

Программирование на языке Python

§ 62. Массивы

§ 63. Алгоритмы обработки массивов

§ 64. Сортировка

§ 65. Двоичный поиск

§ 66. Символьные строки

§ 67. Матрицы

§ 68. Работа с файлами

Программирование на языке Python

§ 66. Символьные строки

Операции со строками

Объединение (конкатенация) :

```
s1 = "Привет"
```

```
s2 = "Вася"
```

```
s = s1 + ", " + s2 + "!"
```

"Привет, Вася!"

Срезы:

```
s = "0123456789"
```

```
s1 = s[3:8] # "34567"
```

ЭТОТ СИМВОЛ НЕ
ВХОДИТ!

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

Операции со строками

Срезы:

```
s = "0123456789"  
s1 = s[:8] # "01234567"
```

от начала строки

```
s = "0123456789"  
s1 = s[3:] # "3456789"
```

до конца строки

```
s1 = s[::-1] # "9876543210"
```

реверс строки

Операции со строками

Срезы с отрицательными индексами:

```
s = "0123456789"
```

```
s1 = s[:-2] # "01234567"
```

N-2

```
s = "0123456789"
```

```
s1 = s[-6:-2] # "4567"
```

N-6

N-2

Удаление и вставка символов



Строка – это неизменяемый объект!

Удаление:

```
s = "0123456789"
s1 = s[:3] + s[9:]    # "0129"
           "012"      "9"
```

Вставка:

```
s = "0123456789"
s1 = s[:3] + "ABC" + s[3:]
           "012ABC3456789"
```

Стандартные функции

Верхний/нижний регистр:

```
s = "aAbBcC"  
s1 = s.upper()    # "AABVCC"  
s2 = s.lower()    # "aabbcc"
```

Проверка на цифры:

```
s = "abc"  
print ( s.isdigit() )    # False  
s1 = "123"  
print ( s1.isdigit() )    # True
```

... и много других.

Поиск в строках

```
s = "Здесь был Вася."  
n = s.find ( "с" )      # n = 3  
if n >= 0:  
    print ( "Номер символа", n )  
else:  
    print ( "Символ не найден." )
```



Находит первое слева вхождение подстроки!

Поиск с конца строки:

```
s = "Здесь был Вася."  
n = s.rfind ( "с" )      # n = 12
```


Пример обработки строк

```
print ( "Введите имя, отчество и фамилию:" )
s = input()
n = s.find ( " " )
name = s[:n]      # вырезать имя
s = s[n+1:]
n = s.find ( " " )
name2 = s[:n]      # вырезать отчество
s = s[n+1:]        # осталась фамилия
s = s + " " + name[0] + "." + name2[0] + "."
print ( s )
```

Пример обработки строк

Решение в стиле Python:

```
print ( "Введите имя, отчество и фамилию:" )
s = input()
fio = s.split()
s = fio[2] + " " + fio[0][0] + "." + fio[1][0] + "."
print ( s )
```

Василий

fio[0]

Алибабаевич

fio[1]

Хрюндиков

fio[2]

Преобразования «строка» – «число»

Из строки в число:

```
s = "123"
N = int ( s )          # N = 123
s = "123.456"
X = float ( s )        # X = 123.456
```

Из числа в строку:

```
N = 123
s = str ( N )          # s = "123"
s = "{:5d}".format(N)  # s = "  123"

X = 123.456
s = str ( X )          # s = "123.456"
s = "{:7.2f}".format(X) # s = " 123.46"
s = "{:10.2e}".format(X) # s = " 1.23e+02"
```

Преобразования числа в строку

Из числа в строку:

```
N = 123
s = str(N)           # s = "123"
s = "{:5d}".format(N) # s = "  123"
```

Значение переменной N будет записано по формату d (целое число) в 5 позициях, т.е. в начале строки будут стоять 2 пробела

```
X = 123.456
s = str(X)           # s = "123.456"
s = "{:7.2f}".format(X) # s = " 123.46"
s = "{:10.2e}".format(X) # s = " 1.23e+02"
```

Форматы f (с фиксированной точкой) и e (с плавающей точкой, в экспоненциальной форме)

Задача

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.

Пример:

Введите выражение:

12*3+45

Ответ: 81

Решение

```
s = input("Введите строку:")
```

```
for f in s:
```

```
    if f=="+" or f=="-" or f=="*" or f=="/":
```

```
# f - первый знак
```

```
        n=s.find(f)
```

```
# номер первого знака
```

```
        s1=s[:n]
```

```
# первое число
```

```
        s2=s[n+1:]
```

```
# оставшая часть строки без первого числа и первого знака
```

```
        break
```

```
for z in s2:
```

```
    if z=="+" or z=="-" or z=="*" or z=="/":
```

```
# z - второй знак
```

```
        m=s2.find(z)
```

```
# номер второго знака
```

```
        s3=s2[:m]
```

```
# второе число
```

```
        s4=s2[m+1:]
```

```
# оставшая часть строки (третье число))
```

```
        break
```

```
a=int(s1)
```

```
# преобразовываем первое число
```

```
b=int(s3)
```

```
# преобразовываем второе число
```

```
c=int(s4)
```

```
# преобразовываем третье число
```

Решение (продолжение)

```
if f=="+" :  
    if z=="+":  
        print(a+b+c)  
    elif z=="-":  
        print (a+b-c)  
    elif z=="*":  
        print (a+b*c)  
    elif z=="/":  
        print (a+b//c)  
elif f=="-":  
    if z=="+":  
        print(a-b+c)  
    elif z=="-":  
        print (a-b-c)  
    elif z=="*":  
        print (a-b*c)  
    elif z=="/":  
        print (a-b//c)
```

```
elif f=="*":  
    if z=="+":  
        print(a*b+c)  
    elif z=="-":  
        print (a*b-c)  
    elif z=="*":  
        print (a*b*c)  
    elif z=="/":  
        print (a*b//c)  
elif f=="/":  
    if z=="+":  
        print(a//b+c)  
    elif z=="-":  
        print (a//b-c)  
    elif z=="*":  
        print (a//b*c)  
    elif z=="/":  
        print (a//b//c)
```