

Pascal 12. Динамические массивы

Тип динамического массива конструируется следующим образом:

array of *тип элементов* (одномерный массив)
array [,] of *тип элементов* (двумерный массив)
и т.д.

Переменная типа динамический массив представляет собой ссылку. Поэтому динамический массив нуждается в инициализации (выделении памяти под элементы).

Для выделения памяти под динамический массив используется два способа. Первый способ использует операцию **new** в стиле вызова конструктора класса:

```
var
  a: array of integer;
  b: array [, ] of real;
begin
  a := new integer[5];
  b := new real[4,3];
end.
```

В этом случае можно также задавать инициализацию массива значениями:

```
a := new integer[3] (1,2,3);
b := new real[4,3] ((1,2,3), (4,5,6), (7,8,9), (0,1,2));
```

Второй способ использует стандартную процедуру **SetLength**:

```
SetLength(a,10);
SetLength(b,5,3);
```

Процедура **SetLength** обладает тем преимуществом, что при ее повторном вызове старое содержимое массива сохраняется.

Если объявлен массив массивов

```
var c: array of array of integer;
```

то его инициализацию можно провести только с помощью **SetLength**:

```
SetLength(c,5);
for i := 0 to 4 do
  SetLength(c[i],3);
```

Для инициализации такого массива с помощью **new** следует ввести имя типа для **array of integer**:

```
type IntArray = array of integer;
var c: array of IntArray;
...
c := new IntArray[5];
for i := 0 to 4 do
  c[i] := new integer[3];
```

При описании инициализацию динамического массива можно проводить в сокращенной форме:

```
var
  a: array of integer := (1,2,3);
  b: array [, ] of real := ((1,2,3), (4,5,6), (7,8,9), (0,1,2));
  c: array of array of integer := ((1,2,3), (4,5), (6,7,8));
```

При этом происходит выделение памяти под указанное справа количество элементов. При таком способе в инициализаторе одномерного массива должно содержаться не менее 2 элементов.

Динамические массивы одного типа можно присваивать друг другу, при этом обе переменные-ссылки будут указывать на одну память:

```
var a1: array of integer;
var a2: array of integer;
a1 := a2;
```

Следует обратить внимание, что для динамических массивов принята [структурная эквивалентность типов](#): можно присваивать друг другу и передавать в качестве параметров подпрограмм динамические массивы, совпадающие по структуре.

Чтобы одному динамическому массиву присвоить копию другого массива, следует воспользоваться стандартной функцией **Copy**:

```
a1 := Copy(a2);
```

Длина массива (количество элементов в нем) возвращается стандартной функцией **Length** или свойством **Length**:

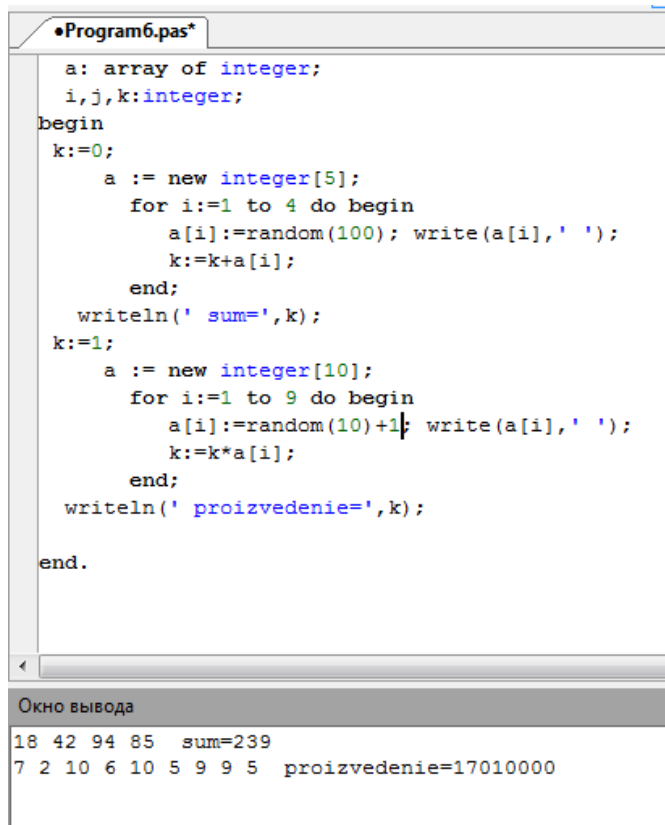
```
l := Length(a);
l := a.Length;
```

Для многомерных массивов длина по каждой размерности возвращается стандартной функцией **Length** с двумя параметрами или методом **GetLength(i)**:

```
l := Length(a,0);  
l := a.GetLength(0);
```

ПРИМЕР:

Найти сумму четырех случайных элементов массива и произведение 9 случайных элементов массива.



The image shows a screenshot of a Pascal program editor window titled "Program6.pas*". The code defines two arrays, 'a' and 'b', and calculates their sum and product. The first array 'a' has 5 elements, and the second array 'b' has 10 elements. The program uses random numbers to populate the arrays and then calculates the sum of the first 4 elements of 'a' and the product of all 10 elements of 'b'.

```
•Program6.pas*  
a: array of integer;  
i,j,k:integer;  
begin  
  k:=0;  
  a := new integer[5];  
  for i:=1 to 4 do begin  
    a[i]:=random(100); write(a[i], ' ');  
    k:=k+a[i];  
  end;  
  writeln(' sum=',k);  
  k:=1;  
  a := new integer[10];  
  for i:=1 to 9 do begin  
    a[i]:=random(10)+1; write(a[i], ' ');  
    k:=k*a[i];  
  end;  
  writeln(' proizvedenie=',k);  
end.
```

Окно вывода

```
18 42 94 85 sum=239  
7 2 10 6 10 5 9 9 5 proizvedenie=17010000
```