Задача 1.

Вводная

**ФИО**

Напишите программу, которая преобразует строку, содержащую имя, отчество и фамилию человека, к форме

<фамилия> <инициалы>

ВХОДНЫЕ ДАННЫЕ

Входная строка содержит имя, отчество и фамилию, разделённые одиночными пробелами.

ВЫХОДНЫЕ ДАННЫЕ

Программа должна вывести в одной строке сначала фамилию, а потом (через пробел) – инициалы.

ПРИМЕРЫ

|  |  |
| --- | --- |
| Donald Ervin Knuth | Knuth D.E. |

Начало формы

Задача 2.

Лёгкая

**Удалить фрагмент**

Дана строка, в которой буква h встречается минимум два раза. Удалите из этой строки первое и последнее вхождение буквы h, а также все символы, находящиеся между ними.

ВХОДНЫЕ ДАННЫЕ

Вводится строка

ВЫХОДНЫЕ ДАННЫЕ

Выведите ответ на задачу

ПРИМЕРЫ

|  |  |
| --- | --- |
| **ВВОД** | **ВЫВОД** |
| In the hole in the ground there lived a hobbit | In tobbit |

Начало формы

Задача 3.

Лёгкая

**Две половинки**

Дана строка. Разрежьте ее на две равные части (если длина строки — четная, а если длина строки нечетная, то длина первой части должна быть на один символ больше). Переставьте эти две части местами, результат запишите в новую строку и выведите на экран.

ВХОДНЫЕ ДАННЫЕ

Вводится строка

ВЫХОДНЫЕ ДАННЫЕ

Выведите ответ на задачу

ПРИМЕРЫ

|  |  |
| --- | --- |
| **ВВОД** | **ВЫВОД** |
| Hi |  iH |

Начало формы

Задача 4

Напишите программу, которая подсчитывает сумму всех цифр в данной строке. Например, в строке **"Sochi-2014"** сумма цифр равна 7.

Задача 5

В строке могут встречаться цифры и иные символы. Числом будем считать последовательность цифр, которая не может быть расширена до большего числа. Например, в строке "10+14=24" встречаются числа 10, 14, 24. Напишите программу, которая подсчитывает сумму всех чисел в данной строке.

ВХОДНЫЕ ДАННЫЕ

Программа получает на вход строку текста, длиной до 100000 символов. Строка может содержать произвольные ASCII-символы, в том числе пробелы.

ВЫХОДНЫЕ ДАННЫЕ

Программа должна вывести единственное число — сумму всех чисел в данной строке.

ПРИМЕР

|  |  |
| --- | --- |
| **ВВОД** | **ВЫВОД** |
| 10+14=24 | 48 |

Задача 6

Есть хорошо известная детская задача: продолжите последовательность

1
11
21
1211
111221
312211

Более точно, последовательность строится так. Первая строка последовательности содержит одну цифру 1. Каждая следующая строка является описанием предыдущей и состоит из четного числа символов, разбитых на пары. Первый символ пары означает количество раз, сколько нужно повторить первый символ пары.

Например, первая строка состоит из одной единицы. Поэтому вторая строка последовательности 11.

Вторая строка состоит из двух единиц. Поэтому третья строка последовательности это 21.

Третья строка состоит из одной двойки (12) и одной единицы (11).

Четвертая строка состоит из одной единицы (11), одной двойки (12) и двух единиц (21).

Более подробно об этой последовательности можно [прочитать в википедии](http://en.wikipedia.org/wiki/Look-and-say_sequence).

Напишите программу, которая строит k-ю строку данной последовательности.

ВХОДНЫЕ ДАННЫЕ

Программа получает на вход одно натуральное число k, не превосходящее 40.

ВЫХОДНЫЕ ДАННЫЕ

Программа должна вывести k-ю строку данной последовательности.

ПРИМЕР

|  |  |
| --- | --- |
| **ВВОД** | **ВЫВОД** |
| 4 | 1211 |

Решение задачи №1

КОД НА ЯЗЫКЕ PYTHON, ПЕРВЫЙ СПОСОБ:

s = input()
left = s.find(' ')
right = s.rfind(' ')
print(s[right + 1:], s[0] + "." + s[left + 1] + ".")

 КОД НА ЯЗЫКЕ PYTHON, ВТОРОЙ СПОСОБ:

a, b, c = input().split()
print(c, a[0] + '.' + b[0] + '.')

Решение задачи №2

s = input()
s = s[:s.find('h')] + s[s.rfind('h') + 1:]
print(s)

Решение задачи №3

Номером элемента строки, до которого надо сделать срез, является наименьшее целое число, которое не меньше чем половина длины строки. Таким образом номер этого элемента (len(s)+1)//2 (целочисленное деление с округлением вверх)

s = input()
print(s[(len(s)+1)//2:] + s[0:(len(s)+1)//2])

Решение задачи №4

S = input()
ans = 0
for char in S:
    if "0" <= char <= "9":
        ans += int(char)
print(ans)

Решение задачи №5

Заведем переменную last\_number, в которой будем хранить число, записанное в конце просмотренной части строки. Просматриваем в цикле все символы строки. Если символ — цифра, то last\_number умножаем на 10 и добавляем эту цифру. Иначе число в конце просмотренной части строки закончилось, поэтому нужно обнулить переменную last\_number, сначала добавив его к ответу.

Чтобы программа работала корректно, когда исходная строка заканчивается цифрой, проще всего добавить в конец строки один символ, не являющийся цифрой.

S = input() + " "
ans = 0
last\_number = 0
for char in S:
    if '0' <= char <= '9':
        last\_number = 10 \* last\_number + int(char)
    else:
        ans += last\_number
        last\_number = 0
print(ans)

Решение задачи №6

Запишем в строку S значение "1". Затем в цикле k−1 раз получаем из строки S новую строку и записываем результат обратно в S. Для получения новой строки просматриваем символы строки S по одному, каждый новый символ сравниваем с предыдущим и считаем количество появления этого символа в конце просмотренной части.

После окончания алгоритма выводим строку S.

k = int(input())
S = "1"
for num in range(k - 1):
    T = ""
    i = 0
    while i < len (S):
        j = 1
        while i + 1 < len(S) and S[i + 1] == S[i]:
            j += 1
            i += 1
        T += str(j) + S[i]
        i += 1
    S = T
print(S)