

Задача А. Хитрый лис

Имя входного файла: `gcd.in`
Имя выходного файла: `gcd.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Ник Уайлд хочет проверить очередную аферу. В Зверополисе существует давняя забава — «Математическая зверолотерея».

Правила предельно просты: игрок вытаскивает из огромной колоды наудачу n карт. Значения карт могут повторяться. Также для каждой новой игры берут новую колоду, состав которой не знает никто. Выигрыш игрока подсчитывается как наибольший общий делитель вытащенных карт.

Разумеется, Ник хочет выиграть как можно больше. Для этого он может незаметно убрать одну карту.

Ник просит вас помочь ему с этой задачей.

Формат входного файла

В первой строке входного файла находится одно натуральное число n ($2 \leq n \leq 10^5$) — количество карт, которые вытянул Ник. Во второй строке находится n натуральных чисел a_i ($1 \leq a_i \leq 10^9$) — значения вытащенных карт.

Формат выходного файла

В выходной файл выведите два числа, разделенных пробелом — номер карты, которую Нику нужно убрать и выигрыш, который Ник после этого получит.

Если правильных ответов несколько, выведите любой.

Пример

<code>gcd.in</code>	<code>gcd.out</code>
3 6 12 8	3 6
5 2 2 3 4 6	3 2

Система оценивания

Первая группа состоит из тестов, для которых выполняется ограничение $n \leq 1000$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 47 балла.

Вторая группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 53 балла.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку «Request feedback» на вкладке «Runs».

Задача В. Прыжки по камням

Имя входного файла: `stones.in`
Имя выходного файла: `stones.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Джуди Хопс бежит на очень важную встречу с Ником, и тут как назло перед ней оказалось огромное озеро. Обходить его времени нет, требуется какой-то другой план. К счастью для Джуди, на поверхности озера виднеются несколько камней, по которым она сможет прыгать.

Для простоты представим озеро в виде координатной прямой, на которой Джуди стоит в точке 0, а конец озера находится в точке с координатой n . Несмотря на то, что Джуди — зайчиха и хороша в прыжках, прыгать она может только на 1 и 2 метра вперед соответственно (то есть из точки с координатой x она может попасть в точки с координатами $x + 1$ и $x + 2$). Зная координаты камней, Джуди хочет найти минимальное количество прыжков, которое ей нужно совершить, чтобы добраться из точки с координатой 0 в точку с координатой n . Однако ее также с детства учили делать все максимально оптимально, поэтому она также хочет из всех путей с минимальным количеством прыжком выбрать лексикографически минимальный. Каждый прыжок описывается последовательностью чисел 1 и 2 (i -е число соответствует длине i -го прыжка).

Помогите Джуди как можно быстрее решить эту задачу — хоть она и дама, очень сильно опаздывать на встречу все-таки некрасиво.

Формат входного файла

В первой строке входного файла находятся два числа n, k ($1 \leq n \leq 10^5; 0 \leq k < n$) — координата, в которую Джуди надо попасть, и количество камней на озере соответственно.

В следующей строке через пробел находятся k чисел x_i ($1 \leq x_i \leq n$) — координаты камней. Гарантируется, что для всех $x_1 > 0$, а также для всех $i > 1$ верно, что $x_i > x_{i-1}$.

Формат выходного файла

В первой строке выходного файла выведите целое число — минимальное количество прыжков, которое требуется, чтобы добраться из координаты 0 в координату n . Во второй строке выведите оптимальный путь — последовательность 1 и 2 без пробелов.

Путь должен быть минимальным по длине, а из всех таких, лексикографически минимальным.

Если Джуди не удастся никак допрыгнуть до точки с координатой n , то есть придется обходить озеро, в этом случае в единственной строке выходного файла выведите «-1» (без кавычек).

Примеры

<code>stones.in</code>	<code>stones.out</code>
5 3 2 3 4	3 212
7 3 2 3 4	-1

Комментарий

Последовательность чисел A является лексикографически меньше последовательности B , если $A \neq B$ и существует такое $0 \leq k \leq |A|$, что $A_0 = B_0, A_1 = B_1, \dots, A_k = B_k$ и $A_{k+1} < B_{k+1}$.

Система оценивания

Первая группа состоит из тестов, для которых выполняются ограничения $1 \leq n \leq 1000$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 34 балла.

Вторая группа состоит из тестов, для которых выполняется ограничение, что ответ всегда существует. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 35 баллов.

Третья группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 31 балл.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку «Request feedback» на вкладке «Runs».

Задача С. Ленивцы и забор

Имя входного файла: `fence.in`
Имя выходного файла: `fence.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Ленивцы во главе с Блицем решили покрасить забор. Забор состоит из n досок, каждая из которых изначально непокрашена.

Они собираются красить забор по следующему алгоритму: у некоторых досок встанет по одному ленивцу с краской, эти доски сразу будут покрашены, затем Блиц подаст команду сдвинуться на одну доску вправо, после выполнения этой команды каждый ленивец покрасит ту доску, напротив которой он оказался. Эта доска может быть уже покрашена, тогда ленивец напротив этой доски просто отдыхает. Всего Блиц подает несколько таких команд, возможно ноль. Также известно, что ленивцы изначально располагаются так, чтобы в результате выполнения команд не оказаться за забором, то есть перед выполнением команды никакой ленивец не должен оказаться напротив доски с номером n .

Блиц хочет покрасить забор определенным образом. То есть для каждой доски он знает, покрашенной она должна быть или нет после выполнения всех команд.

Пока ленивцы только собираются прийти, чтобы осуществить задумку Блица, поэтому он не знает точно, сколько их придет. Поэтому он просит вас для всех i от 1 до n посчитать, какого наименьшего количества команд можно достичь, если в покраске забора будут участвовать i ленивцев.

Формат входного файла

В первой строке содержится натуральное число n ($1 \leq n \leq 10^6$) — количество досок в заборе.

В следующей строке содержится строка, состоящая из n символов, каждый из которых либо `'.'` — это означает, что доска должна остаться непокрашенной, либо `'#'` — доска должна быть покрашенной после выполнения всех команд.

Формат выходного файла

В единственной строке выведите n чисел a_i — наименьшее количество команд, которое придется подать Блицу, если в покраске забора будут участвовать i ленивцев, либо -1 , если при таком количестве ленивцев никаким образом невозможно достичь требуемой раскраски.

Пример

<code>fence.in</code>	<code>fence.out</code>
7 .#####.	4 2 1 1 0 -1 -1
6 .###.#	-1 -1 -1 0 -1 -1
10 ..###.###	-1 2 -1 1 -1 0 -1 -1 -1 -1

Система оценивания

Первая группа состоит из тестов, для которых выполняется ограничение $n \leq 1000$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 49 баллов.

Вторая группа состоит из тестов, для которых выполняется ограничение $n \leq 10^6$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 51 баллов.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку `Request feedback` на вкладке `Runs`.

Задача D. Треугольное число многоугольника

Имя входного файла: `triangles.in`
Имя выходного файла: `triangles.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Главная причина, по которой Блиц настолько быстр — постоянная работа над собой и многочисленные тренировки. В эту субботу Блиц тренируется решать на скорость математические ребусы.

В одном из них ленивцу дается правильный n -угольник, вершины которого покрашены в белый или черный цвет. Треугольным числом многоугольника называется количество способов выбрать три вершины одного цвета в этом многоугольнике так, чтобы треугольник, построенный на этих трех вершинах, был равнобедренным. Решить ребус для одного многоугольника Блицу показалось слишком просто, поэтому он решил развлечь себя. Блиц последовательно изменил цвет одной из вершин многоугольника на противоположный m раз и пересчитал треугольное число многоугольника.

Поскольку для успешного завершения тренировки нужно быть уверенным, что ты не допустил ошибок, Блиц попытался найти кого-нибудь, кто сможет проверить его вычисления. К сожалению, никто в Зверополисе не обладает нужными навыками, поэтому Блиц обратился к вам за помощью в проверке своих результатов.

Формат входного файла

В первой строке даны два целых числа n, m ($3 \leq n \leq 200\,000$, $1 \leq m \leq 200\,000$) — количество вершин в многоугольнике и количество запросов соответственно.

Во второй строке дана строка длины n , состоящая из символов «W» и «B», i -й символ которой равен «W», если i -я вершина изначально имеет белый цвет, и «B», если i -я вершина изначально имеет черный цвет.

В следующих m строках дано по одному целому числу id ($1 \leq id \leq n$) — запрос смены цвета на противоположный у id -й вершины.

Формат выходного файла

После каждого запроса выведите одно целое число — количество способов выбрать три вершины одного цвета в многоугольнике так, что они образуют равнобедренный треугольник.

Пример

<code>triangles.in</code>	<code>triangles.out</code>
3 4	0
WWW	0
1	1
2	0
3	
2	
<code>triangles.in</code>	<code>triangles.out</code>
6 6	4
BBBBBB	2
1	2
3	2
5	4
2	8
4	
6	

Система оценивания

Первая группа состоит из тестов, для которых выполняется ограничение $3 \leq n \leq 200$, $1 \leq m \leq 200$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 25 баллов.

Вторая группа состоит из тестов, для которых выполняется ограничение $3 \leq n \leq 1000$, $1 \leq m \leq 1000$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 25 баллов.

Третья группа состоит из тестов, для которых выполняется ограничение $3 \leq n \leq 200\,000$, $m = 1$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 25 баллов.

Четвертая группа состоит из тестов, для которых выполняется ограничение $3 \leq n \leq 200\,000$, $1 \leq m \leq 200\,000$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 25 баллов.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку «Request feedback» на вкладке «Runs».