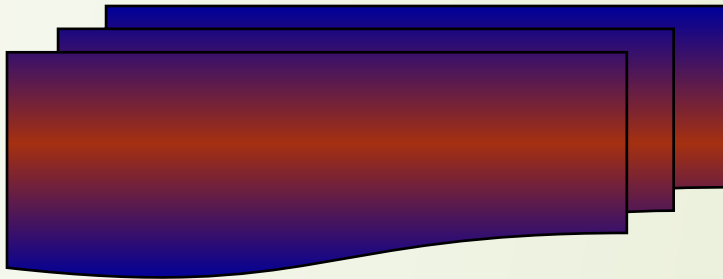




# Массивы



Массив – совокупность однотипных элементов, объединённых общим именем.

Массивы бывают

и

Одномерными

Многомерными



# Массивы

Размер массива ограничивается только объёмом рабочей памяти ПЭВМ.

В математике и информатике массив называется одномерным, если для получения доступа к его элементам достаточно одной индексной переменной.

Элементы двумерного массива имеют 2 индекса.

## Объявление одномерного массива выглядит следующим образом:

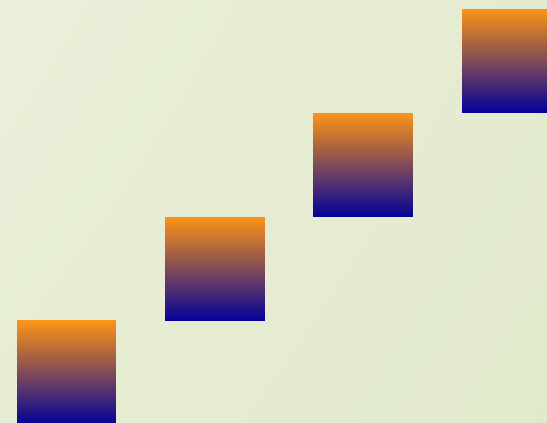
**VAR**

**имя\_массива: ARRAY [нач\_индекс . .  
кон\_индекс ] OF тип данных;**

**Например:**

**VAR mas: ARRAY[1..25] OF Integer;**

**Таким образом, мы объявили одномерный массив mas целых чисел. Номер (индекс) первого элемента 1, последний - 25.**



Объявление массивов может производиться и немного другим способом.

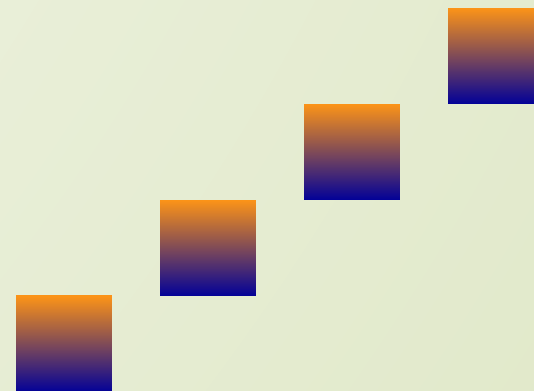
Сначала объявляется тип пользователя

(в нашем случае типа "ARRAY"), а затем и переменная на основе этого типа данных.

Например:

```
TYPE tab=ARRAY[1..25] OF Integer;
```

```
VAR mas: tab;
```



В следующем примере объявлено несколько переменных типа массив:

**VAR**

**mas\_1: ARRAY[1..10] OF Real;**

**mas\_2: ARRAY[5 .. 16, 2..5] OF Integer;**

**sst: ARRAY[-20..40, 1..4] OF Char;**



Тип индексов элементов массива может быть только простым, хотя наиболее часто в качестве индексов элементов массива применяют целые

числа.



Массивы, содержащие элементы разных типов, и индексы различных простых типов:

**VAR** mas: array [1..15] of real;

{описан массив из 15 вещественных чисел}

**www:** array [(mon,tue,wed)] of integer;

{описан массив из трёх целых чисел, индексы элементов массива имеют перечислимый тип и принимают значение названий дней недели mon, tue,wed}





Массивы, содержащие элементы разных типов, и индексы различных простых типов:

**ast: array ['A'..'Z'] of boolean;**

{описан массив элементов логического типа, тип индексов – ограниченный символьный}



**art: array [(black,white)] of 11..20;**

{описан массив целых чисел с индексами black, white. Каждый элемент массива может принимать значения от 11 до 20}



## SVZ: array [byte] of integer;

{описан массив из 256 целых чисел с индексами стандартного типа byte (от 0 до 255)}

1  
0

Какие массивы описаны ниже?

Vin: array ['a'..'d'] of 1..20:



# Объясните значение следующих понятий:

Массив

В

Array

Индекс

byte

Boolean

н

Тип

е

Тип элементов

массива

Формирован

ие

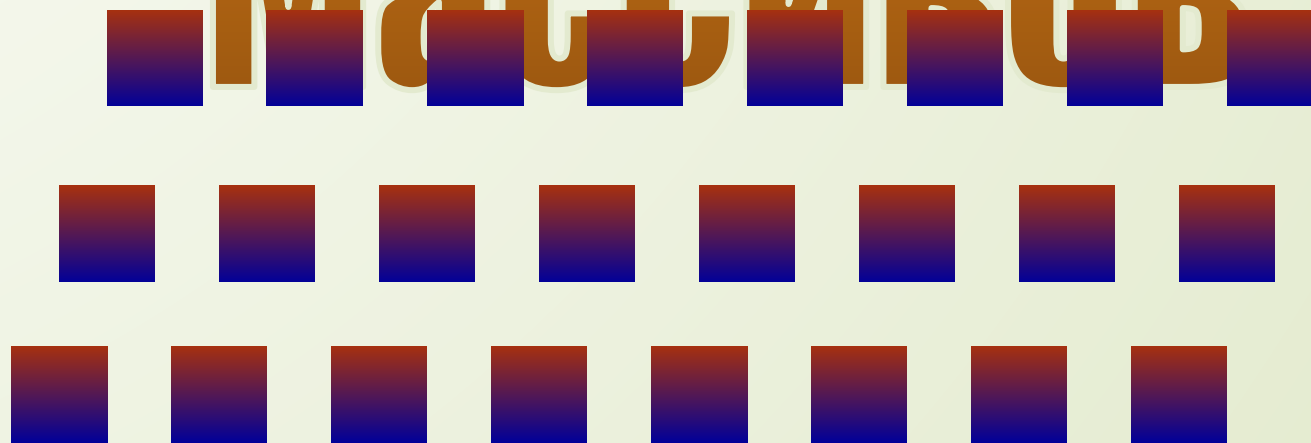
массива

Объявление

массива

# Способы формирования

## Массивов



# Способы формирования

1

3

СИВОВ



# I. Запрос значений элементов в ходе выполнения программы.

*Нижеприведённый фрагмент демонстрирует запрос  $n$  элементов массива с именем  $K$ .*

```
Writeln ('Введите',n,' элементов  
массива');
```

```
For l:=1 to n do read(K[l]);
```



## II. Формирование массива с помощью арифметического выражения.

1  
5

При таком способе формирования для наглядности желательно выводить полученный массив на экран, т.к. глядя на формулу, бывает трудно сразу определить, чему равны его элементы.

```
For l:=1 to n do  
begin K[l]:=3*sqr(l);  
write (K[l], ' ');
```



### III. Формирование массива с помощью генератора случайных чисел.

Здесь используется ключевое слово **Random(n)** – генератор случайных целых чисел в промежутке  $[0;n)$ .

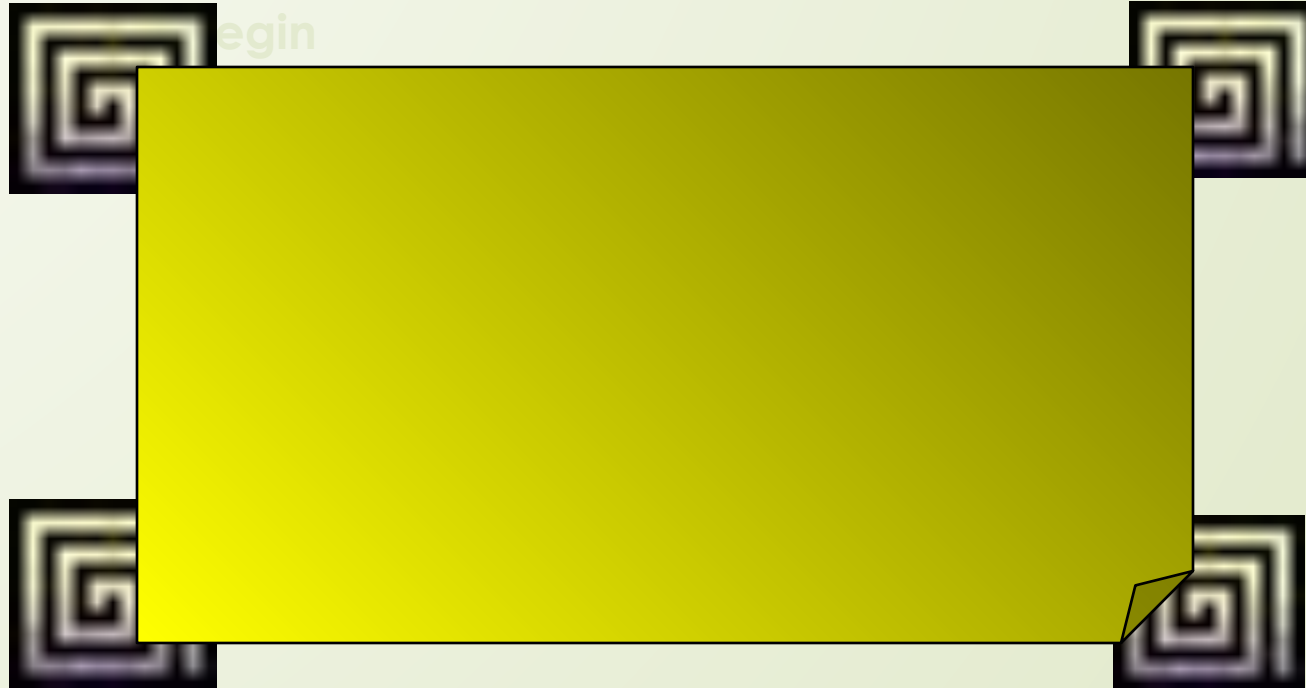
Если требуется сформировать массив, элементы которого лежат в диапазоне от  $a$  до  $b$ , используется выражение

**Random(b-a)+a.**



Обычно в паре с *Random* используется оператор *Randomize*, устанавливающий зависимость случайных чисел от времени запуска программы,

```
Randomize;  
For i:=1 to 10 do  
begin
```



## Нахождение суммы, произведения, количества элементов

| Сумма  | Произведение   | Количество   |
|--|--|--|
| <b>S:=0;</b><br><b>for I:=1 to n do</b><br><b>if &lt;условие&gt;</b><br><b>then S:=S+a(i);</b> | <b>P:=1;</b><br><b>for I:=1 to n do</b><br><b>if &lt;условие&gt;</b><br><b>then P:=P*a(i);</b> | <b>K=0;</b><br><b>for I=1 to n do</b><br><b>if &lt;условие&gt;</b><br><b>then K=K+1;</b> |

Здесь S - сумма, P – произведение, K – количество элементов массива A(n).

## Какие задачи решают следующие фрагменты программ?

|  |  |
|--|--|
| <p><b>a) S:=0;</b><br/><b>for I:=1 to n do</b><br/><b>if a(i) mod 2=0 then</b><br/><b>S:=S+a(i);</b></p>                   | <p><b>c) T:=0;</b><br/><b>for I:=1 to n do</b><br/><b>if a(i)&lt;=i then T:=T+1;</b></p> |
| <p><b>b) S:=0;</b><br/><b>for I:=1 to n do</b><br/><b>if (a(i) &lt;5) and (a(i)&gt;8.5)</b><br/><b>then S:=S+a(i);</b></p> | <p><b>d) R:=1;</b><br/><b>for I:=1 to n do</b><br/><b>R:=R*a(i);</b></p>                 |