

## Практическая работа № 7

### Работа со строками

**Цель работы:** Изучить строковый тип string. Изучить стандартные функции работы со строками, научиться выполнять стандартные операции со строками.

#### Содержание работы.

#### Основные понятия.

1 Строка представляет собой особую форму одномерного массива символов, которая имеет существенное отличие. Массив символов имеет фиксированную длину (количество элементов), которая определяется при описании. Строка имеет две разновидности длины:

- Общая длина строки, которая характеризует размер памяти, выделяемый строке при описании;
- Текущая длина строки (всегда меньше или равна общей длине), которая показывает количество смысловых символов строки в каждый конкретный момент времени.

2 Строка – упорядоченная последовательность символов. Количество символов в строке называется ее длиной. Длина строки может лежать в диапазоне от 0 до 255. Каждый символ строковой величины занимает 1 байт памяти и имеет числовой код в соответствии с таблицей кодов ASCII.

3 Для описания строковых переменных существует предопределенный тип string. В общем виде описание строковой переменной будет выглядеть следующим образом:

Var <имя\_переменной>: string[<максимальная длина строки>]

4 Операция слияния (сцепления, конкатенации) применяется для соединения нескольких строк в одну, обозначается знаком «+». Операция слияния применима для любых строковых выражений, как констант, так и переменных.

5 Операции отношения позволяют сравнивать строки на отношение равенства (=), неравенства (<>), больше (>), меньше (<), больше или равно (>=), меньше или равно (<=). В результате сравнения двух строк получается логическое значение (true или false). Сравнение строк производится слева направо по символу до первого несовпадающего символа, большей считается та строка, в которой первый несовпадающий символ имеет больший код в таблице кодировки. Если строки имеют различную длину, но в общей части символы совпадают, считается, что короткая строка меньше. Строки равны, если они имеют равную длину и соответствующие символы совпадают.

#### 6 Стандартные функции для работы со строками

- Сору (S, poz, n) выделяет из строки S, начиная с позиции poz, подстроку из n символов. Здесь S – любое строковое выражение, poz, n – целочисленные выражения.

- Concat ( $s_1, s_2, \dots, s_n$ ) выполняет слияние строк  $s_1, s_2, \dots, s_n$  в одну строку.
- Length( $S$ ) определяет текущую длину строкового выражения  $S$ . Результат – значение целого типа.
- Pos(sub $S$ ,  $S$ ) определяет позицию первого вхождения подстроки sub $S$  в строку  $S$ . Результат – целое число, равное номеру позиции, где находится первый символ искомой подстроки. Если вхождение подстроки не обнаружено, то результат функции будет равен 0.

#### 7 Стандартные процедуры для работы со строками:

- Delete ( $S$ , poz,  $n$ ) удаляет из строки  $S$ , начиная с позиции poz, подстроку из  $n$  символов. Здесь  $S$  – строковая переменная (в данном случае нельзя записать никакое другое строковое выражение, кроме имени строковой переменной, т.к. только с именем переменной связана область памяти, куда будет помещен результат выполнения процедуры); poz,  $n$  – любые целочисленные выражения.
- Insert(sub $S$ ,  $S$ , poz) вставляет в строку  $S$ , начиная с позиции poz, подстроку sub $S$ . Здесь sub $S$  – любое строковое выражение,  $S$  – строковая переменная (именно ей будет присвоен результат выполнения процедуры), poz – целочисленное выражение.

#### 8 Процедуры преобразования типов:

- Str( $x$ ,  $S$ ) преобразует число  $x$  в строковый формат. Здесь  $x$  – любое числовое выражение,  $S$  – строковая переменная. В процедуре есть возможность задавать формат числа  $x$ . Например, str( $x$ : 8: 3,  $S$ ), где 8 – общее число знаков в числе  $x$ , а 3 – число знаков после запятой.
- Val( $S$ ,  $x$ , kod) преобразует строку символов  $S$  в число  $x$ . Здесь  $S$  – строковое выражение,  $x$  – числовая переменная (*именно туда будет помещен результат*), kod – целочисленная переменная (*мина integer*), которая равна номеру позиции в строке  $S$ , начиная с которой произошла ошибка преобразования, если преобразование прошло без ошибок, то переменная kod равна 0.

#### Задание

- 1 Составить и записать алгоритм решения задачи
- 2 Написать программу
- 3 Ввести программу и запустить её на исполнение.
- 4 Протестировать алгоритм с различными данными.
- 5 Отладить программу (найти и исправить возможные ошибки).
- 6 Проанализировать программу и сделать вывод.

#### Исходные данные:

Дана строка. Определить, сколько раз в нее входит группа букв ешь.

#### Решение:

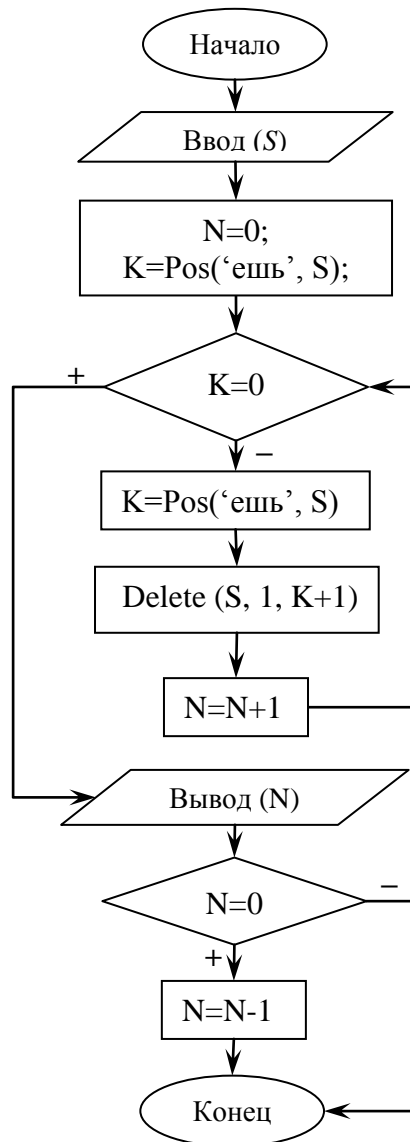
- 1 Алгоритм задачи:

а) для поиска первого вхождения сочетания «ешь» используем функцию Pos, она покажет первый символ к;

б) дальше надо отрезать от строки символы с первого по к+1 и снова воспользоваться указанной функцией;

в) выполнять эти действия, пока значение Pos не станет нулевым, но когда это произошло, N все равно увеличивается на 1, поэтому печатаем N – 1.

Получим алгоритм:



1. Начало;
2. Ввод (S);
3. N=0;
4. K=Pos('ешь', S);
5. Пока K <> 0 выполнять  
нц  
K=Pos('ешь', S);  
Delete (S, 1, K+1);  
N=N+1;  
кц
6. Если N>0 то N=N+1;
7. Вывод (N-1);
8. Конец.

2 Программа:

```
program stroka1;
var K, N:integer;
    S:string;
begin
    S:= edit1.text;
    N:=0;
```

```

K:=Pos('ешь', S);
while K<>0 do
begin
    K:=Pos('ешь', S);
    N:=N+1;
    Delete (S, 1, K+1);
end;
if N>0 then N:=N-1;
Edit3.Text:='В тексте ешь встречается ', N, ' раз'); end.

```

- 3 Программный код для запуска      4 Тестирование программы с различными данными:

The image shows two side-by-side screenshots from a Pascal IDE. The left screenshot displays the source code of a program named 'stroka1.pas'. The code defines variables K (integer) and N (integer), and S (string). It prompts the user to enter text, reads it into S, and then enters a loop where it counts the occurrences of the substring 'ешь' in S. After each occurrence is found, it deletes the substring and increments N. After the loop, it decrements N by 1 and displays the final count. The right screenshot shows the program's execution. It prompts for text input, and two examples are shown. In the first example, the input is 'переживаешь? перебедаешь? садись на диету!', and the output is 'В тексте ешь встречается 2 раз'. In the second example, the input is 'Раз, два, три, четыре, пять! Вышел зайчик погулять', and the output is 'В тексте ешь встречается 0 раз'.

```

*stroka1.pas
program stroka1;
var K, N:integer;
    S:string;
begin
    writeln('Введите текст');
    readln(S);
    N:=0;
    K:=Pos('ешь', S);
    while K<>0 do
    begin
        K:=Pos('ешь', S);
        N:=N+1;
        Delete (S, 1, K+1);
    end;
    if N>0 then N:=N-1;
    writeln('В тексте ешь встречается ', N, ' раз');
end.

```

Введите текст  
переживаешь? перебедаешь? садись на диету!  
В тексте ешь встречается 2 раз

Введите текст  
Раз, два, три, четыре, пять! Вышел зайчик погулять  
В тексте ешь встречается 0 раз

- 6 Программа выдает верное значение, если заданная комбинация букв присутствует в тексте и если не присутствует.

### Задания к практической работе.

1 Даны целые положительные числа  $n_1$  и  $n_2$  и строки  $S_1$  и  $S_2$ . Получить из этих строк новую строку, содержащую первые  $n_1$  символов  $S_1$  и последние  $n_2$  символов строки  $S_2$ .

2 Даны целое положительное число  $n$  и строка  $S$ . Преобразовать строку  $S$  в строку длины  $n$  следующим образом: если длина  $S$  больше  $n$ , то отбросить первые символы, если меньше, то в начало добавить «.».

3 Даны строки  $S_1$  и  $S_2$ . Проверить, содержится ли  $S_1$  в строке  $S_2$ .

4 Даны строки  $S_1$  и  $S_2$ . Удалить из строки  $S_1$  подстроку  $S_2$ , если  $S_2$  не содержится в  $S_1$ , вывести  $S_1$  без изменений

5 Даны строки  $S_1$  и  $S_2$ . Удалить из строки  $S_1$  первую подстроку  $S_2$ , если  $S_2$  не содержится в  $S_1$ , вывести  $S_1$  без изменений

6 Даны строки  $S_1$  и  $S_2$ . Удалить из строки  $S_1$  последнюю подстроку  $S_2$ , если  $S_2$  не содержится в  $S_1$ , вывести  $S_1$  без изменений

7 Даны строки  $S_1$  и  $S_2$ . Удалить из строки  $S_1$  все подстроки  $S_2$ , если  $S_2$  не содержится в  $S_1$ , вывести  $S_1$  без изменений

8 Даны строки  $S$ ,  $S_1$  и  $S_2$ . Заменить в строке  $S$  последнюю подстроку  $S_1$  на  $S_2$ , если  $S_1$  не содержится в  $S$ , вывести  $S$  без изменений

9 Даны строки  $S$ ,  $S_1$  и  $S_2$ . Заменить в строке  $S$  первую подстроку  $S_1$  на  $S_2$ , если  $S_1$  не содержится в  $S$ , вывести  $S$  без изменений

10 Даны строки  $S$ ,  $S_1$  и  $S_2$ . Заменить в строке  $S$  все подстроки  $S_1$  на  $S_2$ , если  $S_1$  не содержится в  $S$ , вывести  $S$  без изменений

11 Дано полное имя файла, т.е. путь к файлу, имя и расширение. Выделить из строки только имя файла без расширения.

12 Дано полное имя файла, т.е. путь к файлу, имя и расширение. Выделить из строки только расширение файла без точки.

13 Дано полное имя файла, т.е. путь к файлу, имя и расширение. Выделить из строки только имя первого каталога без символа \.

14 Дано полное имя файла, т.е. путь к файлу, имя и расширение. Выделить из строки только имя последнего каталога без символа \.

15 Дана строка, содержащая круглые скобки. Выдать сообщение, правильно ли расставлены скобки (количество открывающихся соответствует количеству закрывающихся)

16 Строка содержит одно слово. Проверить, будет ли оно читаться одинаково слева направо и наоборот.

17 В строке заменить все пробелы на «\_», посчитать количество замененных символов

18 В строке заменить все пробелы на «\*», посчитать количество замененных символов

- 19 Дана строка. Определить, сколько в ней «,»
- 20 Дана строка. Определить, сколько в ней «;»
- 21 Удалить часть символьной строки в скобках (вместе со скобками)
- 22 Удалить часть символьной строки в кавычках (вместе с кавычками)
- 23 Дана строка, содержащая кавычки. Выдать сообщение, правильно ли расставлены кавычки (количество открывающихся соответствует количеству закрывающихся)
- 24 Дана строка. Найти количество слов, начинающихся с буквы А
- 25 Дана строка. Найти количество слов, начинающихся с буквы А
- 26 Дана строка. Заменить первую букву слов, начинающихся с буквы А, на малую букву
- 27 Дана строка. Заменить последнюю букву слов, заканчивающихся на е, на большую букву
- 28 Дана строка. Найти количество букв к.
- 29 Дана строка. Определить, сколько раз в нее входит группа букв ель
- 30 Дана строка. Определить, сколько раз в нее входит группа букв ого