

Программирование, 9-й класс

Листок 6: Строки.

Почитать про строки можно здесь: справочник, стр. 15 (глава 3).

Во всех задачах запрещается использование констант, обозначающих порядковые номера любых символов в таблице ASCII.

A. Палиндром

Строка называется *палиндромом*, если она читается слева направо и справа налево одинаково.

Программа должна вывести слово YES, если введённое слово — палиндром, и слово NO, если оно не является палиндромом.

Решите эту задачу, используя не более $N//2$ операций сравнения символов и не используя сравнений строк и их срезов.

Input	Output
kazak	YES

B. Количество слов

Напишите программу, которая выводит самое длинное слово переданной её символьной строки. Слово — это последовательность непробельных символов, ограниченная пробелами или концами строки.

Input	Output
Yesterday, all my troubles seemed so far away	8

C. Самое длинное слово

Напишите программу, которая выводит самое длинное слово переданной её символьной строки. Слово — это последовательность непробельных символов, ограниченная пробелами или концами строки.

Input	Output
abra cadabra fibra	cadabra 7

D. Замена регистра - I

Дана строка, содержащая буквы латинского алфавита, знаки препинания и цифры. Напечатать строку, в которой вместо строчных букв исходной строки будут соответствующие прописные и наоборот.

Input	Output
hELLO	Hello

E. Красные и синие - I

Дана цепочка, состоящая из синих (B) и красных (R) точек. Нужно удалить наименьшее количество синих точек так, чтобы сначала шли только синие, а потом — только красные.

В этой задаче запрещается использовать вспомогательные строки, срезы, а также сравнивать строки, а не отдельные символы.

Input	Output
BBBRBRBRR	BBBRRRR 2

F. Слова наоборот

На вход программе подаётся строка, содержащая слова, разделённые пробелами (можно считать, что строка содержит только строчные буквы и пробелы и есть как минимум одно слово). Программа должна напечатать строку, содержащую те же слова в обратном порядке, которые разделены *одним* пробелом (сами слова не меняются, меняется их порядок).

Напечатанная строка не должна начинаться с пробела или заканчиваться им.

Например, дана строка: ' abcd efg h prst '

Программа должна вывести строку: 'prst efg h abcd'

Г. Количество чисел

Дана строка, содержащая произвольные символы. Посчитать количество *натуральных чисел*, записанных в этой строке.

Натуральное число — последовательность цифр, начинающаяся не с нуля и не являющаяся частью другой последовательности, образующей натуральное число.

Например:

для входной строки `abc123 2023 000340004` программа должна вывести число `3`

Комментарий к первому примеру: строки `'0004'`, `'23'`, `'40004'` не являются записью натурального числа в смысле данного выше определения, т.к. содержатся как подстроки в других натуральных числах, соответственно `'340004'`, `'2023'`, `'340004'`.

Input	Output
<code>abc123 2023 000340004</code>	<code>3</code>

Input	Output
<code>2 0000 0 00</code>	<code>1</code>

Н. Шифр Цезаря

Юлий Цезарь использовал свой способ шифрования текста. Каждая буква заменялась на следующую по алфавиту через K позиций по кругу. Необходимо по заданной шифровке определить исходный текст.

Input	Output
<code>XPSE</code>	<code>WORD</code>
<code>1</code>	

И. IP-адрес

IP-адрес это четырёхбайтовый код, который принято записывать в виде четырех десятичных чисел, разделенных точками. Каждое из чисел может принимать значения от 0 до 255. Вот примеры правильных IP-адресов:

`127.0.0.0`

`192.168.0.1`

`255.0.255.255`

Напишите функцию, которая будет возвращать `True`, если переданная строка является правильным IP-адресом, и `False` в противном случае.

Input	Output
<code>127.0.0.1</code>	<code>YES</code>

Ж. Калькулятор

Напишите программу, которая вычисляет арифметическое выражение, введённое в виде символьной строки. Выражение содержит только целые числа и знаки сложения и вычитания.

Функцию `eval` использовать нельзя.

Input	Output
<code>1+12-54+68-17</code>	<code>10</code>

К. Распаковка строки

Будем рассматривать только строчки, состоящие из заглавных латинских букв. Например, рассмотрим строку `AAAABCCCCCDDDD`. Длина этой строки равна 14. Поскольку строка состоит только из латинских букв, повторяющиеся символы могут быть удалены и заменены числами, определяющими количество повторений. Таким образом, данная строка может быть представлена как `4AB5C4D`. Длина такой строки 7. Описанный метод мы назовём упаковкой строки.

Напишите программу, которая берёт упакованную строчку и восстанавливает по ней исходную строку.

Input	Output
<code>AB2C</code>	<code>ABCC</code>

L. *Входит ли одна строка в другую*

Даны две строки: P и T. Найти количество вхождений строки P в строку T, т.е. количество таких индексов строки T, начиная с которых строка P входит в строку T.

Решите эту задачу без использования дополнительных строк и срезов, а также без сравнения срезов (разрешается сравнивать только отдельные символы).

Input	Output
AAABAABVVABAABVVABVABA AAB	3

Input	Output
AAAAA AA	4

M. *Форматирование строки*

Дана текстовая строка, содержащая буквы латинского алфавита, пробелы, запятые и точки. Отформатируйте этот текст по следующим правилам:

- В начале и конце строки не должно быть пробелов.
- Все слова разделяются ровно одним пробелом.
- Точки и запятые пишутся слитно с предыдущим словом, после знака препинания ставится пробел.

Выведите полученную строку.

В приведённом примере одинарные кавычки приведены для того, чтобы были видны пробелы во входной строке и их отсутствие в выходной строке. Программа не должна их выводить.

Input	Output
' Hello , world. '	'Hello, world.'

N. *Из десятичной в римскую*

Напишите программу, которая выводит запись переданного ей десятичного числа в римской системе счисления. Подробнее о правилах перевода смотреть здесь

Input	Output
1234	MCCXXXIV

O. *Палиндромы без учёта регистра*

Дано слово, состоящее только из заглавных и строчных латинских букв. Проверьте, является ли это слово палиндромом, если считать заглавные и строчные буквы не различающимися. Выведите слово YES, если слово является таким палиндромом и слово NO, если не является.

Решение оформите в виде функции `IsPalindrome(S)`, возвращающей значение типа `bool`. При решении этой задачи нельзя пользоваться вспомогательными массивами или строками.

Input	Output
Radar	YES

P. *Палиндромы без учёта пробелов*

Дана строка, состоящая из строчных латинских букв и пробелов. Проверьте, является ли она палиндромом без учета пробелов (например, "аргентина манит негра").

В этой задаче запрещается использовать вспомогательные строки, срезы, а также сравнивать строки, а не отдельные символы.

Input	Output
ab a	YES

Q. *Палиндромы без учёта гласных букв*

Возьмем произвольное слово и сделаем с ним следующую операцию: поменяем местами его первую согласную букву с последней согласной буквой, вторую согласную букву с предпоследней согласной буквой и т.д. Если после этой операции мы вновь получим исходное слово, то будем называть такое слово негласным палиндромом.

Например, слова `sos`, `rare`, `rotor`, `gong`, `karaoke` являются негласными палиндромами.

Вам требуется написать программу, которая по данному слову определяет, является ли оно негласным палиндромом.

Перечень согласных букв английского алфавита вам предлагается составить самостоятельно.

В этой задаче запрещается использовать вспомогательные строки, срезы, а также сравнивать строки, а не отдельные символы.

Input	Output
<code>tennete</code>	<code>YES</code>

R. *Магическая последовательность*

Даны последовательности: 1, 11, 21, 1211, 111221, 312211, 13112221, 1113213211, ...

Выпишите k -ю последовательность.

Input	Output
4	1211

S. *Majority*

Известно, что в строке один из символов встречается чаще остальных вместе взятых. Вывести этот символ, сделав один проход по строке. Разрешается использовать дополнительные переменные, но не строки.

Input	Output
<code>abcabcaaaa</code>	<code>a</code>

T. *Замена регистра - II*

Дана строка, содержащая произвольные символы. Измените регистр символов, являющихся буквами латинского алфавита в этой строке так, чтобы первая буква каждого слова была заглавной, а остальные буквы — строчными.

Слово — это последовательность подряд идущих букв, ограниченная слева и справа либо концом исходной строки, либо символами не буквами.

Input	Output
<code>this is an example</code>	<code>This Is An Example</code>

U. *Красные и синие - II*

Дана цепочка, состоящая из синих (B) и красных (R) точек. Нужно удалить наименьшее количество красных точек так, чтобы сначала шли только синие, а потом — только красные.

В этой задаче запрещается использовать вспомогательные строки, срезы, а также сравнивать строки, а не отдельные символы.

Input	Output
<code>BVBRBRBRBRBBVBRBR</code>	<code>BVVVVVVVVBRBR</code> 5

V. *Красные и синие - III*

Дана цепочка, состоящая из синих (B) и красных (R) точек. Нужно удалить наименьшее одинаковое количество синих и красных точек так, чтобы сначала шли только синие, а потом — только красные.

В этой задаче запрещается использовать вспомогательные строки, срезы, а также сравнивать строки, а не отдельные символы.

Input	Output
<code>BVBRBBRBRBRBRBRBR</code>	<code>BVVBRBRBRBRBR</code> 4

W. *Метод бутерброда*

Секретное агентство «Super-Secret-no» решило для шифрования переписки своих сотрудников использовать «метод бутерброда».

Сначала буквы слова нумеруются в таком порядке: первая буква получает номер 1, последняя буква — номер 2, вторая — номер 3, предпоследняя — номер 4, потом третья ... и так для всех букв. Затем все буквы записываются в шифр в порядке своих номеров. В конец зашифрованного слова добавляется знак «диез» (#), который нельзя использовать в сообщениях.

Например, слово `sandwich` зашифруется в `shacnidw#`.

Вводится слово, зашифрованное методом бутерброда. Длина слова не превышает 20 букв. Выведите расшифрованное слово.

Input	Output
Aabrrbaacda#	Abracadabra

X. Маска файла

Миша готовится к ЕГЭ по информатике. Сейчас он изучает задачу A4, в которой описывается работа с масками файлов.

Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которой также могут встречаться следующие символы.

Символ «?» (вопросительный знак) означает ровно один произвольный символ.

Символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

Напишите программу, которая для каждого имени файла определит, подходит ли оно под заданную маску. Гарантируется, что в маске файла присутствует не более одного символа «*».

Input	Output
?or*.d?? fort.doc	YES

Y. Сделать палиндромом

Дано слово, состоящее только из строчных латинских букв. Определите, какое наименьшее число букв нужно дописать к этому слову справа так, чтобы оно стало палиндромом.

Input	Output
abcd	3

Z. Следующий палиндром

Рассмотрим все натуральные числа, запись которых в десятичной системе счисления является палиндромом (при этом запись не начинается с нуля). Например, числа 121 и 1331 являются палиндромами, а число 123 — нет. По данному числу N найдите N -е в порядке возрастания число-палиндром.

Программа получает на вход одно натуральное число N , не превосходящее 100000.

Программа должна вывести одно натуральное число — N -е в порядке возрастания число-палиндром.

Input	Output
4321	4334