

Задача А. Оптимальное бинарное дерево поиска

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

Рассмотрим множество $S = \{e_1, e_2, \dots, e_n\}$, состоящее из n различных элементов таких, что $e_1 < e_2 < \dots < e_n$. Рассмотрим бинарное дерево поиска, состоящее из элементов S . Чем чаще производится запрос к элементу, тем ближе он должен располагаться к корню. Стоимостью $cost$ доступа к элементу e_i из S в дереве будем называть значение $cost(e_i)$, равное числу ребер на пути, который соединяет корень с вершиной, содержащей элемент. Имея частоту запросов к элементам из S — $f(e_1), f(e_2), \dots, f(e_n)$ — определим общую стоимость дерева следующим образом:

$$\sum_{i=1}^n f(e_i) \cdot cost(e_i)$$

Дерево, имеющее наименьшую стоимость, считается наилучшим для поиска элементов из S . Именно поэтому оно называется Оптимальным Бинарным Деревом Поиска. Ваша задача — найти стоимость Оптимального Бинарного Дерева Поиска.

Формат входных данных

Состоит из нескольких тестов, каждый из которых расположен в отдельной строке. Первое число в строке n ($1 \leq n \leq 5000$) указывает на размер множества S . Следующие n неотрицательных целых чисел описывают частоты запросов элементов из S : $f(e_1), f(e_2), \dots, f(e_n)$. Известно, что $0 \leq f(e_i) \leq 100$. Сумма n по всем тестам не больше 5000.

Формат выходных данных

Для каждого теста в отдельной строке выведите стоимость Оптимального Бинарного Дерева Поиска.

Пример

стандартный ввод	стандартный вывод
1 5	0
3 10 10 10	20
3 5 10 20	20
6 1 3 5 10 20 30	63

Задача В. 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 С. Утилитаризм

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5.5 секунд
Ограничение по памяти: 256 мегабайт

В британском королевстве есть n городов, пронумерованных от 1 до n . Некоторые города связаны двусторонними дорогами. Всего есть ровно $n-1$ дорога, и между любыми двумя городами есть ровно один путь. Также с каждой дорогой связана некоторая ценность.

Сегодня, чтобы почтить отцов-основателей (их было, кстати, k) королевства, нынешний король Лелуш решил выбрать k различных дорог и подарить по одной дороге каждому из основателей. Чтобы избежать ненужных конфликтов, требуется, чтобы дороги не касались своими концами.

На самом деле, Лелушу не так-то и важно, кто получит какую дорогу. С другой стороны, его волнует суммарная ценность подарочных дорог. Вам нужно помочь Лелушу и выбрать k дорог, максимизируя эту суммарную ценность.

Формат входных данных

Первая строка содержит два целых числа n и k ($2 \leq n \leq 250\,000, 1 \leq k \leq n-1$) — количество городов и количество дорог, которые нужно подарить.

Каждая из следующих $n-1$ содержит три целых числа v_i, u_i, c_i ($1 \leq v_i, u_i \leq n, -10^6 \leq c_i \leq 10^6$) — концы очередной дороги и её ценность.

Формат выходных данных

Если раздать дороги нельзя, выведите «Impossible».

Иначе выведите одно число — максимальную суммарную цену k дорог.

Примеры

стандартный ввод	стандартный вывод
5 1 1 2 2 2 3 3 2 4 10 4 5 6	10
5 2 1 2 2 2 3 3 2 4 10 4 5 6	9
5 3 1 2 2 2 3 3 2 4 10 4 5 6	Impossible

Задача D. 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 Е. 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 фитилями. 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

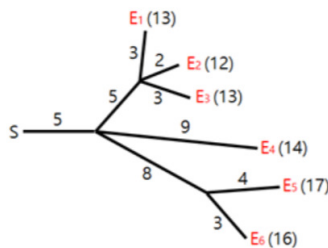


Рис. 1: Схема соединения

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.

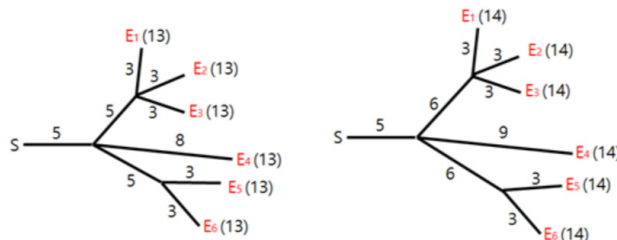


Рис. 2: Примеры изменения длин, после которых взрывы происходят одновременно

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. Балансировка дерева

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

Дано корневое дерево с N вершинами, корень дерева находится в вершине под номером 1. Каждое ребро, скажем i -тое, имеет два параметра, сопоставленные ему: C_i — стоимость ребра и параметр D_i .

Ваша задача — сделать равными стоимости всех путей из корня дерева в любой лист. Стоимость пути равна сумме всех стоимостей ребер на пути. Для того, чтобы решить задачу, Вы можете уменьшать или увеличивать стоимость какого-либо ребра. Для того чтобы сделать стоимость i -го ребра равной X , требуется $D_i \cdot |X - C_i|$ времени (в любой момент времени можно менять стоимость только одного ребра, т.е. все операции производятся последовательно, а не одновременно).

Найдите наименьшее время, требуемое для того, чтобы сделать равными стоимости всех путей из корня дерева в любой лист. Более того, выведите также новые стоимости ребер. Если существует более одного решения, то выведите любое из них.

Отметим, что вы можете сделать стоимости ребер отрицательными, но все они должны оставаться целыми.

Формат входных данных

Первая строка содержит единственное целое число T ($1 \leq T \leq 200\,000$) — количество тестовых случаев. Далее следует описание тестов в следующем формате:

Первая строка каждого теста содержит единственное целое число N ($1 \leq N \leq 200\,000$) — количество вершин в дереве. Каждая из следующих $N - 1$ строк содержит разделенные пробелами целые числа u_i, v_i, C_i, D_i ($1 \leq u_i, v_i \leq N$, $1 \leq C_i, D_i \leq 10^6$), задающие ребро с параметрами C_i и D_i , соединяющее вершины с номерами u_i и v_i .

Гарантируется, что сумма N по всем тестовым случаям не превосходит 200 000.

Формат выходных данных

Для каждого тестового случая выведите N строк. Первая строка должна содержать целое число — затраченное время. В каждой из следующих $N - 1$ строк i -я строка должна содержать новую стоимость i -го ребра.

Примеры

стандартный ввод	стандартный вывод
1	19
5	5
1 2 5 4	15
1 3 15 15	10
2 4 3 2	10
2 5 5 1	

Задача G. Соня и задача без легенды

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

Сова Соня не придумала легенду к этой задаче и просто написала вам формальное условие.

Дан массив длины n , содержащий целые положительные числа. За одну операцию можно увеличить или уменьшить на 1 любое число в массиве. Требуется сделать массив строго возрастающим за минимальное количество операций. Числа можно изменять произвольным образом, в том числе они могут стать отрицательными или равными 0.

Формат входных данных

Первая строка входных данных содержит единственное число n ($1 \leq n \leq 300006$) — длину массива.

Следующая строка содержит n целых чисел a_i ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите одно число — минимальное количество операций, которое потребуется применить, чтобы массив стал строго возрастающим.

Примеры

стандартный ввод	стандартный вывод
7 2 1 5 11 5 9 11	9
5 5 4 3 2 1	12