

## Задача А. Петя и прямоугольники

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.25 секунд  
Ограничение по памяти: 256 мегабайт

Маленький Петя очень любит прямоугольники. Петя дал маме список прямоугольников, которые он хочет получить в подарок на Новый Год. Каждый прямоугольник характеризуется  $w$  и высотой  $h$ .

Мама хочет сделать Пете приятное и купить все прямоугольники из его списка. Мама отправилась в магазин и узнала, что цена одного прямоугольника равна его площади. К ее счастью, в магазине действует предновогодняя акция, позволяющая покупать прямоугольники не по одному, а сразу наборами. Стоимость одного набора равна ширине самого широкого прямоугольника, умноженной на высоту самого высокого прямоугольника из этого набора. Обратите внимание, что поворачивать прямоугольники (тем самым меняя местами ширину и высоту) нельзя. Помогите маме Пети купить все прямоугольники из списка ее сына, потратив на это наименьшее количество денег.

### Формат входных данных

В первой строке записано число  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ) — количество прямоугольников в списке Пети. В каждой из следующих  $n$  строк записаны по 2 целых положительных числа, не превышающих  $10^6$ , — ширина и высота очередного прямоугольника.

### Формат выходных данных

Выведите одно число — наименьшее количество денег, которое может потратить мама чтобы купить Пете все прямоугольники из его списка.

### Пример

стандартный ввод	стандартный вывод
4 100 1 15 15 20 5 1 100	500

## Задача В. Коды, сохраняющие порядок

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 1024 мегабайта

Двоичный код — это код, где каждому символу сопоставляется последовательность из единиц и нулей. Код называется префиксным, если ни одно кодовое слово не является префиксом другого. Код называется сохраняющим порядок, если лексикографический порядок кодовых слов совпадает с алфавитным порядком символов.

Рассмотрим текст над алфавитом, содержащим  $n$  символов, в котором  $a_1$  раз встречается первый символ,  $a_2$  раз встречается второй символ,  $\dots$ ,  $a_n$  раз встречается  $n$ -й символ. Длина текста после кодирования его префиксным кодом, где первому символу сопоставлена строка длины  $l_1$ , второму — строка длины  $l_2$ , и т. д., будет равна  $a_1 \cdot l_1 + a_2 \cdot l_2 + \dots + a_n \cdot l_n$ .

Требуется найти сохраняющий порядок префиксный код, минимизирующий длину закодированного текста.

### Формат входных данных

Первая строка содержит число  $n$  — число символов в алфавите ( $2 \leq n \leq 2000$ ). Следующая строка содержит  $n$  целых чисел — сколько раз каждый символ встречается в тексте:  $a_1, a_2, \dots, a_n$ . Числа положительные и не превосходят  $10^9$ .

### Формат выходных данных

Выведите  $n$  двоичных последовательностей — искомый код.

### Пример

стандартный ввод	стандартный вывод
5	00
1 8 2 3 1	01
	10
	110
	111

## Задача С. Автобусные остановки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	256 мегабайт

В деревне есть  $n$  домов, расположенных вдоль главной дороги, которую можно воспринимать как числовую прямую.  $i$ -й дом имеет координату  $x_i$ .

Жители предпочитают автобусные остановки рядом с их домом, и чем дальше автобусная остановка, тем более несчастливы они. *Недовольство* дома определяется как квадрат расстояния между домом и ближайшей к нему автобусной остановкой. Ваша задача — построить  $k$  автобусных остановок вдоль главной дороги так, чтобы сумма недовольств домов была минимальна.

*Обратите внимание:* остановка может быть построена в любой точке числовой прямой, необязательно совпадающей с точкой какого-то из домов.

Формально, пусть ближайшая остановка к  $i$ -му дому находится в точке  $p_i$ . Тогда вы хотите минимизировать:

$$\sum_{i=1}^n |x_i - p_i|^2$$

### Формат входных данных

Первая строка содержит два целых числа  $n, k$  ( $1 \leq n \leq 5 \cdot 10^4$ ,  $1 \leq k \leq \min(n, 100)$ ).

Вторая строка содержит  $n$  целых чисел  $x_i$  ( $1 \leq x_i \leq 10^5$ ).

### Формат выходных данных

Выведите одно число — ответ с относительной или абсолютной погрешностью не более  $10^{-6}$ .

### Пример

стандартный ввод	стандартный вывод
3 2 1 2 4	0.5000000000000000

### Замечание

Пусть построили автобусные остановки в координатах 1.5 и 4.0. Тогда:

- Недовольство первого дома:  $(x_1 - p_1)^2 = (1.0 - 1.5)^2 = 0.25$
- Недовольство второго дома:  $(x_2 - p_1)^2 = (2.0 - 1.5)^2 = 0.25$
- Недовольство третьего дома:  $(x_3 - p_2)^2 = (4.0 - 4.0)^2 = 0.00$

Таким образом, суммарное недовольство равно  $0.25 + 0.25 + 0.00 = 0.5$

## Задача D. Задача «ИЛИ»

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 256 мегабайт

Дан массив  $a$  из  $n$  целых чисел. Стоимость отрезка массива — побитовое «ИЛИ» его элементов. Пусть вы разбили массив на  $k$  непустых отрезков и посчитали сумму их стоимостей. Какое максимальное значение вы можете получить?

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $k$  ( $1 \leq k \leq n \leq 2 \cdot 10^5$ ).

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i < 2^{30}$ ) — элементы массива  $a$ .

### Формат выходных данных

Выведите одно целое число — ответ на задачу.

### Пример

стандартный ввод	стандартный вывод
5 2 1 4 3 4 8	20

## Задача Е. 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 F. Ciel и гондолы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Лиса Ciel зашла в парк аттракционов. И вот, она в очереди на колесо обозрения. В очереди стоит  $n$  людей (хотя нет, скорее лис): мы будем считать, что первая лиса стоит в начале очереди, а  $n$ -я лиса стоит в хвосте очереди.

Всего имеется  $k$  гондол, мы распределяем лис по гондолам следующим образом:

- Когда подплывает первая гондола,  $q_1$  лис переходят из начала очереди в подплывшую гондолу.
- Затем, когда подплывает вторая гондола,  $q_2$  лис из начала оставшейся очереди переходит в эту гондолу.
- ...
- Оставшиеся  $q_k$  лис идут с последней ( $k$ -ю) гондолу.

Обратите внимание, что числа  $q_1, q_2, \dots, q_k$  должны быть положительными. Из условия следует, что  $\sum_{i=1}^k q_i = n$  и  $q_i > 0$ .

Вы знаете как лисам не хочется задерживаться в гондолах с незнакомцами. Итак, Ваша задача — найти оптимальный способ размещения (то есть определить оптимальную последовательность  $q$ ), чтобы угодить всем. Для каждой пары лис  $i$  и  $j$  задано значение  $u_{ij}$ , обозначающее степень незнакомости. Можете считать, что  $u_{ij} = u_{ji}$  для всех  $i, j$  ( $1 \leq i, j \leq n$ ) и что  $u_{ii} = 0$  для всех  $i$  ( $1 \leq i \leq n$ ). Тогда значение незнакомости в гондоле определяется как сумма значений незнакомости между всеми парами лис, которые находятся в этой гондоле. Общее значение незнакомости определяется как сумма значений незнакомости по всем гондолам.

Помогите лисе Ciel найти минимальное возможное значение общей незнакомости при некотором оптимальном распределении лис по гондолам.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $k$  ( $1 \leq n \leq 4000$  and  $1 \leq k \leq \min(n, 800)$ ) — количество лис в очереди и количество гондол. В следующих  $n$  строках записано по  $n$  целых чисел — матрица  $u$ , ( $0 \leq u_{ij} \leq 9$ ,  $u_{ij} = u_{ji}$  и  $u_{ii} = 0$ ).

Пожалуйста, используйте методы быстрого чтения (например, для Java используйте `BufferedReader` вместо `Scanner`).

### Формат выходных данных

Выведите целое число — минимальное возможное значение общей незнакомости.

## Примеры

стандартный ввод	стандартный вывод
5 2 0 0 1 1 1 0 0 1 1 1 1 1 0 0 0 1 1 0 0 0 1 1 0 0 0	0
8 3 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0	7

## Замечание

В первом примере можно распределить лис вот так: 1, 2 идут в одну гондолу, 3, 4, 5 идут в другую гондолу.

Во втором примере оптимальное распределение таково: 1, 2, 3 | 4, 5, 6 | 7, 8.

## Задача G. Жесткие диски

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

У вас есть кластер. В нем есть  $n$  пар жестких дисков (HDD). Каждый диск может быть представлен как точка с целой координатой на бесконечной прямой. В каждой паре один из HDD главный, а второй — запасной.

Вы хотите настроить  $k$  компьютеров, которые тоже представляются как точки с целыми координатами на той же прямой, после чего соединить все диски с компьютерами при помощи проводов. В конечном итоге, основной и запасной диски из одной пары должны быть соединены с одним и тем же компьютером (и только с ним), но каждый компьютер может быть соединен с любым количеством дисков (возможно с нулем). Каждый провод соединяет один HDD с одним компьютером, и длина провода равна расстоянию между соответствующими точками на прямой. Необходимо найти минимальную суммарную длину всех проводов.

### Формат входных данных

В первой строке содержатся два целых числа  $n$  и  $k$  — количество пар дисков и количество компьютеров, соответственно ( $2 \leq k \leq n \leq 100\,000$ ;  $4 \leq k \times n \leq 100\,000$ ). Каждая из следующих  $n$  строк содержат по два целых числа  $a_i$  и  $b_i$  — координаты основного и запасного дисков ( $-10^9 \leq a_i, b_i \leq 10^9$ ).

### Формат выходных данных

Вывод должен содержать ровно одно число — ответ на задачу.

### Пример

стандартный ввод	стандартный вывод
5 2 6 7 -1 1 0 1 5 2 7 3	13

### Замечание

В примере оптимальные позиции компьютеров — 0 и 6. Соединим вторую и третью пары дисков с первым компьютером, а остальные со вторым. Тогда суммарная длина проводов, ведущих к первому компьютеру, будет 3, и 10 для второго, что в сумме дает 13.



## Задача Н. Шоппинг

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	256 мегабайт

Сегодня долгожданный для всех школьников — первый день каникул нового учебного года. Наша главная героиня — Дени — учится в 10 классе. Она хорошо подготовилась к сегодняшнему дню и выяснила, что в центре города находятся  $N$  магазинов. Теперь Дени планирует вместе со своими друзьями посетить некоторые из них. В городе есть  $M$  пар магазинов  $(x_i, y_i)$ , соединенных двусторонними дорогами. Для каждой дороги известно время, которое требуется для перемещения по ней, оно одно и то же для перемещения в обоих направлениях. Никакой магазин не соединен дорогой сам с собой, никакая пара магазинов не соединена более чем одной дорогой.

Дени очень суеверна и одно из её суеверий заключается в том, что она верит, что время, потраченное на перемещения между магазинами, должно нацело делиться на  $D$ . При этом Дени с друзьями не может перемещаться между магазинами слишком долго, её путь должен занимать суммарно не больше  $K$ . Дени очень любопытна. Она хочет выяснить, сколько существует различных способов начать свой путь в некотором магазине, перемещаться по дорогам между магазинами, и закончить путь в некотором магазине (возможно посещая по пути некоторые магазины и/или дороги более одного раза). Дени помнит, что у нее есть друг-программист — вы — и она просит написать программу, которая вычислит количество корректных способов перемещаться между магазинами. Дени считает способ корректным, если её время в пути не превышает  $K$  и делится на  $D$ . Вы немедленно указываете Дени, что количество путей может быть слишком большим, поэтому Дени просит вывести остаток от деления количества путей на число 1 000 000 007.

### Формат входных данных

На первой строке ввода находятся четыре целых числа  $N$ ,  $M$ ,  $D$  и  $K$  ( $2 \leq N \leq 30$ ,  $2 \leq M \leq 435$ ,  $2 \leq D \leq K \leq 10^9$ ).

На каждой из следующих строк находятся по три целых числа  $x_i$ ,  $y_i$  и  $t_i$  ( $1 \leq x_i, y_i \leq N$ ,  $1 \leq t_i \leq 10$ ) — они задают двустороннюю дорогу между магазинами  $x_i$  и  $y_i$ , перемещение по которой занимает  $t_i$  ( $1 \leq i \leq M$ ).

### Формат выходных данных

Выведите остаток от деления количества искомых путей на 1 000 000 007.

## Примеры

стандартный ввод	стандартный вывод
3 3 2 2 1 2 1 2 3 2 3 1 1	8
5 7 5 10 1 3 8 2 5 7 3 4 3 1 4 2 2 3 1 1 5 4 4 5 4	58
5 9 2 20 1 2 1 2 3 2 3 1 1 3 4 1 4 5 2 5 3 1 1 5 1 2 4 1 2 5 1	989802661
5 7 5000000 5000000 1 3 8 2 5 7 3 4 3 1 4 2 2 3 1 1 5 4 4 5 4	598634781

## Задача I. Гладкие числа

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

Назовем число гладким, если его цифры, начиная со старшего разряда, образуют неубывающую последовательность. Упорядочим все такие числа в возрастающем порядке и присвоим каждому номер. Вам требуется по номеру  $N$  вывести  $N$ -ое гладкое число.

### Формат входных данных

На вход программы поступает номер  $N$  ( $1 \leq N \leq 2147483647$ ).

### Формат выходных данных

Выведите соответствующее номеру  $N$  гладкое число.

### Примеры

стандартный ввод	стандартный вывод
3	3
11	12

## Задача J. Сбежать через лист

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.3 секунд  
Ограничение по памяти: 256 мегабайт

Дано дерево на  $n$  вершинах (пронумерованных от 1 до  $n$ ) с корнем в вершине 1. В вершине  $i$  записаны два числа:  $a_i$  и  $b_i$ .

Вы можете прыгнуть из вершины в любую вершину в её поддереве. Стоимость такого прыжка из вершины  $x$  в вершину  $y$  равна произведению  $a_x$  и  $b_y$ . Суммарная стоимость пути между вершинами, состоящего из нескольких прыжков равна сумме стоимостей прыжков в нём. Для каждой вершины посчитайте минимальную стоимость пути от неё до какого-либо листа. Обратите внимание, что корень дерева не является листом, даже если имеет степень 1.

Учтите, что нельзя совершать прыжок из вершины в ту же вершину.

### Формат входных данных

В первой строке содержится целое число  $n$  ( $2 \leq n \leq 10^5$ ) — количество вершин в дереве.

Во второй строке через пробел заданы  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $-10^5 \leq a_i \leq 10^5$ ).

Во третьей строке через пробел заданы  $n$  целых чисел  $b_1, b_2, \dots, b_n$  ( $-10^5 \leq b_i \leq 10^5$ ).

В следующих  $n - 1$  строках содержатся пары целых чисел  $u_i$  и  $v_i$  ( $1 \leq u_i, v_i \leq n$ ), разделённых пробелом, обозначающие ребро между вершинами  $u_i$  и  $v_i$  в дереве.

### Формат выходных данных

Выведите  $n$  целых чисел через пробел,  $i$ -е из которых обозначает минимальную стоимость, чтобы добраться от вершины с номером  $i$  до какого-либо листа.

### Примеры

стандартный ввод	стандартный вывод
3 2 10 -1 7 -7 5 2 3 2 1	10 50 0
4 5 -10 5 7 -8 -80 -3 -10 2 1 2 4 1 3	-300 100 0 0

### Замечание

В первом тестовом примере вершина 3 сама является листом, поэтому ответ равен 0. Для вершины 2 прыжок в вершину 3 стоит  $a_2 \times b_3 = 50$ . Для вершины 1 прыжок в вершину 3 стоит  $a_1 \times b_3 = 10$ .

Во втором тестовом примере вершины 3 и 4 являются листьями, поэтому ответ для них равен 0. Для вершины 2 прыжок в вершину 4 стоит  $a_2 \times b_4 = 100$ . Для вершины 1 необходимо сначала прыгнуть в вершину 2 прыжком стоимостью  $a_1 \times b_2 = -400$ , а затем прыгнуть из 2 в 4 за  $a_2 \times b_4 = 100$ .

## Задача К. Не выделяйся

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

Дан массив целых неотрицательных чисел  $a$  длины  $n$  и число  $k$ . За одну операцию можно изменить любой элемент массива на 1. Необходимо сделать так, чтобы разница всех соседних элементов массива  $a$  была не больше  $k$  по абсолютному значению, за минимальное количество операций. При этом делать элементы массива отрицательными запрещено.

### Формат входных данных

В первой строке входных данных даны два целых неотрицательных числа  $n$  и  $k$  ( $1 \leq n \leq 200\,000$ ,  $0 \leq k \leq 10^9$ ) — длина массива  $a$  и максимальная допустимая разница соседних элементов.

Во второй строке входных данных даны  $n$  целых неотрицательных чисел  $a_i$  ( $0 \leq a_i \leq 10^9$ ) — элементы массива  $a$ .

### Формат выходных данных

Выведите одно число — минимальное количество операций, которые надо сделать, чтобы абсолютное значение разницы любых соседних элементов массива было не больше  $k$ .

### Примеры

стандартный ввод	стандартный вывод
6 1 2 10 0 2 4 3	10
6 3 2 10 2 6 4 3	6
4 1 1 4 1 4	4
10 1 10 9 8 7 6 5 4 3 2 1	0
3 0 1 1 3	2

## Задача М. Калила и Димна на лесозаготовках

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

Калила и Димна — два шакала. Они живут в огромных джунглях. Однажды шакалы решили устроиться на завод лесозаготовки и подработать.

Управляющий завода хочет, чтобы они отправились в джунгли и срубили  $n$  деревьев высотой  $a_1, a_2, \dots, a_n$ . Для этого Калила и Димна купили цепную пилу в магазине. Каждый раз, когда они используют пилу на дереве номер  $i$ , они уменьшают высоту этого дерева на единицу. Каждый раз Калила и Димна должны заправить пилу для использования. Цена заправки зависит от того, какие деревья полностью спилены (дерево считается полностью спиленным, если его высота равна 0). Если максимальный идентификатор полностью срубленного дерева равняется  $i$  (первоначально это дерево имело высоту  $a_i$ ), то цена заправки пилы равняется  $b_i$ . Если ни одно дерево не срублено полностью, то заправлять пилу запрещается. Изначально пила заправлена. Известно, что для каждого  $i < j$ ,  $a_i < a_j$  и  $b_i > b_j$ , а также  $b_n = 0$  и  $a_1 = 1$ .

Калила и Димна хотят полностью срубить все деревья с минимальными затратами. Они ждут Вашей помощи! Поможете?

### Формат входных данных

В первой строке записано целое число  $n$  ( $1 \leq n \leq 10^5$ ). Во второй строке записано  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ). В третьей строке записано  $n$  целых чисел  $b_1, b_2, \dots, b_n$  ( $0 \leq b_i \leq 10^9$ ).

Гарантируется, что  $a_1 = 1$ ,  $b_n = 0$ ,  $a_1 < a_2 < \dots < a_n$  и  $b_1 > b_2 > \dots > b_n$ .

### Формат выходных данных

В единственной строке должна быть записана минимальная стоимость вырубания всех деревьев.

### Примеры

стандартный ввод	стандартный вывод
5 1 2 3 4 5 5 4 3 2 0	25
6 1 2 3 10 20 30 6 5 4 3 2 0	138

## Задача N. Иннофон

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	512 мегабайт

Одна телекоммуникационная компания планирует в скором будущем выпустить на рынок сразу два инновационных смартфона. Эти смартфоны будут называться «иннофон» и «иннофон плюс». Устройства уже полностью готовы к производству, и последняя задача, которую необходимо решить руководству компании, — выбрать оптимальную цену для каждого из смартфонов.

Аналитики компании провели исследование, в результате которого построили следующую модель. Всего есть  $n$  потенциальных покупателей инновационных смартфонов. Для принятия решения  $i$ -й покупатель использует следующий алгоритм, характеризующийся двумя числами  $a_i$  и  $b_i$  ( $a_i \geq b_i$ ):

- если цена на «иннофон плюс» не больше  $a_i$ , то он покупает «иннофон плюс»,
- иначе, если цена на «иннофон» не больше  $b_i$ , то он покупает «иннофон»,
- иначе он не покупает ничего.

Руководство компании хочет установить цены на «иннофон» и «иннофон плюс» таким образом, чтобы обе цены были целым числом, цена «иннофона» была не больше цены «иннофона плюс», и при этом суммарная стоимость проданных смартфонов была максимальна.

Требуется написать программу, которая находит максимально возможную суммарную стоимость проданных смартфонов.

### Формат входных данных

В первой строке содержится целое число  $n$  ( $1 \leq n \leq 150\,000$ ) — число потенциальных покупателей.

В следующих  $n$  строках содержатся по два целых числа  $a_i, b_i$  ( $0 \leq b_i \leq a_i \leq 10^9$ ) — характеристики алгоритма выбора телефона покупателем с номером  $i$ .

### Формат выходных данных

Выведите одно целое число — максимальную возможную суммарную стоимость проданных смартфонов.

### Примеры

стандартный ввод	стандартный вывод
5 80 20 60 50 40 40 15 10 70 30	220
1 50 0	50

### Пояснение к примеру

В первом примере для достижения максимальной суммы следует назначить цены на «иннофон» и «иннофон плюс» равными 40 и 70 соответственно. Тогда первый и пятый покупатель купят «иннофон плюс», второй и третий покупатель купят «иннофон», четвертый покупатель не купит ничего. Суммарная стоимость проданных смартфонов будет  $70 + 40 + 40 + 0 + 70 = 220$ .

Во втором примере нужно сделать цену «иннофона плюс» равной 50. Цена на «иннофон» при этом не важна.

## Задача О. Доставка пиццы

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

Флатландия — одномерная страна. Это значит, что каждая точка имеет только одну координату. Во Флатландии все любят пиццу (потому что она достаточно плоская).

Там есть  $n$  пиццерий и  $m$  покупателей.  $i$ -я пиццерия находится в точке  $s_i$ , а  $i$ -й покупатель — в точке  $c_i$ . Координаты любых двух пиццерий различны, но координаты покупателей могут совпадать.

Каждый покупатель хочет заказать пиццу и потратить минимально возможное количество денег.  $i$ -я пиццерия продаёт пиццу по цене  $p_i$ . Доставка из точки  $x_1$  в точку  $x_2$  стоит  $(x_1 - x_2)^2$ .

К сожалению, некоторые покупатели не любят некоторые пиццерии, поэтому они не будут заказывать пиццу оттуда. А именно,  $i$ -й покупатель не закажет пиццу из пиццерий  $d_{i,1}, \dots, d_{i,k_i}$ .

Для каждого покупателя найдите цену, за которую он закажет пиццу.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 200000$ ) — количество пиццерий и покупателей, соответственно.

$i$ -я из следующих  $n$  строк содержит два целых числа  $s_i$  и  $p_i$  ( $0 \leq s_i \leq 10^9$ ,  $1 \leq p_i \leq 10^9$ ) — координата  $i$ -й пиццерии и цена пиццы там, соответственно.

$i$ -я из следующих  $m$  строк содержит целые числа  $c_i, k_i, d_{i,1}, \dots, d_{i,k_i}$  ( $0 \leq c_i \leq 10^9$ ,  $0 \leq k_i \leq n - 1$ ,  $1 \leq d_{i,j} \leq n$ ) — координата  $i$ -го покупателя, количество пиццерий, которые он не любит, и номера этих пиццерий, соответственно.

Также гарантируется, что  $\sum k_i \leq 400000$ .

### Формат выходных данных

Выведите  $m$  чисел,  $i$ -е из которых — цена, за которую  $i$ -й покупатель закажет пиццу.

### Пример

стандартный ввод	стандартный вывод
3 3	11
1 7	34
10 5	13
8 9	
3 0	
3 1 1	
6 2 1 2	

### Замечание

Первый покупатель любит все пиццерии, поэтому закажет пиццу из первой. Это будет стоить  $7 + (3 - 1)^2 = 11$ .

Второй покупатель не любит пиццу из первой пиццерии, несмотря на то, что это самый дешёвый вариант. Он закажет пиццу из второй. Это будет стоить  $9 + (10 - 3)^2 = 34$ .

Третий покупатель любит пиццу только из третьей пиццерии, поэтому он закажет её там, и это будет стоить  $9 + (8 - 6)^2 = 13$ .



## Задача Р. Транспортировка кошек

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

Zxr960115 содержит большое хозяйство. Он кормит  $m$  милых кошечек и держит у себя  $p$  кормильщиков. Через ферму проходит прямая дорога, а вдоль дороги расположено  $n$  холмов, пронумерованных от 1 до  $n$ , слева направо. Расстояние от холма  $i$  до  $i-1$  равняется  $d_i$  метров. Кормильщики живут на холме 1.

Однажды кошечкам захотелось порезвиться и они разбежались. Кошка  $i$  пошла к холму  $h_i$ , дошла до него в момент времени  $t_i$ , а затем стала ждать кормильщика на холме  $h_i$ . Кормильщики должны собрать всех разбежавшихся кошек. Каждый кормильщик идет прямо от холма номер 1 до холма номер  $n$ , не останавливаясь у какого-либо холма, и собирает всех кошек, **ожидающих** на каждом холме. Кормильщики двигаются со скоростью 1 в единицу времени и достаточно сильны, чтобы собрать сколько угодно кошек.

Например, пусть имеется два холма ( $d_2 = 1$ ) и одна кошечка, которая дошла до холма 2 ( $h_1 = 2$ ) в момент времени 3. Тогда, если кормильщик отправится за кошками от холма 1 в момент времени 2 или 3, то он сможет забрать эту кошку. Но если он отправится от холма 1 в момент времени 1, то он не сможет этого сделать. Если кормильщик отправится за кошкой в момент времени 2, то кошка будет ждать его 0 единиц времени, если же он отправится в момент времени 3, то кошка будет ждать его 1 единицу времени.

Ваша задача — составить расписание отправки от холма 1 для кормильщиков так, чтобы общее время ожидания кошек до того как их заберут было минимальным.

### Формат входных данных

В первой строке входных данных содержится три целых числа  $n, m, p$  ( $2 \leq n \leq 10^5, 1 \leq m \leq 10^5, 1 \leq p \leq 100$ ).

Во второй строке содержится  $n - 1$  положительных целых чисел  $d_2, d_3, \dots, d_n$  ( $1 \leq d_i < 10^4$ ).

В каждой из следующих  $m$  строк содержится по два целых числа  $h_i$  и  $t_i$  ( $1 \leq h_i \leq n, 0 \leq t_i \leq 10^9$ ).

### Формат выходных данных

Выведите целое число, минимальную сумму времен ожидания всех кошек.

### Пример

стандартный ввод	стандартный вывод
4 6 2 1 3 5 1 0 2 1 4 9 1 10 2 10 3 12	3

## Задача Q. Красные и синие лампочки

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

Есть  $n$  лампочек с номерами  $1, 2, \dots, n$ . Надо ровно  $m$  из них сделать красными, а все остальные — синими, чтобы максимизировать следующую величину: по всем  $i \in [1, n - 1]$ : если лампочки  $i$  и  $i + 1$  разного цвета, то добавить к ответу  $a[i]$ .

### Формат входных данных

Первая строка содержит два числа —  $n$  и  $m$  ( $2 \leq n \leq 2 \cdot 10^5, 1 \leq m \leq n - 1$ ).

Вторая строка содержит  $n - 1$  число  $a_1, a_2, \dots, a_{n-1}$  ( $1 \leq a_i \leq 10^9$ ).

### Формат выходных данных

Выведите максимальную сумму, которую можно получить, если ровно  $m$  лампочек красные.

### Примеры

стандартный ввод	стандартный вывод
6 2 3 1 4 1 5	11
7 6 2 7 1 8 2 8	10
11 7 12345 678 90123 45678901 234567 89012 3456 78901 23456 7890	46207983