Символьные строки. Посимвольная обработка строк.

С помощью компьютера можно решать весьма разнообразные задачи обработки текста: от составления платежных ведомостей до верстки газет. При обработке текстов компьютер должен уметь оперировать не только числами, но и различными нецифровыми символами. Познакомимся с основными приемами обработки текста на компьютере. Тема нашего урока: «Символьные строки. Посимвольная обработка строк».

Будем полагать, что ***текст*** — это произвольная последовательность символов некоторого алфавита. Алфавитом может служить любое множество символов. Символьный тип данных служит для представления символа, который есть на клавиатуре. Всем имеющимся на клавиатуре символам ставится в соответствие целое число — код символа. Всего кодируется 256 символов. В таблице кодов среди всех символов можно выделить следующие четыре группы:

1. цифры от 0 (код 48) до 9 (код 57);
2. латинские прописные буквы от «A» (код 65) до «Z» (код 90);
3. латинские строчные буквы от «a» (код 97) до «z» (код 122);
4. русские прописные буквы от «А» (код 128) до «Я» (код 159).

Символы в пределах каждой из групп следуют друг за другом в алфавитном порядке и их коды изменяются при этом на 1. Каждый символ строковой величины занимает 1 байт памяти.

Данные символьного типа могут быть представлены двумя способами:

1.графическим, когда соответствующий символ записывается в апострофах, например: '+' (символ «плюс»);

2.с помощью целочисленного кода, когда рядом со знаком целочисленного кода # указывается целое число — код данного символа в таблице ASCII. Например, #65 — представление латинской прописной буквы «A» с помощью целочисленного кода.

Строкой символов, или символьной (строковой, текстовой) константой, будем называть последовательность символов, заключенную в кавычки. Среди строк

пустая строка играет ту же роль, что ноль среди чисел. Максимальная длина текстовой строки — 255 символов.

В Паскале существует тип данных, предназначенный для обработки слов (цепочки символов). *Такой тип данных называется строковым (тип string).* Строки выводятся на экран посредством стандартной процедуры write {writeln) и вводятся с клавиатуры с помощью стандартной процедуры read {readlrn).

Объявление данных типа string:

1.если данные являются константами:

const <имя_переменной>=<значение>;

1.если данные являются переменными:

var

<имя_переменной>: string;

<имя_переменной>: string[20];

В квадратных скобках указывается максимальная длина строки. Если длина строки не указана, то она считается равной 255 символам — максимально возможной длине.

Операции над строками.

Для строк определена ***операция объединения (сцепления),*** которая обозначается знаком + и объединяет несколько строк в одну.

Например:

A:=’ Новосибирск’;

B:=’ столица Сибири’

C:=A+’- ’+B+’!’

Переменная C иметь значение ‘Новосибирск- столица Сибири!’

Операции отношения (сравнения двух строк) **=,<, >,** >=,<=, <> .В результате сравнения двух строк получается логическое значение (true или false). Сравнение происходит слева направо до первого несовпадающего символа, и та строка считается больше, в которой первый несовпадающий символ имеет больший номер в таблице символов (кодировки).

Например:

1. 'строка'<>'строки'(верно, т.к. не совпадают последние символы),
2. 'Abc'<'abc'(отношение истинно, т.к. код символа 'A' равен 65 в десятичной системе счисления, а код символа 'a' - 97),
3. 'год'>'век'(отношение верно, т.к. буква 'г' в алфавите стоит после буквы 'в', а, следовательно.
4. ‘кот ’=’ кот’ (отношение истинно, т.к совпадают все символы)

Закрепление

Задание 1

Поставьте знак сравнения (>, <, =) между парами строк и объясните свой ответ.

'Процессор'____'Процесс'

'Balkon'___'balkon'

'balkon'____'balken'

'кошка '____'кошка'

'коТ'_____'кот'

Задание 2

Что будет выведено на экран в результате выполнения следующей программы:

```
var s: string;
i, j: integer;
begin
s := 'программа';
for i := 1 to length (s) do
begin
for j := 1 to i - 1 do write(' ');
writeln (s[i])
end;
end.
```