

Строки

Функции и процедуры для работы со строками

Строка – это последовательность СИМВОЛОВ.

- Каждый символ занимает 1 байт памяти (код ASCII).
- Количество символов в строке называется ее длиной(от 0 до 255).



Пример констант :

' Основы программирования'
'8-918-456-45-**'.

- Строковая переменная описывается следующим образом:

Var <идентификатор> : **String** [<max длина строки>];

Например:

Var Name : String[20];

- Символы внутри строки индексируются, начиная с единицы. Отдельный символ идентифицируется именем строки с индексом, заключенным в квадратные скобки.
- Например: `Name[5]`.

1	2	3	4	5	6	7
Е	В	Г	Е	Н	И	Й

Name[5] -
Н

Индекс может быть положительной константой, переменной величиной и выражением целого типа. Значение индекса не должно выходить за границы описания.

- Пример: Name[i], Slovo[k+1], Family[7]

Тип **String** и стандартный тип **Char** совместимы.

Строковые выражения строятся из **строковых констант, переменных, функций и знаков операций.**

Операции над строковыми данными

- **Операция сцепления (+)** применяется для соединения нескольких строк в одну результирующую строку. Сцеплять можно как строковые константы, так и переменные.
- Например:

‘Мега’+‘байт’

В результате получится строка *‘Мегабайт’*

Длина результирующей строки ≤ 255 символов

- **Операции отношений** (**=, <, >, <=, >=, <>**) производят сравнение двух строк, в результате чего получают логическую величину (**True** или **False**).
- Сравнение строк производится слева направо до первого несовпадающего символа, и большей считается та строка, в которой первый несовпадающий символ имеет больший номер в таблице символьной кодировки.

- Строки равны, если они полностью совпадают по длине и содержат одни и те же символы. Например (найти ошибки):

Выражение	Результат
<code>'film1' < 'film2'</code>	True
<code>'pascal' > 'Pascal'</code>	True
<code>'ФАЙЛ_' <> 'ФАЙЛ'</code>	True
<code>'MSDOS' = 'MSDOS'</code>	False

Функции и процедуры для работы со строками

1. Функция **Copy(S, Poz, N)** – выделяет из строки **S** подстроку длиной **N** символов, начиная с позиции **Poz**.

N и **Poz** – целочисленные выражения.

Например:

Значение S	Выражение	Результат
‘КАТАЛОГ’	Copy(S, 5, 3)	‘ЛОГ’
‘PASCAL’	Copy(S, 2, 2)	‘AS’

2. Функция **Concat (S1, S2 , ..., SN)** выполняет сцепление (конкатенацию) строк **S1, ..., SN** в одну строку.

Например:

Значение S	Выражение	Результат
‘Гига’	Concat(S, ‘ба’, ‘йт’)	‘Гигабайт’

3. Функция **Length(S)** – определяет текущую длину строки **S**. Результат - значение целого типа.

Например:

Значение S	Выражение	Результат
'Pascal_'	Length(S)	7

4. Функция **Pos (S1, S2)** – обнаруживает первое появление в строке **S2** подстроки **S1**. В результате получают целое число, равное номеру позиции, где находится первый символ подстроки **S1**. Если в **S2** подстроки **S1** не обнаружено, результат равен 0.

Например:

Значение S2	Выражение	Результат
'Капитан'	Pos ('ит', S2)	4
'каталог'	Pos ('k', S2)	0

5. Процедура **Delete(S, Poz, N)** – удаляет **N** символов из строки **S**, начиная с позиции **Poz**.

Например:

Исходное значение S	Оператор	Конечное значение S
‘Каталог’	Delete(S, 3, 2)	‘Калог’
‘Компьютер’	Delete(S, 2, 8)	‘К’

6. Процедура **Insert (S1, S2, Poz)** – вставляет строку **S1** в строку **S2**, начиная с позиции **Poz**.

Исходное значение S2	Оператор	Конечное значение S2
‘ЭВМ РС’	Insert(‘IBM-’, S2,5)	‘ЭВМ IBM-PC’

- Функция **Ord**, преобразовывает букву в ее числовой код.
- Функция **Chr** противоположна функции **Ord**, преобразовывает числовой код символа в сам символ.

Пример:

```
Program number;  
var N:integer;  
    X:char;  
begin  
    X:='L';  
    writeln (X);  
    N:=Ord (X);  
    writeln (N);  
    X:=Chr (N);  
    writeln(X);  
readln  
end.
```

Зд1: Получить из слова «ВЕЛИЧИНА» слова «НАЛИЧИЕ» с использованием функций для работы со строками.

```
Program Получитьслово;  
Var S1, S2 : String[10];  
Begin  
  S1:= 'ВЕЛИЧИНА';  
  S2:= Copy(S1, 7, 2) + Copy(S1, 3, 4) + S1[2];  
  WriteLn(S2)  
End.
```

Зд.2: Подсчитать в символьной строке количество цифр, предшествующих первому символу «!».

Program Количество;

Var S : String;

K, i : Byte;

Begin

WriteLn('Введите строку');

ReadLn(S);

K:=0;

i:=1;

While (i <= Length(S)) **And** (S[i] <> '!') **do Begin**

If (S[i] >= '0') **And** (S[i] <= '9') **Then** K:=K+1;

i:=i+1

End;

WriteLn('Количество цифр до символа «!» равно ', K)

End.

Задания для самостоятельного решения:

1. Составить программу получения из слова «дисковод» слова «воск», используя операцию сцепления и функцию Copy.
2. Составить программу получения слова «правило» из слова «операция», используя процедуры Delete и Insert.

Домашнее задание:

- теория;
- **Задача1.** В заданной строке удалить все пробелы.