

Задача А. Минимальный разрез

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Мощностью массива $[b_1, \dots, b_m]$ назовём величину $M([b_1, \dots, b_m]) = b_1 \cdot 1 + b_2 \cdot 2 + \dots + b_m \cdot m$.

Вам дан массив $[a_1, \dots, a_n]$, стоимость разреза между индексами i и $i + 1$ равна c_i . Найдите такое разбиение массива на части, чтобы минимизировать сумму мощностей частей и стоимостей разрезов.

Формально, найдите такие целые индексы $1 \leq i_1 < \dots < i_k < n$, что

$$M([a_1, \dots, a_{i_1}]) + c_{i_1} + M([a_{i_1+1}, \dots, a_{i_2}]) + c_{i_2} + \dots + c_{i_k} + M([a_{i_k+1}, \dots, a_n])$$

минимально.

Формат входных данных

В первой строке записано единственное целое число n ($2 \leq n \leq 10^6$) — длина данного массива.

Во второй строке записано n целых чисел a_1, \dots, a_n — сам массив ($0 \leq a_i \leq 10^6$).

В третьей строке записаны $n - 1$ целое число c_1, \dots, c_{n-1} — стоимости разрезов ($0 \leq c_i \leq 10^{12}$).

Формат выходных данных

Выведите одно целое число — минимальную сумму мощностей частей и стоимостей разрезов.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 9 8	14
4 1 2 3 5 3 2 7	20

Замечание

В первом примере массив выгодно вообще не разбивать.

Во втором примере выгодно массив разбить на две части по два элемента. Тогда $M([1, 2]) + 2 + M([3, 5]) = 5 + 2 + 13 = 20$.

Задача В. Задача без легенды

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан неубывающий массив a размера n , а также стоимостей элементов c . Вам нужно выбрать подпоследовательность (возможно, пустую) с минимальной стоимостью и вывести эту стоимость. Стоимость подпоследовательности из элементов i_1, i_2, \dots, i_k это сумма стоимостей ее элементов плюс сумма «расстояний» между ее соседними элементами. «Расстояние» между элементами i и j , ($i < j$) — это $(7a_j - 4a_i)(5a_j - 2a_i)(4a_j - a_i)$.

Формат входных данных

В первой строке вводится натуральное число n ($1 \leq n \leq 10^6$). Во второй строке вводится массив целых чисел a ($0 \leq a_i \leq 10^6$). В третьей строке вводится массив целых чисел c ($-10^{18} \leq c_i \leq 10^{18}$)

Формат выходных данных

Выведите одно число — минимальную стоимость подпоследовательности.

Примеры

стандартный ввод	стандартный вывод
2 0 1 2 -6	-6
3 1 2 4 -6 5 1	-6
5 1 2 2 2 4 0 -2 -1 -5 2	-5
4 1 1 3 5 7 -10 8 -7	-10
5 0 1 4 4 4 -8 2 10 4 5	-8
2 1 1 -100 -100	-173

Замечание

Так как ответ может быть очень большим, рекомендуется использовать `__int128` или еще что-нибудь.

Задача С. Настойки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В линию выстроены n настоек, причем настойка 1 находится слева, а настойка n — справа. Каждая настойка увеличит ваше здоровье на a_i , если ее выпить. a_i может быть отрицательным, что означает, что настойка уменьшит ваше здоровье.

Вы начинаете с 0 здоровья и будете идти слева направо, от первой настойки до последней. Для каждой настойки вы можете выбрать, выпить ли ее. Вы должны следить за тем, чтобы ваше здоровье всегда было неотрицательным.

Какое наибольшее количество настоек вы можете выпить?

Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 200\,000$) — количество настоек.

Следующая строка содержит n целых чисел a_1, a_2, \dots, a_n ($-10^9 \leq a_i \leq 10^9$), которые обозначают изменения здоровья после употребления данных настоек.

Формат выходных данных

Выведите одно целое число — максимальное количество настоек, которое вы можете выпить, чтобы ваше здоровье всегда было неотрицательным.

Примеры

стандартный ввод	стандартный вывод
6 4 -4 1 -3 1 -3	5
5 1 3 -4 2 1	5
12 -3 -3 -7 -7 -1 -7 3 3 -2 -1 0 -7	5

Замечание

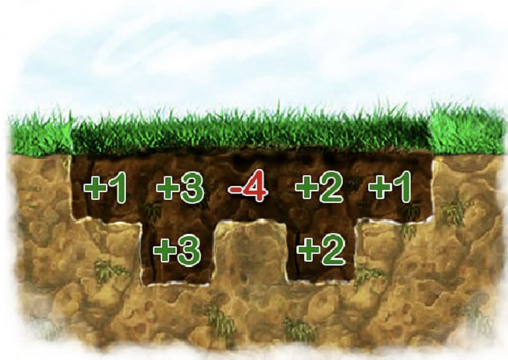
В примере, вы можете выпить 5 настоек, приняв настойки 1, 3, 4, 5 и 6. Невозможно выпить все 6 настоек, потому что в какой-то момент ваше здоровье станет отрицательным.

Задача D. Археологи

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Ваша команда охотников за сокровищами нашла огромные залежи сокровищ! Это можно представить как n мест для раскопок на прямой. Каждое из этих мест имеет свою доходность p_i , то есть за выкапывание каждого метра в глубину в этом месте ваша команда получает p_i юаней.

Конечно, вы хотите получить как можно больше юаней. Но чтобы не создавать оползней, после ваших раскопок не должно получиться слишком отвесных склонов. Формально, для любых двух соседних мест раскопок их глубина должна отличаться не больше чем на 1. В частности, глубина в 1-м и n -м местах должна быть 0 или 1. Какую максимальную прибыль можно получить?



Формат входных данных

Первая строка содержит целое положительное n ($1 \leq n \leq 250\,000$). Вторая строка содержит n целых чисел p_i ($-10^6 \leq p_i \leq 10^6$)

Формат выходных данных

Выведите одно число — максимальную прибыль, которую можно получить.

Примеры

стандартный ввод	стандартный вывод
5 1 3 -4 2 1	8
4 1 1 -2 3	5
5 -1 -3 0 -5 -4	0

Задача E. Фейерверк

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Фейерверк — это одна из самых захватывающих частей фестиваля. Для фейерверка очень важно, чтобы все взрывные устройства, соединенные набором фитилей с точкой поджога, взорвались одновременно в нужный момент. Так как взрывные устройства, используемые для фейерверка, очень опасны, они установлены на некотором расстоянии от точки поджога, и соединены с ней фитилями. Чтобы соединить несколько взрывных устройств с точкой поджога, фитили соединяются наподобие ребёр дерева, как показано на Рисунке 1. Искра появляется на точке поджога и распространяется по фитилям. Когда искра достигает соединения нескольких фитилей, она начинает распространяться по всем ним. Скорость искры постоянна. Рисунок 1 показывает, как 6 взрывных устройств $\{E_1, E_2, \dots, E_6\}$ могут быть соединены, а также какой длины должен быть каждый использованный для этого фитиль. Кроме того, показано время взрыва каждого устройства, если считать, что искра в точке поджога появилась в момент времени 0

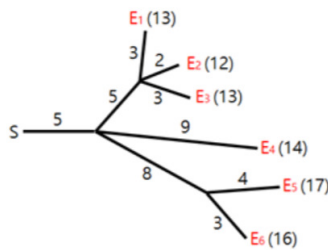


Рис. 1: Схема соединения

Хьюнмин должен подготовить схему соединения. К сожалению, в его схеме взрывные устройства могут взорваться не в одно и тоже время. Необходимо изменить длины некоторых фитилей, так чтобы все устройства взорвались в одно и тоже время. Например, чтобы все взрывные устройства на рисунке взорвались в момент времени 13 длины фитилей надо изменить как показано слева на Рисунке 2. Чтобы все взрывные устройства на Рисунке 1 взорвались в момент времени 14 длины фитилей надо изменить как показано слева на Рисунке 2.

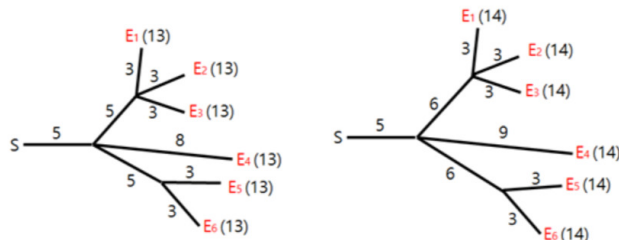


Рис. 2: Примеры изменения длин, после которых взрывы происходят одновременно

Стоимость изменения длины фитиля равна абсолютному значению разности длин до и после изменения. Например, стоимость изменения схемы на Рисунке 1 до схемы слева на Рисунке 2 равна 6. Стоимость изменения схемы на Рисунке 1 до схемы слева на Рисунке 2 равна 5.

Длину провода можно уменьшить до 0.

По заданной схеме соединения, необходимо изменить длины некоторых фитилей, так, чтобы все взрывные устройства взорвались одновременно, с минимальной стоимостью.

Формат входных данных

Все числа во входных данных целые положительные. Пусть N обозначает количество точек соединения фитилей, M — количество взрывных устройств ($1 \leq N+M \leq 300\,000$). Точки соединения фитилей пронумерованы от 1 до N . Точка поджога находится в соединении номер 1. Взрывные устройства пронумерованы от $N+1$ до $N+M$.

Входные данные даны в следующем формате:

$N\ M$

$P_2\ C_2$

$P_3\ C_3$

...

$P_N\ C_N$

$P_{N+1}\ C_{N+1}$

...

$P_{N+M}\ C_{N+M}$

$P_i, 1 \leq P_i < i$, обозначает точку соединения фитилей, к которой проведен фитиль от точки соединения или взрывного устройства номер i . C_i обозначает длину фитиля, который их соединяет ($1 \leq C_i \leq 10^9$). В каждой точке соединения, кроме той, в которой расположена точка поджога, сходятся хотя бы 2 фитиля. С каждым взрывным устройством соединен ровно 1 фитиль.

Формат выходных данных

Выведите минимальную стоимость изменения длин фитилей, после которого все взрывные устройства взорвутся одновременно.

Примеры

стандартный ввод	стандартный вывод
4 6 1 5 2 5 2 8 3 3 3 2 3 3 2 9 4 4 4 3	5

Задача F. Приключение

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Теплым весенним днем группа из N школьников-программистов гуляла в окрестностях города Кисловодска. К несчастью, они набрали на большую и довольно глубокую яму. Как это случилось — непонятно, но вся компания оказалась в этой яме.

Глубина ямы равна H . Каждый школьник знает свой рост по плечи h_i и длину своих рук l_i . Таким образом, если он, стоя на дне ямы, поднимет руки, то его ладони окажутся на высоте $h_i + l_i$ от уровня дна ямы. Школьники могут, вставая друг другу на плечи, образовывать вертикальную колонну. При этом любой школьник может встать на плечи любого другого школьника. Если под школьником i стоят школьники j_1, j_2, \dots, j_k , то он может дотянуться до уровня $h_{j_1} + h_{j_2} + \dots + h_{j_k} + h_i + l_i$.

Если школьник может дотянуться до края ямы (то есть $h_{j_1} + h_{j_2} + \dots + h_{j_k} + h_i + l_i \geq H$), то он может выбраться из нее. Выбравшиеся из ямы школьники не могут помочь оставшимся.

Найдите наибольшее количество школьников, которые смогут выбраться из ямы до прибытия помощи, и перечислите их номера.

Формат входных данных

В первой строке входного файла записано натуральное число N ($1 \leq N \leq 10^5$) — количество школьников, попавших в яму.

Далее в N строках указаны по два целых числа: рост i -го школьника по плечи h_i и длина его рук l_i ($1 \leq h_i, l_i \leq 10^9$).

В последней строке указано целое число — глубина ямы H ($1 \leq H \leq 10^9$).

Формат выходных данных

В первой строке выведите K — максимальное количество школьников, которые смогут выбраться из ямы. Если $K > 0$, то во второй строке выведите их номера в том порядке, в котором они вылезают из ямы. Школьники нумеруются с единицы в том порядке, в котором они заданы во входном файле. Если существует несколько решений, выведите любое.

Примеры

стандартный ввод	стандартный вывод
2 10 4 5 2 20	0
6 6 7 3 1 8 5 8 5 4 2 10 5 30	4 2 5 3 1

Задача G. Бэкапы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вы управляете IT-компанией, которая создает бэкапы компьютерных данных для больших офисов. Делать бэкапы скучно, и поэтому вы разрабатываете свою систему таким образом, чтобы разные офисы могли создавать бэкапы данных друг друга, пока вы сидите дома и играете в компьютерные игры.

Все офисы ваших клиентов расположены вдоль одной улицы. Вы решили объединить офисы в пары, и для каждой пары офисов вы проложите сетевой кабель между двумя зданиями, чтобы они могли создавать бэкапы данных друг друга.

Однако сетевые кабели стоят дорого. Ваша местная телекоммуникационная компания предоставит вам только k сетевых кабелей, что означает, что вы можете организовать резервное копирование только для k пар офисов, (в общей сложности $2k$ офисов). Ни один офис не может принадлежать более чем одной паре (то есть все эти $2k$ офисов должны быть разными).

Кроме того, телекоммуникационная компания взимает плату за километр. Это означает, что вам нужно выбрать эти k пар офисов, чтобы использовать как можно меньше кабелей. Другими словами, вам нужно выбрать пары таким образом, чтобы при сложении расстояний между двумя офисами в каждой паре общее расстояние было как можно меньше.

Например, предположим, что у вас было пять клиентов, офисы которых расположены как показано ниже: в 1 км, 3 км, 4 км, 6 км и 12 км от начала улицы. Телекоммуникационная компания предоставит вам только $k = 2$ кабелей.

Лучше всего соединить первый и второй офис и третий и четвертый офис. Таким образом вы используете все $k = 2$ кабелей и длина первого кабеля будет $3 - 1 = 2$ км, и длина второго кабеля будет $6 - 4 = 2$ км. Таким образом, суммарная длина кабелей будет 4км. Можно показать, что меньшую длину получить нельзя.

Формат входных данных

Первая строка содержит два числа: n и k , обозначающих количество офисов на улице ($2 \leq n \leq 100\,000$) и количество доступных вам кабелей ($1 \leq k \leq \frac{n}{2}$).

Следующих n строк содержат по одному целому числу: ($0 \leq s \leq 1\,000\,000\,000$), i -ое число означает расстояние от i -ого офиса до начала улицы. Офисы отсортированы по расстоянию до начала улицы, от ближайшего до наиболее далёкого. Не существует двух офисов, которые находятся в одном месте.

Формат выходных данных

Вы должны вывести ровно одно целое положительное число - минимальную длину кабелей, необходимую чтобы соединить $2k$ офисов при помощи k кабелей.

Примеры

стандартный ввод	стандартный вывод
5 2 1 3 4 6 12	4

Замечание

Пример входных данных соответствует примеру разобранному в условии.

Задача Н. Бэкапы с восстановлением

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Вы управляете IT-компанией, которая создает, бэкапы компьютерных данных для больших офисов. Делать бэкапы скучно, и поэтому вы разрабатываете свою систему таким образом, чтобы разные офисы могли создавать бэкапы данных друг друга, пока вы сидите дома и играете в компьютерные игры.

Все офисы ваших клиентов расположены вдоль одной улицы. Вы решили объединить офисы в пары, и для каждой пары офисов вы проложите сетевой кабель между двумя зданиями, чтобы они могли создавать бэкапы данных друг друга.

Однако сетевые кабели стоят дорого. Ваша местная телекоммуникационная компания предоставит вам только k сетевых кабелей, что означает, что вы можете организовать резервное копирование только для k пар офисов, (в общей сложности $2k$ офисов). Ни один офис не может принадлежать более чем одной паре (то есть все эти $2k$ офисов должны быть разными).

Кроме того, телекоммуникационная компания взимает плату за километр. Это означает, что вам нужно выбрать эти k пар офисов, чтобы использовать как можно меньше кабелей. Другими словами, вам нужно выбрать пары таким образом, чтобы при сложении расстояний между двумя офисами в каждой паре общее расстояние было как можно меньше.

Например, предположим, что у вас было пять клиентов, офисы которых расположены как показано ниже: в 1 км, 3 км, 4 км, 6 км и 12 км от начала улицы. Телекоммуникационная компания предоставит вам только $k = 2$ кабелей.

Лучше всего соединить первый и второй офис и третий и четвертый офис. Таким образом вы используете все $k = 2$ кабелей и длина первого кабеля будет $3 - 1 = 2$ км, и длина второго кабеля будет $6 - 4 = 2$ км. Таким образом, суммарная длина кабелей будет 4км. Можно показать, что меньшую длину получить нельзя.

Формат входных данных

Первая строка содержит два числа: n и k , обозначающих количество офисов на улице ($2 \leq n \leq 100\,000$) и количество доступных вам кабелей ($1 \leq k \leq \frac{n}{2}$).

Следующих n строк содержат по одному целому числу: ($0 \leq s \leq 1\,000\,000\,000$), i -ое число означает расстояние от i -ого офиса до начала улицы. Офисы отсортированы по расстоянию до начала улицы, от ближайшего до наиболее далёкого. Не существует двух офисов, которые находятся в одном месте.

Формат выходных данных

Вы должны сначала вывести одно целое положительное число - минимальную длину кабелей, необходимую чтобы соединить $2k$ офисов при помощи k кабелей.

Затем нужно вывести k строк, в каждой по два числа от 1 до N — номера домов, которые нужно соединить кабелем. Вы можете выводить пары в любом порядке.

Примеры

стандартный ввод	стандартный вывод
5 2	4
1	3 4
3	1 2
4	
6	
12	

Замечание

Пример входных данных соответствует примеру разобранным в условии.

Задача I. Честный дележ

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

У вас есть массив неотрицательных целых чисел a_1, a_2, \dots, a_n .

Вам нужно разделить его на k непустых подотрезков: $[1; b_1], [b_1 + 1; b_2], \dots, [b_{k-1} + 1; n]$.

Обозначим сумму на i -м отрезке как s_i и максимум на i -м отрезке как m_i . Ваша задача сделать $|s_i - s_{i+1}| \leq \max(m_i, m_{i+1})$ для всех $1 \leq i \leq k - 1$.

Формат входных данных

В первой строке находятся два целых числа n и k : размер массива и необходимое количество отрезков ($3 \leq k \leq n \leq 100\,000$).

В следующей строке находится n целых чисел a_1, a_2, \dots, a_n : данный массив ($0 \leq a_i \leq 50\,000$).

Формат выходных данных

Если разделить массив указанным образом возможно, выведите “Yes” на первой строке и $k - 1$ целых чисел b_1, b_2, \dots, b_{k-1} , разделенных пробелами на второй строке.

Числа должны удовлетворять $1 \leq b_1 < b_2 < \dots < b_{k-1} < n$.

Также неравенства $|s_i - s_{i+1}| \leq \max(m_i, m_{i+1})$ должны быть выполнены для всех $1 \leq i \leq k - 1$.

Если существует несколько возможных решений, выведите любое.

Если разделение невозможно, выведите “No” в единственной строке.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
5 3	Yes
17 18 17 30 35	2 4

Задача J. Электрик

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

В деревне J есть n домов с доступом в интернет. В одном из домов живёт злодей, который занимается своими злодейскими делами: постоянно пишет нам гневные комментарии в соцсетях в фейкового аккаунта. Мы не знаем, в каком именно доме он живёт, но хотим это выяснить.

На электроподстанции есть n выключателей, по одному для каждого дома. Когда выключатель включён, в доме горит свет, а когда выключен — света нет, а значит, нет и доступа в интернет. Изначально все выключатели включены. В начале каждого часа мы можем изменить состояние выключателей любым желаемым образом: какие-то включить, какие-то выключить. После этого в течение следующего часа мы мониторим соцсети: если в доме злодея есть свет, комментарии обязательно появятся, а если света нет — комментариев не будет. После нашего расследования мы должны вернуть всё в исходное состояние: снова включить свет во всех домах.

К сожалению, если свет в доме злодея выключается более k раз, он может заподозрить неладное, и мы хотим этого избежать. Важно отметить, что свет может безопасно отсутствовать долгое время, но важно именно количество раз, когда мы его выключаем. Какое минимальное количество часов потребуется, чтобы мы могли уверенно определить, в каком доме живёт злодей, и восстановить свет во всех домах?

Формат входных данных

Первая строка содержит целое число t : количество тестовых случаев ($1 \leq t \leq 2 \cdot 10^5$). Далее следуют t тестовых случаев.

Каждый тестовый случай задаётся на отдельной строке и содержит два целых числа n и k : количество домов и максимальное допустимое количество отключений света в доме злодея ($1 \leq n, k \leq 10^{18}$).

Формат выходных данных

Для каждого тестового случая выведите строку с одним целым числом: минимальное количество часов, необходимое для определения дома злодея и восстановления света во всех домах.

Примеры

стандартный ввод	стандартный вывод
5	2
3 1	2
4 1	1
2 2	5
17 3	7
73 6	
1	0
1 1	

Замечание

В первом тестовом случае $n = 3$ и $k = 1$. Мы можем начать с того, что в первый час выключим свет в домах с номерами 1 и 2. Затем, спустя час, мы можем восстановить свет в доме номер 1, и ещё через час — восстановить свет в доме номер 2. К началу третьего часа мы точно будем знать, где живёт злодей, и мы восстановим свет во всех домах.

Задача К. Австралийская ПСП

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Как известно, в Австралии смотрят на вещи под самыми разными углами, поэтому и правильные скобочные последовательности они задают нестандартным способом:

- Пустая скобочная последовательность считается правильной.
- Если S считается правильной, то $)S($, (S) , $[S]$, $]S[$, $\{S\}$, $\}S\{$, $\langle S \rangle$ и $\rangle S \langle$ тоже считаются правильными.
- Если S и T считаются правильными, то $S + T$ тоже считается правильной (здесь $+$ означает конкатенацию строк).

Мальчик Вася решил посетить Австралию. Но вот беда, для этого надо пройти австралийский тест на интеллект! В самом сложном задании теста даётся строка s , состоящая из скобок и к ней даются m заданий двух видов:

1. Заменить скобку на позиции a_i .
2. Сказать, считается ли подстрока s на позициях с l_i по позицию r_i включительно правильной скобочной последовательностью в Австралии.

Вася очень просит вас помочь ему пройти тест.

Формат входных данных

Первая строка содержит единственное целое число n ($1 \leq n \leq 200\,000$) — длину скобочной последовательности.

Во второй строке содержится строка s длины n , состоящая из скобок $() [] \{ \} \langle \rangle$ — исходная строка, данная Васе.

В третьей строке содержится целое число m ($1 \leq m \leq 200\,000$) — количество заданий теста.

В следующих m строках заданы запросы. В i -й из следующих строк в начале содержится целое число t_i ($1 \leq t_i \leq 2$).

- Если $t_i = 1$, то далее строка содержит целое число a_i и символ c_i ($1 \leq a_i \leq n$). В этом случае требуется в строке s на позиции a_i заменить скобку на c_i . Гарантируется, что c_i является одной из скобок $() [] \{ \} \langle \rangle$.
- Если $t_i = 2$, то далее строка содержит два целых числа l_i и r_i ($1 \leq l_i \leq r_i \leq n$). В этом случае требуется узнать, считается ли подстрока s на позициях с l_i по позицию r_i правильной скобочной последовательностью в Австралии.

Формат выходных данных

Для каждого запроса второго типа выведите «Yes» (без кавычек), если скобочная последовательность считается правильной и «No» (без кавычек) иначе.

Примеры

стандартный ввод	стандартный вывод
6)()((7 2 1 6 1 4) 2 2 5 1 3 [1 4] 2 1 6 2 4 5	Yes Yes Yes No
10 >()) ([< } { 6 2 1 10 1 3 (2 1 10 2 2 5 1 2) 2 1 10	Yes No No Yes

Замечание

В первом примере:

1. В первом задании просят сказать, считается ли подстрока «)()((» правильной. Ответ «Yes», так как «)» и «()» считаются правильными, а эта подстрока представляется, как сумма таких строк.
2. Во втором задании просят заменить скобку на 4-й позиции на). После этого строка будет равна «)()()».
3. В третьем задании просят сказать, считается ли подстрока «()» правильной. Ответ «Yes» аналогично первому заданию.
4. В четвёртом задании просят заменить скобку на 3-й позиции на [. После этого строка будет равна «) ([()».
5. В пятом задании просят заменить скобку на 4-й позиции на]. После этого строка будет равна «) ([()».
6. В шестом задании просят сказать, считается ли подстрока «) ([()» правильной. Ответ «Yes» аналогично предыдущим заданиям, так как «[]» тоже считается правильной.
7. В седьмом задании просят сказать, считается ли подстрока «]» правильной. Но нетрудно убедиться, что правильной она не считается, поэтому ответ «No».

Во втором примере:

1. В первом задании просят сказать, считается ли подстрока «>()) ([< } {» правильной. Ответ «Yes», так как «()» и «>()» считаются правильными, поэтому их сумма «()» считается правильной, поэтому строка «>()) ([< } {» считается правильной. Также «} {» считается правильной, поэтому исходная подстрока считается правильной.
2. Во втором задании просят заменить скобку на 3-й позиции на (. После этого строка будет равна «>()) ([< } {».

3. В третьем задании просят сказать, считается ли подстрока «>() [] [<]{» правильной. Ответ «No», так как иначе строка «() (» считалась бы правильной, но нетрудно убедиться, что это не так.
4. В четвёртом задании просят сказать, считается ли подстрока «() (» правильной. Нетрудно убедиться, что правильной она не считается, поэтому ответ «No»
5. В пятом задании просят заменить скобку на 2-й позиции на). После этого строка будет равна «>) () [] [<]{».
6. В шестом задании просят сказать, считается ли подстрока «>) () [] [<]{» правильной. Ответ «Yes», так как «) () (» правильная, поэтому и «>) () (<» правильная, поэтому и исходная подстрока правильная.

Система оценки

Тесты к этой задаче состоят из 6 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Доп. ограничения			Необх. группы	Комментарий
		n	m	t_i		
0	0	–	–	–	–	Тесты из условия.
1	16	$n \leq 100$	$m \leq 100$	–	0	
2	15	$n \leq 10\,000$	$m \leq 10\,000$	–	0, 1	
3	12	$n \leq 10\,000$	–	$t_i = 2$	–	
4	13	–	–	–	–	В любой момент строка состоит только из скобок (и)
5	20	–	–	$t_i = 2$	3	
6	24	–	–	–	0 – 5	Offline-проверка.

Задача L. Калькулятор

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Имеется калькулятор, который умеет выполнять с текущим числом на экране следующие операции:

- Умножить текущее число на A .
- Умножить текущее число на B .
- Прибавить к текущему числу единицу.
- Прибавить к текущему числу число C .

Здесь A , B и C — некоторые целочисленные константы. Определите, какое наименьшее количество операций требуется, чтобы получить из числа 1 число n .

Формат входных данных

Во вводе задано несколько строк. Каждая строка содержит очередной тестовый случай, задаваемый четырьмя целыми числами n , A , B и C ($1 \leq n \leq 10^{19}$, $2 \leq A, B \leq 10$, $1 \leq C \leq 10$). Входные данные содержат не более 100 тестов.

Формат выходных данных

Для каждого тестового случая выведите $k + 1$, где k — количество сделанных операций. За ним выведите все $k + 1$ промежуточных чисел, включая 1 и n . Если последовательностей из k операций несколько, можно вывести любую из них.

Пример

стандартный ввод	стандартный вывод
1 7 8 9	1 1
5 2 3 9	4 1 3 4 5
576 8 9 7	4 1 9 72 576
100000 2 3 1	19 1 3 9 18 19 38 76 77 231 693 694 2082 2083 6249 62