

Задача 1. Калькулятор "Макси"

Ограничение по времени: 1 секунда на тест

Фирма "Макси" собирается производить калькуляторы для целых неотрицательных длинных чисел. Она объявила конкурс на лучший редактор для ввода чисел. Вам предлагается принять участие в этом конкурсе. На клавиатуре калькулятора этой фирмы расположены десять кнопок с цифрами и четыре управляющих клавиши для редактирования: **Home**, **End**, **Left** и **Right**. Курсор на экране калькулятора представляется отрезком вертикальной прямой линии. Считается, что в начальном состоянии экран калькулятора пуст, и курсор находится в самой левой позиции экрана. По команде **Home** курсор переводится в начало строки и устанавливается на позицию перед первым символом. По команде **End** курсор переводится в конец строки и становится после последнего символа. По командам **Left** и **Right** курсор переводится относительно его текущего положения на одну позицию влево или вправо соответственно. Калькулятор работает в режиме вставки символов. Любая цифра вставляется перед текущей позицией курсора. Если курсор стоит в самой левой позиции экрана, то команда **Left** игнорируется. Аналогично, если он находится после последней цифры текущего состояния редактируемого числа, то команда **Right** игнорируется. Если символов на экране нет, то команды **Home** и **End** игнорируются.

Вам необходимо написать программу, которая обрабатывает последовательность команд и цифр, введенных с клавиатуры калькулятора, и выводит получившееся число на его экран. Командам **Home**, **End**, **Left** и **Right** во входной последовательности соответствуют символы 'H', 'E', 'L' и 'R'.

Входные данные

Во входном файле записана строка текста, состоящая из цифр и символов 'H', 'E', 'L' и 'R'. Длина строки не превосходит 10^6 символов. Гарантируется, что хотя бы одна цифра во входной строке присутствует.

Выходные данные

В выходной файл необходимо вывести строку текста, содержащую число — результат выполнения команд. Незначащие нули в полученном числе нужно удалить.

Пример

<i>input.txt</i>	<i>output.txt</i>
0RR12EL97	1972

Задача 2. Скобки

Ограничение по времени: 1 секунда на тест

Дана строка, состоящая из открывающих и закрывающих круглых скобок. Необходимо посчитать, сколько подстрок данной строки являются правильными скобочными последовательностями.

Правильной скобочной последовательностью называется последовательность, в которой количество открывающих скобок равно количеству закрывающих, и в любом начальном отрезке последовательности открывающих скобок не меньше, чем закрывающих.

Входные данные

В выходной файл записана строка текста, состоящая из символов "(" и ")". Длина строки больше нуля и не превосходит 10^5 символов.

Выходные данные

В выходной файл необходимо выдать целое число, равное искомому количеству подстрок, являющихся правильными скобочными последовательностями.

Пример

<i>input.txt</i>	<i>output.txt</i>
(())	4

Задача 3. Слияние или поглощение

1 секунда на тест

В некоторой стране X каждому банку один раз в год по взаимному согласию с целью увеличения капитала разрешается объединиться с одним другим банком. Существует два способа их объединения. Первый способ объединения банков это слияние. Слияние разрешается в том случае, когда различие между капиталами банков не превосходит 1000000. Если же разница между капиталами банков больше, то разрешается их объединение вторым способом — поглощением. Вновь созданные банки в текущем году не имеют права участвовать в процессе слияния или поглощения.

По заданному реестру капиталов банков государства X Вам нужно определить, сколько различных вариантов слияний и поглощений можно провести между банками в этом году.

Входные данные

В первой строке входного файла записано целое число N — количество банков ($1 < N \leq 10^6$). В следующих N строках, по одному на строке, находятся целые числа K_i — капиталы банков, записанные в невозрастающем порядке ($0 < K_i \leq 10^9, 1 \leq i \leq N$).

Выходные данные

В выходной файл нужно вывести через пробел два целых числа в одной строке. Первое число — это количество возможных слияний банков, а второе — количество возможных поглощений.

Пример

<i>input.txt</i>	<i>output.txt</i>
5 5123000 4509990 3515543 2356100 2134588	3 7

Задача 4. Волк и коза

Ограничение по времени: 1 секунда на тест

Вспомним известную задачу о мужике, волке, козе и капусте, который оказался на острове посреди реки. Перед ним стояла задача так перевезти всех на берег, чтобы коза не съела капусту, а волк не съел козу. Перед вами стоит другая задача. Мужик хочет первой вывезти с острова капусту. А для того, чтобы волк не съел козу, он решил животных друг от друга отгородить. Помогите ему построить ограду так, чтобы количество возведенных препятствий было минимально.

В вашем распоряжении имеется карта острова, имеющего форму прямоугольника. Она расчерчена на клетки одинакового размера. На ней отмечены естественные препятствия и места расположения животных.

Входные данные

В первой строке входного файла через пробел записаны два натуральных числа N и M — количество строк и столбцов на карте, соответственно ($4 < N, M \leq 500$). В следующих N строках описана карта. Каждая строка — это последовательность из M символов.

Клетки карты, являющиеся естественными препятствиями, например, вода или валуны, обозначены символом '#'. Клетки, по которым могут перемещаться волк и коза, обозначены символом '.'. Местоположение волка отмечено символом 'V', а положение козы символом 'K'. Считается, что животные могут переходить с одного места острова на другое, если только соответствующие этим местам клетки на карте имеют общие стороны.

Выходные данные

В выходной файл необходимо вывести одно целое число — наименьшее количество препятствий, которое надо возвести, чтобы волк не съел козу. Если это сделать невозможно, то выведите число -1.

Пример

<i>input.txt</i>	<i>output.txt</i>
5 9 ###.##### #..#....# #.V...K.# #....#..######	1