

## 1. Массивы

При описании массива (таблица) указывается число его компонент, и это число остается постоянным. Для реализации прямого доступа к любой компоненте массива используются специальные обозначения. Все компоненты массива имеют общее имя – имя массива. Указание на конкретный элемент массива осуществляется с помощью индексов. Обозначение компоненты массива имеет вид:

имя массива [индекс, индекс, ..., индекс].  
 $k[1,4,2]$ .

Количество индексов в обозначении компонент массива определяет размерность массива. Массив описывается в разделе описания переменных. При этом описание массива включает описание типа массива и типа индексов. Для описания используется конструкция

имя массива: *array [1..40] of тип;*  
*k: array [1..50] of real;*

Одномерный массив  $k$ . Индексы изменяются от 1 до 50. Элементы массива имеют вещественный тип и будут обозначаться  $k[8]$ ,  $k[14]$ . Для обозначения типа индекса в этом описании использован отрезочный тип данных. Этот тип не относится к стандартным, может образовываться непосредственно программистом. Описание этого типа задается с помощью констант, разделенных двумя точками: 1..50. Первая константа указывает нижнюю границу отрезка, а вторая – его верхнюю границу. Переменная, относящаяся к этому типу (в нашем случае переменная, обозначающая индекс массива), должна принимать значение того же типа, что и границы отрезка (*integer*). Причем эти значения должны принадлежать указанному отрезку. Если же значение переменной отрезочного типа выходит за пределы указанного отрезка, то возникает ошибка. Границы отрезка в описании переменной отрезочного типа не могут быть вещественного типа. Если компоненты массива имеют более одного индекса, то в описании массива должен быть описан тип каждого индекса.

*matr:array [1..10,1..4] of real;*  
*mat [1..4];*

Тот факт, что элементы массива имеют явное обозначение, делает их равнодоступными в любой момент выполнения программы, но это налагает определенные ограничения на использование типа памяти ЭВМ. Обычно для этих целей используется основная память ЭВМ, которая называется памятью с произвольной выборкой, т. к. время выборки и записи информации в такой памяти одинаково для всех ее ячеек. Поэтому, если массив находится в основной памяти, его компоненты одинаково доступны. Однако, если массив так велик, что не вмещается в основную память, его хранят в виде отдельного файла на периферийных устройствах. Для обработки же вызывают в основную память ту часть массива, которая может там разместиться.

Пусть одномерный массив описан в разделе описания переменных так:

*var*  
*vector:array[1..50] of real;*

Тогда ввод начальных значений компонент этого массива может быть представлен:  
*for k:=1 to 50 do read(vector[k]);*

Приведенный фрагмент программы плох тем, что он годится только для ввода таких массивов, которые содержат ровно 50 компонент. Если же программа должна быть написана для работы с массивом, который при каждом новом исполнении программы может иметь другое число компонент, то целесообразно описать этот массив в разделе описаний переменных как массив с максимально возможным кол-вом компонент. Предположим, что кол-во компонент массива не может быть больше 100. Это может быть записано так:

*program wwod;*  
*uses crt;*  
*const m=100;*  
*var vector:array [1..m] of real;*  
*k:integer;*

```
BEGIN
  clrscr;
  for k:=1 to m do read (vector[k]); {ввод массива с клавиатуры}
END.
```

ИЛИ ТАК:

```
program wwod;
uses crt;
const m=100;
var vector:array [1..m] of real;
    k:integer;
BEGIN
  clrscr;
  randomize;
  for k:=1 to m do vector[k]:=random(50)-20 ; { массив задан случайным образом}
  writeln('задан массив');
  for k:=1 to m do write(vector[k]:8:2);writeln;

END.
```

**Задача.** Найти максимальный элемент одномерного массива и его номер. Число элементов массива меньше 100.

```
program maxelem;
const k=100;
var
  x:array [1..k] of real ;
  max:real;
  n,i,jmax:1..k;
BEGIN
  write ('введите размерность массива n ');
  readln (n);
  for i:=1 to n do
    begin
      write('введите x[' ,i, ' ] ');
      readln (x[i]);
    end;
  max:=x[1];
  jmax:=1;
  for i:=2 to n do
    if x[i]>max then
      begin
        max:=x[i];
        jmax:=i;
      end;
  writeln('максимальный элемент', max:6:2);
  writeln('индекс ', jmax);
END.
```

**Задача.** Определим количество элементов одномерного массива, больших среднего арифметического его положительных элементов (обязательно есть такие элементы).

```
program asd;
const n=20;
var x:array[1..n] of integer;
    k,s,i:integer; u:real;
BEGIN
  randomize;
```

```

for i:=1 to n do
  begin x[i]:=random(40)-15; write(x[i]:7); end;
writeln;
s:=0; k:=0; for i:=1 to n do
  if x[i]>0 then begin s:=s+x[i];k:=k+1 end;
u:=s/k;
writeln('cp=',u:7:2);
k:=0; for i:=1 to n do if x[i]>u then k:=k+1;
writeln('ответ ',k);

```

END.

**Задача.** Найти максимальный элемент матрицы  $m \times n$  и его индекс.

```

program maxelement;
const m=5;n=7;
var
  x:array [1..m, 1..n] of real;
  max:real;
  i,j,c,d:integer;
BEGIN
  writeln('размерность матрицы ',m,'x',n);
  for i:=1 to m do
    begin
      writeln('введите через пробел элементы строки ',i);
      for j:=1 to n do read (x[i,j]);
    end;
  max:=x[1,1];
  c:=1; d:=1;
  for i:=1 to m do
    for j:=1 to n do
      if x[i,j]> max then
        begin
          max:=x[i,j];
          c:=i; d:=j;
        end;
  writeln('max строка столбец');
  writeln (max:5:2,c:4,d:7)

```

END.

Дана символьная матрица размера  $m \times n$ . Требуется получить последовательность  $b_1, \dots, b_n$  из нулей и единиц, в которой  $b_i = 1$  тогда и только тогда, когда в  $i$ -м столбце число символов \* не меньше числа "!".

```

program aaa;
const m=3;n=4;
type matr=array[1..m,1..n] of char;
var
  s:matr;
  i,j,k,c:integer;
  b:array[1..n] of char;
BEGIN
  for i:=1 to m do
    for j:=1 to n do
      begin
        writeln ('vv s[',i,' ',j,']');
        readln (s[i,j]);
      end;
  writeln; вывод в виде матрицы на экран
  for i:=1 to m do
    begin

```

```
    for j:=1 to n do write (s[i,j]:3);  
    writeln;  
  end;  
writeln;  
for j:=1 to n do  
  begin  
    k:=0; c:=0;  
    for i:=1 to m do  
      if s[i,j]='*' then k:=k+1  
        else if s[i,j]='!' then c:=c+1;  
      if k>=c then b[j]:='1' else b[j]:='0';  
    end;  
  for j:=1 to n do write (b[j]:7);  
  writeln;  
END.
```