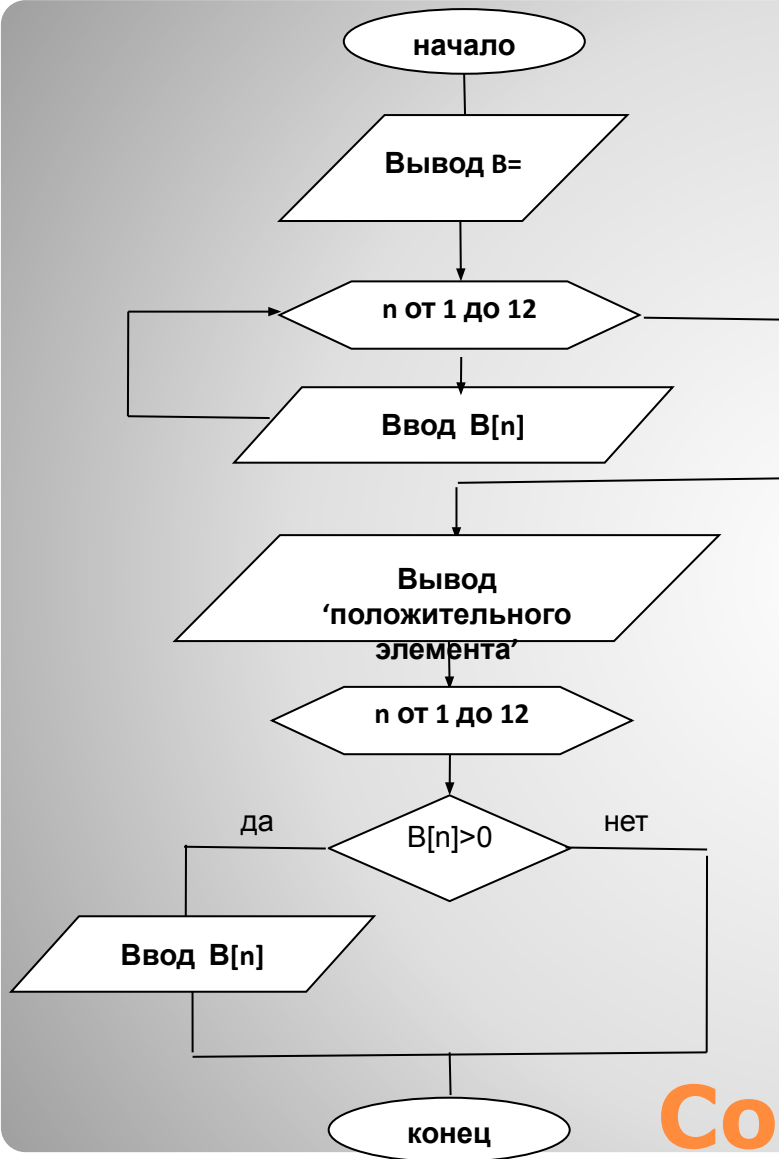
Пример 3

```
алг массива
вещ таб B[1:12]
цел n
нач
вывод B=
для n от 1 до 12
    повторять
        нц
        ввод B[n]
        кц
вывод положительные элементы
для n от 1 до 12
    повторять
        нц
        если B[n]>0
        то
        вывод B[n]
        кц
```

КОН

```
Program massiv;
var B:array [1..12] of real;
    n: integer;
Begin
writeln( 'B=' );
for n:=1 to 12
    do
        read(B[n]);
writeln( ' ПОЛОЖИТЕЛЬНЫЕ ЭЛЕМЕНТЫ ');
for n:=1 to 12
    do
        if B[n]>0
        then
            write(B[n]);

End.
```



```

Program massiv;
var B:array [1..12] of
  real;
  n: integer;
Begin
  writeln( 'B=' );
  for n:=1 to 12
  do
    read(B[n]);
  writeln( ' ПОЛОЖИТЕЛЬНИЕ  
ELEMENTI ' );
  for n:=1 to 12
  do
    if B[n]>0
    then
      write(B[n]);
  End.
  
```

Составим блок схему

Двумерные массивы

Таблица чисел, состоящая из нескольких строк и столбцов, в языках программирования называется **двумерным массивом**.

Пример:

A=	2				5	6
	4	V=	8	-5		
	-5		5	0,5	C=	6
						-5
					0,5	-4

Для обозначения чисел массива используют **индексированные имена**.

Пример :

$V[1,2] = -5$ - это элемент 1 строки 2 столбца массива
 $C[2,1] = 6$ - это элемент 2 строки 1 столбца массива
и т.д.

Описание и ввод двумерного массива на АЯ

В алгоритмическом языке двумерный массив записывается следующим образом:

вещ таб A[1:4,1:3]

Данная запись означает, что задан массив вещественного типа с именем A, количество строк равно 4, количество столбцов 3.

Пример:

	1	-3	4
A=	3	0,4	-5,3
	5	-5	6
	1	5	1,5

алг ввод двумерного массива

вещ таб A[1:4,1:3]

цел n, m

нач

вывод A=

для n **от** 1 **до** 4

для m **от** 1 **до** 3

повторять

нц

ввод A[n,m]

кц

кон

**Алгоритм ввода двумерного
массива**

Описания двумерного массива на Паскале

Общая форма описания двумерного массива на Паскале имеет вид:

```
var <имя массива>: array [<нижняя граница  
индекса строки .. верхняя граница индекса  
строки>, <нижняя граница индекса столбца ..  
верхняя граница индекса столбца>] of <тип  
массива>
```

Пример :

```
var A:array [1..6,1..5] of real
```

алг двумерный массив
вещ таб A[1:4,1:3]
цел n, m
нач

вывод A=
для n **от** 1 **до** 4
для m **от** 1 **до** 3
повторять
 нц
 ввод A[n,m]
 кц

кон

```
Program massiv;  
var A:array [1..4,1..3] of real;  
    n,m: integer;
```

```
Begin  
    writeln( 'A=' );  
    for n:=1 to 4  
    for m:=1 to 3  
    do  
        read(A[n,m]);
```

```
End.
```

**Ввод двумерного массива на
Паскале**

Составим алгоритм и программу нахождения суммы элементов двумерного массива.

Решение:

Пусть A массив вещественных чисел состоящий из 3 строк и 3 столбцов
 S - сумма чисел элементов массива

Пример 1

```
алг двумерный массив
вещ таб A[1:3,1:3], S
цел n, m
нач
    вывод A=
    для n от 1 до 3
    для m от 1 до 3
    повторять
        нц
            ввод A[n,m]
        кц
    вывод сумма элементов =
    S:=0
    для n от 1 до 3
    для m от 1 до 3
    повторять
        нц
            S:=S+A[n,m]
        кц
    вывод S
```

кон

```
Program massiv;
var A:array [1..3,1..3] of real;
    S: real;
    n,m: integer;
Begin
    writeln( 'A=' );
    for n:=1 to 3
    for m:=1 to 3
    do

        read(A[n,m]);

    writeln( 'summa elementov=' );
    S:=0 ;
    for n:=1 to 3
    for m:=1 to 3
    do

        S:=S+A[n,m];

    writeln(S);
```

End.

Найти количество положительных элементов массива.

Решение:

Пусть V массив вещественных чисел состоящий из 5 строк и 4 столбцов

k - количество элементов массива больших нуля

Пример 2

```
алг двумерный массив
вещ таб В[1:5,1:4]
цел n, m, k
нач
    вывод В=
    для n от 1 до 5
    для m от 1 до 4
    повторять
        нц
            ввод В[n,m]
        кц
    вывод k=
    k:=0
    для n от 1 до 5
    для m от 1 до 4
    повторять
        нц
            если В[n,m]>0
            то k:=k+1
        кц
    вывод k
```

кОН

```
Program massiv;
var B:array [1..5,1..4] of real;
    n,m,k: integer;
Begin
    {ввод массива В}
    writeln( 'В=');
    for n:=1 to 5
    for m:=1 to 4
    do

        read(B[n,m]);
    {количество положительных эле-
    ментов}
    writeln( 'k=');
    k:=0;
    for n:=1 to 5
    for m:=1 to 4
    do

        if B[n,m]>0
        then k:=k+1;

    writeln (k);
```

End.