

# Строки в Pascal

Тарасенко О.П.  
Учитель информатики  
МБОУ МО г.Нягань «СОШ№2»

**Строка** - это ограниченная апострофами последовательность любых символов (массив символов), т.е. элементов типа **char**.

В Паскале строке соответствует тип данных **String**.

Описание строки имеет вид:

НАЗВАНИЕ ОПЕРАТОРА	PASCAL
Описание строки с ограниченным количеством символов	<b>var s1:string[ N];</b>
Описание строки с произвольным количеством символов	<b>var s1:string;</b>
Присвоение значения строковой переменной	<b>s1:=' строка'</b>

**s1** – имя строковой переменной;

**N** – количество символов от 1 до 255;

**строка** – символьная строка.



### Буквы

строчные  
a-z a-я

ПРОПИСНЫЕ  
A-Z A-Я

### Цифры

1 2 3 9 0

### Пробелы

□ □ □ □

### Спец. символы

@ # & > %

### Знаки препинания

! ? . ( «

# Действие над строками и переменными

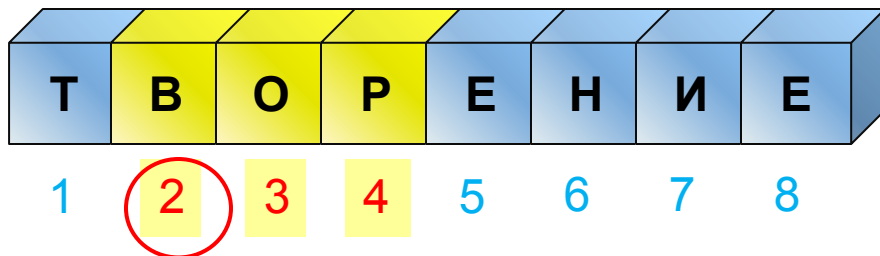
НАЗВАНИЕ ОПЕРАТОРА

Копирование фрагмента строки

PASCAL

`copy(и_строк,поз,N);`

ПРИМЕР



`t:=copy(A,2,3);`

`и_строк` – имя строковой переменной;  
`N` – количество символов (от 1 до 255);  
`поз` – номер начального символа

# Действие над строками и переменными

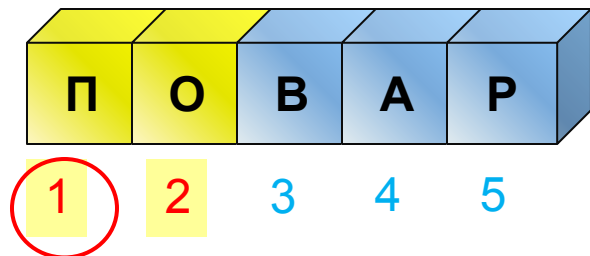
НАЗВАНИЕ ОПЕРАТОРА

Удаление символов из строки

PASCAL

```
delete(и_строк,поз,N);
```

ПРИМЕР



```
t:=delete (stroka1,1,2);
```

**и\_строк** – имя строковой переменной;  
**N** – количество символов (от 1 до 255);  
**поз** – номер начального символа

# Действие над строками и переменными

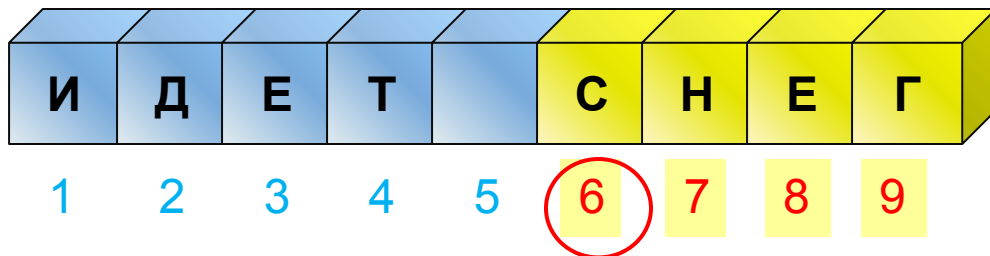
НАЗВАНИЕ ОПЕРАТОРА

Вставка строки

PASCAL

```
insert(п_строк,и_строк,поз);
```

ПРИМЕР



```
insert (m4,m1,2);
```

**и\_строк** – имя строковой переменной;  
**п\_строк** – подстрока (набор символов);  
**N** – количество символов (от 1 до 255);  
**поз** – номер начального символа

# Действие над строками и переменными

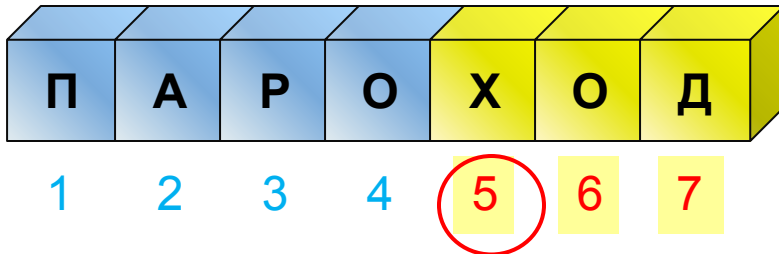
НАЗВАНИЕ ОПЕРАТОРА

Слияние строки

PASCAL

`concat(и_строк1,и_строк2);` или  
`и_строк1+и_строк2`

ПРИМЕР



`new:=concat (p,x);`  
или `p+x`

`и_строк` – имя строковой переменной;  
`п_строк` – подстрока (набор символов);  
`N` – количество символов (от 1 до 255);  
`поз` – номер начального символа

❑ Программа производит объединение строк.

```
var x,y,z:string [10];  
begin  
    x:= 'паро';  
    y:= 'ход';  
    z:= x+y;  
    write (z);  
readln;  
readln;  
end.
```



# Действие над строками и переменными

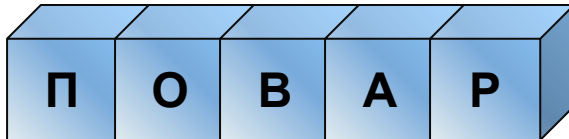
## НАЗВАНИЕ ОПЕРАТОРА

Определение длины строки

## PASCAL

```
length(и_строк);
```

## ПРИМЕР



1 2 3 4 5

```
dlina:=length (stroka1);
```

**и\_строк** – имя строковой переменной;  
**п\_строк** – подстрока (набор символов);  
**N** – количество символов (от 1 до 255);  
**поз** – номер начального символа

# Длина строки

**Program E29F;**

**var x,y :string [20]; k,l,n: integer;**

**begin**

**writeln ('Введите две строки');**

**readln(x);**

**readln(y);**

**k:=length(x); l:=length(y); n:=length(x+y);**

**writeln ('длина первой строки':25, 'длина второй строки':25);**

**writeln(k:25,l:25);**

**writeln(x+y, 'длина строки',n);**

**readln;**

**readln;**

**end.**

# Действие над строками и переменными

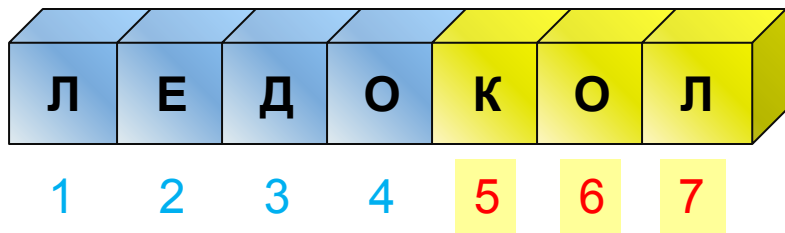
НАЗВАНИЕ ОПЕРАТОРА

Поиск в строке подстроки

PASCAL

```
pos(и_строк,п_строк);
```

ПРИМЕР



```
n:=pos (s,'КОЛ');
```

**и\_строк** – имя строковой переменной;  
**п\_строк** – подстрока (набор символов);  
**N** – количество символов (от 1 до 255);  
**поз** – номер начального символа

- ❑ Программа производит перемещение элементов в обратном порядке.

```
Program E28;  
const n=15;  
var a,b:array[1..n] of char;  
i:integer;  
begin  
  for i:=1 to n do  
    begin  
      read (a[i]);  
      b[n-i+1]:=a[i];  
    end;  
  for i:= 1 to n do  
    write (b[i]);  
readln;  
readln;  
end.
```

# Присваивание

```
var x :string [6];  
begin  
    x:= 'МИМ'+ 'озабоченный';  
    write (x);  
readln;  
readln;  
end.
```

**результат «МИМОЗА»**

**Спасибо за внимание!**