

Программирование в Lazarus

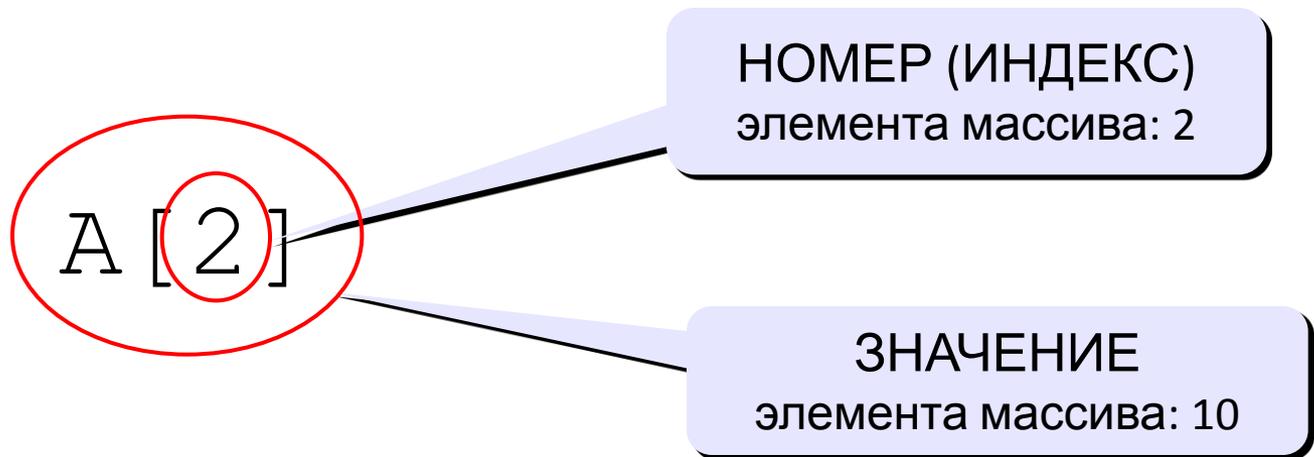
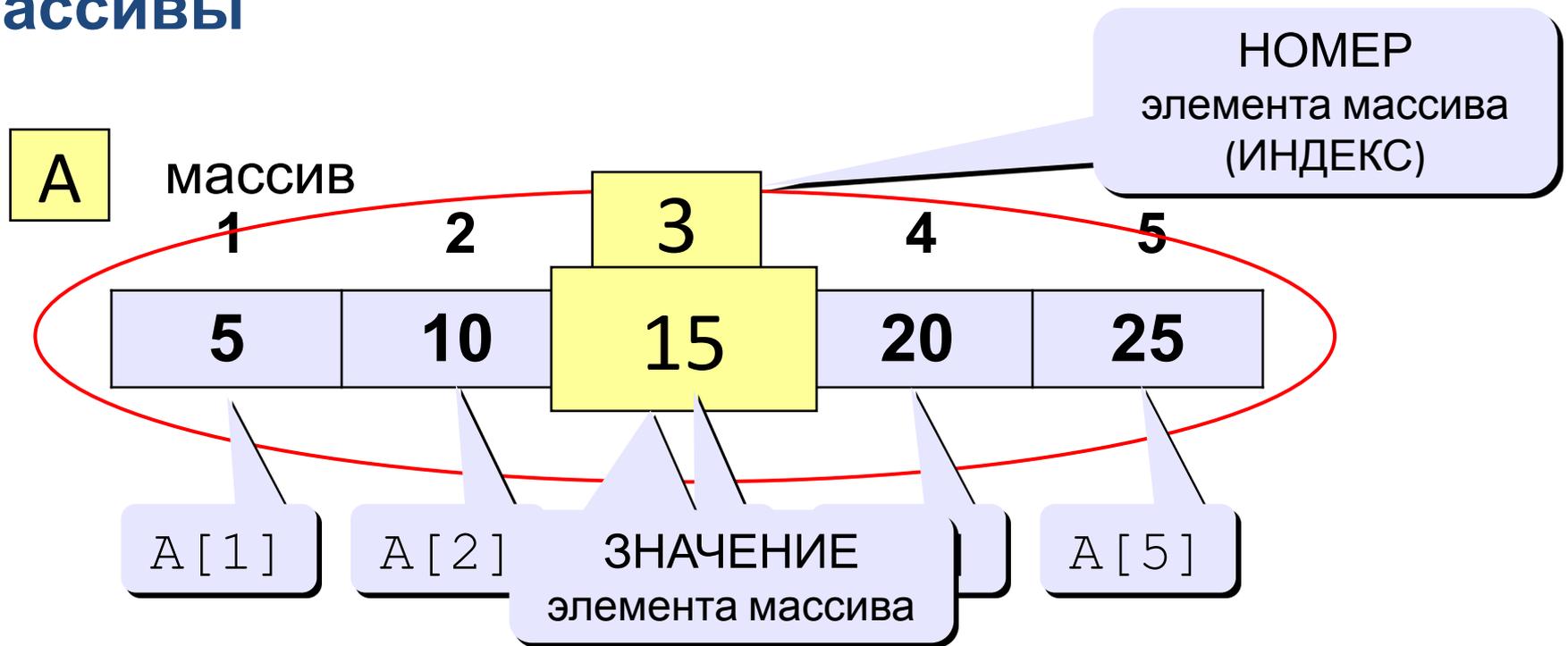
Тема **Массивы**

Массивы

Массив – это упорядоченная последовательность данных одного типа, имеющих общее имя и занимающих непрерывное место в памяти.

Данные в массиве называются элементами массива.

Массивы

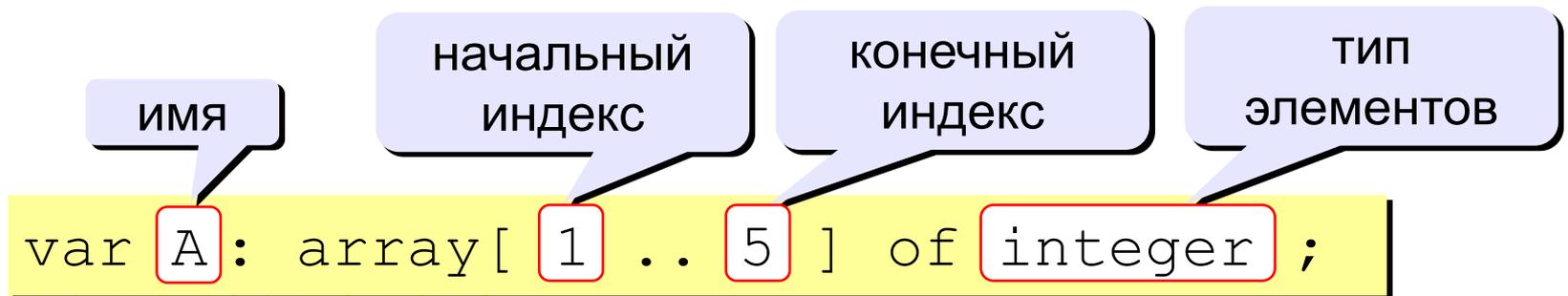


Объявление массивов

Зачем объявлять?

- определить **ИМЯ** массива
- определить **ТИП** массива
- определить **ЧИСЛО ЭЛЕМЕНТОВ**
- **ВЫДЕЛИТЬ МЕСТО В ПАМЯТИ**

Массив целых чисел:



Размер через константу:

```
const N=5;  
var A : array[1..N] of integer;
```

Объявление массивов. Примеры.

Массивы других типов:

```
var X, Y: array [1..10] of real;  
      C: array [1..20] of integer;
```

Другой диапазон индексов:

```
var Q: array [0..9] of real;  
      C: array [-5..13] of integer;
```

Ин

Что неправильно?

```
var a: array [1..1  
             0] of integer;  
...  
  A[5] := 4.5;
```

```
var a: array [1..10] of integer;  
...  
  A[0] := 15;
```

 Что не так?

```
var a: array [0..9] of integer;  
...  
  A[10] := 'X';
```

Массивы

Объявление:

```
const N = 5;  
var a: array[1..N] of integer;  
    i: integer;
```

Ввод массива из поля memo.

```
for i:=1 to frm1.mem1.count do  
    a[i]:=strtoint(frm1.mem1.lines[i-1]);
```

```
for i:=1 to N do a[i]:=a[i]+1;
```

```
Frm1.mem2.clear;  
for i:=1 to n do  
    frm1.mem2.Lines.Append(inttostr(a[i]));
```

Генератор случайных чисел в Паскале

Целые числа

```
var x: integer;
```

```
...
```

в интервале [0,N):

```
x := random ( 100 ); { интервал [0,99] }
```

в интервале [a,b]:

```
x := random(b-a+1) + a; { интервал [a,b] }
```

Вещественные числа

```
var x: real;
```

```
...
```

- в интервале [0,1)

```
x := random; { интервал [0,1) }
```

- в интервале [a,b]

```
x := random*(b-a) + a; { интервал [a,b] }
```

Заполнение массива случайными числами

```
procedure TFrm1.Btn1Click(Sender: TObject);  
var low,top:integer;  
    c:mas;  
begin  
    inp(low,top); //задать границы массива  
    zap(c,low,top); //заполнить массив случайными  
    числами  
    vivod(c); // вывести в поле MEMO  
end;
```

```
const n=10;
type mas=array[1..n] of integer;

procedure inp(var n,m:integer);
begin
  n:=strtoint(frm1.edt1.text);
  m:=strtoint(frm1.edt2.text);
end;

procedure zap(var c:mas; niz,ver:integer);
var i:integer;
begin
  for i:=1 to n do
    c[i]:=random(ver-niz+1)+niz;
end;

procedure vivod(c:mas);
var i:integer;
begin
  frm1.mem1.lines.clear;
  for i:=1 to n do
    frm1.mem1.Lines.Append(inttostr(c[i]));
end;
```

Программирование в Lazarus

Тема Обработка массивов

Подсчет элементов

Задача:

Заполнить массив случайными числами и подсчитать количество нулевых элементов.

Решение:

- 1) записать в счётчик ноль
- 2) просмотреть все элементы массива:
*если очередной элемент = 0, то
увеличить счётчик на 1*
- 3) вывести значение счётчика

Подсчет количества нулевых элементов

```
const N = 5;
Type mas=array [1..N] of integer;
Var A:mas;
    i, count: integer;
...
begin
    { здесь надо заполнить массив }
    count:= 0;
    for i:=1 to N do
        if A[i] = 0 then count:= count + 1;
    lb11.Caption:=inttostr(count);
End;
```

перебираем все
элементы массива

Сумма выбранных элементов

Задача:

Заполнить массив случайными числами в интервале и подсчитать сумму положительных элементов.

Используем переменную S для накопления суммы.

$$S := 0$$
$$S := A[1]$$

$$S := A[1] + A[2]$$

$$S := A[1] + A[2] + A[3] \longrightarrow S := A[1] + A[2] + \dots + A[N]$$

Сумма выбранных элементов

Задача:

Заполнить массив случайными числами и подсчитать сумму положительных элементов.

Решение:

- 1) записать в переменную S ноль
- 2) просмотреть все элементы массива:
*если очередной элемент > 0 ,
то добавить к сумме этот элемент*
- 3) вывести значение суммы

Сумма выбранных элементов

```
const N = 5;  
Type mas= array [1..N] of integer;  
var  A:mas;  i, S: integer;  
  
...  
begin  
    { здесь надо заполнить массив }  
    S:= 0;  
    for i:=1 to N do  
        if A[i] > 0 then S:= S + A[i];  
  
    lbl1.Caption:=inttostr(S) ;  
End;
```

перебираем все
элементы массива