

# Массивы.

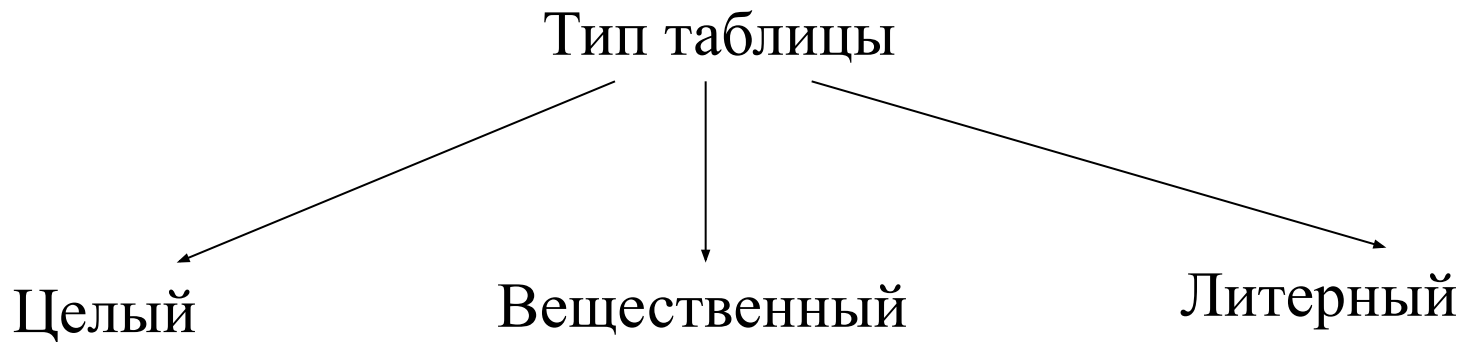
- Понятие массива.
- Заполнение массива.
- Печать массива.
- План программы



# Понятие массива.

**Массив** – это совокупность данных, объединенных общим свойством.

**Например**, список класса, температура за месяц.



# Таблица

Линейная  
(содержит  
одну строку  
или один столбец)

Например, температура  
воздуха за неделю.

10	12	8	10	9	8	8
----	----	---	----	---	---	---

*Строки нумеруются сверху вниз,  
столбцы слева направо.*

Прямоугольная  
(содержит N строк,  
M столбцов)

Например, значение  
функции  $y=x^2$

x	y
1	1
2	4
3	9
4	16



# Объявление массива.

Перед началом работы, нужно определить какие элементы будет содержать массив и сколько будет этих элементов.

**VAR**

**<имя массива>:ARRAY [<нижняя граница>..<верхняя граница>] OF <тип>;**

...

Например,

var

A: array[1..4] of integer;

T: array [1..30] of real;

1
2
3
4

T[4] := 4;

Обращение к любому элементу массива происходит по его номеру.

# Способы заполнения массива

- С помощью команды «присвоить».
- С помощью команды ввода данных с клавиатуры.
- Датчиком случайных чисел.

# Заполнение массива

## С ПОМОЩЬЮ КОМАНДЫ «ПРИСВОИТЬ».

Этот способ применяется в том случае, когда информация в таблице не меняется.

Часть программы, где организовывается заполнение массива оформляют с помощью подпрограмм.

В подпрограмме каждому элементу массива присваивается определенное значение.

```
BEGINN
```

```
TEMP[1] := 'Аксенов' ;
```

```
TEMP[2] := 'Бобров' ;
```

```
TEMP[3] := 'Веснин' ;
```

```
TEMP[4] := 'Гончаров' ;
```

```
TEMP[5] := 'Давыдов' ;
```

```
TEMP[6] := 8 ;
```

```
TEMP[7] := 8 ;
```

**SP: array [1..5] of string; integer;**

10	12	Аксенов	8	8
		Бобров		
		Веснин		
		Гончаров		
		Давыдов		



# Заполнение массива ВВОДОМ С КЛАВИАТУРЫ

Наиболее распространенный способ заполнения массива. Используется, когда для разных таблиц необходимо произвести одни операции (поиск, замена,...).

```
BEGIN
BEGIN
for i:=1 to n do
begin
write( 'Введите \',i,' температуру \') ;
write( 'Введите \',i,' элемент массива' ) ;
readln(TEMP[i]);
end;
end;
```

[Показ программы](#)



# Заполнение массива

## ДАТЧИКОМ СЛУЧАЙНЫХ ЧИСЕЛ

Этот способ заполнения используется тогда, когда важно не нахождение результата работы программы, а сам процесс (правильность алгоритма).

```
BEGIN
for i:=1 to n do
  begin
    TEMP [i] := trunc(random(10));
  end;
```

**RANDOM(N)** – вещественное число из  $[0, N)$ ;

**TRUNC(RANDOM(N))**– целое число из  $[0, N)$

**TRUNC(RANDOM(N)+1)**– целое число из  $[0, N]$

**TRUNC(RANDOM(B-A)+A)**– целое число из  $[A, B)$





# Печать линейного массива.

Линейную таблицу можно распечатать на экране в строчку или в столбик.

```
For i:=1 to n do
  begin
    write (TEMP[i] :5);
  end;
```

[Показ программы](#)

```
For i:=1 to n do
  begin
    writeln(TEMP[i]);
  end;
```

[Показ программы](#)



# План программы для работы с массивами.

```
Program Task;  
Uses Crt;  
const n=...; (количество элементов массива)  
var  
<Имя массива>: array [1..n] of <тип>;  
    i:integer;.....  
BEGIN  
    writeln('Программу составили...');  
{заполнение массива одним из способов}  
    for i:=1 to n do  
        begin  
            .....  
        end;  
{Печать массива одним из способов}  
    for i:=1 to n do  
        begin  
            .....  
        end;
```

[Команды обработки массива](#)

[Вывод результата](#)

END.