

Программирование на алгоритмическом языке

§ 66. Символьные строки

Зачем нужны символьные строки?

симтаб `s[1:80]` | **массив символов**

- ⊖ элементы массива – отдельные объекты
- сложно работать со строками переменной длины

Хочется:

- строка – единый объект
- длина строки может меняться во время работы программы

лит `s` | **символьная строка**

литерный тип

Символьные строки

Присваивание:

```
s := 'Вася пошёл гулять'
```

ЛИТ s

Ввод с клавиатуры:

```
ВВОД s
```

Вывод на экран:

```
ВЫВОД s
```



А если массив?

Отдельный символ:

```
s[4] := 'a'
```

Длина строки:

```
цел n
```

```
n := длин(s)
```

Символьные строки

Задача: заменить в строке все буквы 'а' на буквы 'б'.

```
алг Замена а на б
нач
  ЛИТ S
  ВЫВОД 'Введите строку: '
  ВВОД S
  ЦЕЛ i
  НЦ для i от 1 до ДЛИН(S)
    ЕСЛИ S[i] = 'а' ТО
      S[i] := 'б'
    ВСЕ
  КЦ
  ВЫВОД S
КОН
```

Задачи

«А»: Ввести с клавиатуры символьную строку и заменить в ней все буквы «а» на «б» и все буквы «б» на «а» (заглавные на заглавные, строчные на строчные).

Пример:

Введите строку:

ааббААББссСС

Результат:

ббааББААссСС

Задачи

«В»: Ввести с клавиатуры символьную строку и определить, сколько в ней слов. Словом считается последовательности непробельных символов, отделенная с двух сторон пробелами (или стоящая с краю строки). Слова могут быть разделены несколькими пробелами, в начале и в конце строки тоже могут быть пробелы.

Пример:

Введите строку:

Вася пошел гулять

Найдено слов: 3

Задачи

«С»: Ввести с клавиатуры символьную строку и найдите самое длинное слово и его длину. Словом считается последовательности непробельных символов, отделенная с двух сторон пробелами (или стоящая с краю строки). Слова могут быть разделены несколькими пробелами, в начале и в конце строки тоже могут быть пробелы.

Пример:

Введите строку:

Вася пошел гулять

Самое длинное слово: гулять, длина 6

Операции со строками

Объединение (конкатенация) :

```
s1 := 'Привет'
```

```
s2 := 'Вася'
```

```
s := s1 + ', ' + s2 + '!'
```

'Привет, Вася!'

Срез:

```
s := '123456789'
```

```
s1 := s[3:7] | '34567'
```

с какого символа

до какого символа

Операции со строками

Удаление:

```
s := '123456789'
```

```
удалить (s, 3, 6) | '129'
```

с какого
СИМВОЛА

СКОЛЬКО
СИМВОЛОВ

Вставка:

```
s := '123456789'
```

```
вставить ('ABC', s, 3) | '12ABC3456789'
```

что

куда

с какого
СИМВОЛА

Поиск в строках

```
s := 'Здесь был Вася.'
```

что где

```
n := позиция('с', s)
```

```
если n > 0 то
```

```
    вывод 'Номер символа ', n
```

```
иначе
```

```
    вывод 'Символ не найден.'
```

```
все
```



Находит первое слева вхождение подстроки!

Пример обработки строк

Задача: Ввести имя, отчество и фамилию. Преобразовать их к формату «фамилия-инициалы».

Пример:

Введите имя, отчество и фамилию:

Василий Алибабаевич Хрюндиков

Результат:

Хрюндиков В.А.

Алибабаевич Хрюндиков

Алгоритм:

- найти первый пробел и выделить имя
- удалить имя с пробелом из основной строки
- найти первый пробел и выделить отчество
- удалить отчество с пробелом из основной строки
- «сцепить» фамилию, первые буквы имени и фамилии, точки, пробелы...

Хрюндиков

Хрюндиков В.А.

Пример обработки строк

```
алг FIO
нач
  лит s, name, name2
  цел n
  вывод 'Введите имя, отчество и фамилию'
  ввод s
  n := позиция(' ', s);
  name := s[1:n-1] | вырезать имя
  удалить(s, 1, n)
  n := позиция(' ', s)
  name2 := s[1:n-1] | вырезать отчество
  удалить(s, 1, n) | осталась фамилия
  s := s + ' ' + name[1] + '.' + name2[1] + '.'
  вывод s
кон
```

Задачи

«А»: Ввести с клавиатуры в одну строку фамилию, имя и отчество, разделив их пробелом. Вывести фамилию и инициалы.

Пример:

Введите фамилию, имя и отчество:

Иванов Петр Семёнович

П.С. Иванов

Задачи

«В»: Ввести адрес файла и «разобрать» его на части, разделенные знаком ' / '. Каждую часть вывести в отдельной строке.

Пример:

Введите адрес файла:

C: /фото/2013/Поход/vasya.jpg

C:

фото

2013

Поход

vasya.jpg

Задачи

«С»: Напишите программу, которая заменяет во всей строке одну последовательность символов на другую.

Пример:

Введите строку:

`(X > 0) and (Y < X) and (Z > Y) and (Z <> 5)`

Что меняем: `and`

Чем заменить: `&`

Результат

`(X > 0) & (Y < X) & (Z > Y) & (Z <> 5)`

Преобразования «строка» – «число»

Из строки в число:

```
s := '123'
```

да или нет

цел N, вещ X,
лит s, лог ОК

```
N := лит_в_цел(s, ОК) | N = 123
```

```
если не ОК то вывод 'Ошибка!' все
```

```
s := '123.456';
```

```
X := лит_в_вещ(s, ОК) | X = 123.456
```

```
если не ОК то вывод 'Ошибка!' все
```

Из числа в строку:

```
N := 123
```

```
s := цел_в_лит(N) | '123'
```

```
X := 123.456
```

```
s := вещ_в_лит(X) | '123.456'
```


Задачи

«А»: Напишите программу, которая вычисляет сумму трех чисел, введенную в форме символьной строки. Все числа целые.

Пример:

Введите выражение :

12+3+45

Ответ: 60

«В»: Напишите программу, которая вычисляет выражение, состоящее из трех чисел и двух знаков (допускаются только знаки «+» или «-»). Выражение вводится как символьная строка, все числа целые.

Пример:

Введите выражение :

12-3+45

Ответ: 54

Задачи

«С»: Напишите программу, которая вычисляет выражение, состоящее из трех чисел и двух знаков (допускаются знаки «+», «-», «*» и «/»). Выражение вводится как символьная строка, все числа целые. Операция «/» выполняется как целочисленное деление (`div`).

Пример:

Введите выражение :

12*3+45

Ответ: 81