



# **Массивы в Паскале. Одномерные массивы.**



# Введение

Предположим, что программа работает с большим количеством однотипных данных. Скажем около ста разных целых чисел нужно обработать, выполнив над ними те или иные вычисления. Как вы себе представляете 100 переменных в программе? И для каждой переменной нужно написать одно и тоже выражение вычисления значения? Это очень неэффективно.

# Что такое массив?

**Массив** – упорядоченный набор однотипных переменных, объединенных одним именем.



Под это определение подходит множество объектов: словарь (последовательность слов), мультфильм (последовательность картинок) и т. д.

# Элементы массива

Каждая ячейка содержит элемент массива.

Элементы нумеруются по порядку,  
**но необязательно начиная с единицы** (хотя  
в языке программирования Pascal чаще всего  
именно с нее). Порядковый номер элемента  
массива называется **индексом** этого  
элемента.

Для индексов массивов подходит любой  
порядковый тип, то есть такой, который в  
памяти машины представляется целым  
числом.



# Массивы

Массивы языка Паскаль характеризуются двумя параметрами:

## Размерность

Указывает на порядок расположения массива в памяти.

Различают:  
одномерные,  
двухмерные и  
многомерные  
(размерность больше

2) массивы:

## Размер

Указывает на количество элементов по каждой координате.

Количество элементов определяется как произведение количества

## Объявление массива

Массивы, как и другие переменные, объявляются в разделе описаний **var**. В языке Паскаль тип массива задается с использованием специального слова **array**, и его объявление в программе так:

```
<имя_массива> : array [m..n] of <тип_элементов>;
```

Здесь *m* – номер первого элемента, а *n* – последнего.

**Например:**

```
Program Mass;
```

```
Var M : array [1..5] of integer;
```

**Важно понимать, что это массив из 5 целых чисел имеет имя Mass.**

# Синтаксис при объявлении массива:

1. Между именем типа и именем переменной ставится знак «двоеточие».
2. Array — служебное слово (в переводе с английского означает «массив», «набор»);
3. [m..n] — в квадратных скобках указывается номер первого элемента, затем, после двух точек, номер последнего элемента массива;
4. of — служебное слово (в переводе с английского «из»);
5. `integer` — тип элементов массива.



# Получение элементов массива

Каждый элемент массива — это переменная, которой можно присваивать значения в операторах и функциях. Для того, чтобы указать элемент массива, необходимо записать все его индексы.

**Например:**

```
Spisok[1]:='Иванов';
```

Или

```
A[1]:=3;
```

```
A[3]:=0;
```

```
A[i]:=i+1;
```

**i - номер элемента**

**A[i] - элемент массива, стоящий на i-ом месте**



# Ввод массива с клавиатуры

Массив нельзя ввести с клавиатуры одной командой, для этого организовывается цикл с параметром. Для того чтобы обеспечить правильный ввод массива, необходимо

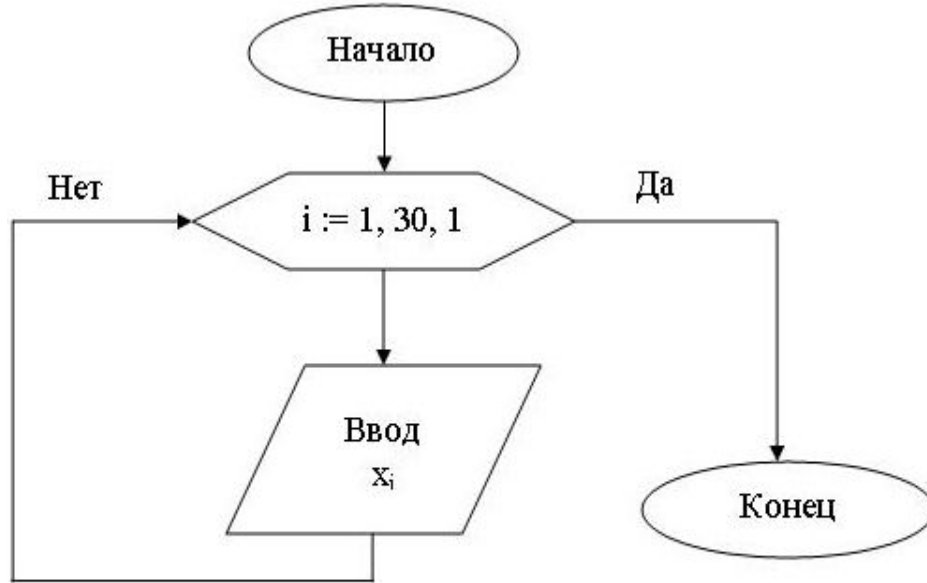
соблюдать следующий алгоритм:

```
<приглашение к вводу массива>;  
<ввод количества элементов  
массива>;  
For i:=1 to <кол-во элементов> do  
Begin  
<приглашение к вводу i-го  
элемента>;  
<ввод i-го элемента>;
```

**End**

# Пример ввода массива с клавиатуры

Ввод массивов осуществляется поэлементно с использованием цикла с параметром, где в качестве параметра используется индекс элемента массива ( $i$ ).



```
program Primer_1;  
var i: integer;  
X: array [1..30] of integer;  
begin  
  for i := 1 to 30 do read (X[i]);  
  readln;  
end.
```

Вывод одномерного массива осуществляется аналогично.

## Объявление массива

Массивы, как и другие переменные, объявляются в разделе описаний **var**. В языке Паскаль тип массива задается с использованием специального слова **array**, и его объявление в программе так:

```
<имя_массива> : array [m..n] of <тип_элементов>;
```

Здесь *m* – номер первого элемента, а *n* – последнего.

**Например:**

```
Program Mass;
```

```
Var M : array [1..5] of integer;
```

**Важно понимать, что это массив из 5 целых чисел имеет имя Mass.**

# Простая программа по работе с

В каком-то смысле с массивами можно работать, как и с обычными переменными.

```
program array_primer;  
var mas, A: array[1..10] of real;  
begin  
  mas[1]:=32;  
  mas[5]:=13;  
  mas[9]:=43;  
  A[1:]=(mas[9]-mas[1])*mas[5];  
  write(A[1]);  
end.
```

