Задача A. 3 точки

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

Даны три целых числа a, b и c — координаты точек на числовой прямой. За одну операцию можно выбрать упорядоченную пару точек, координату одной из них увеличить на 1, а координату другой уменьшить на 1. Иными словами, если у нас были две точки с координатами u и v, мы выбрали пару (u, v), то после операции у нас будут точки с координатами $u+1$ и $v-1$. Определите, возможно ли такими операциями сделать координаты всех точек равными, и если это возможно, то найдите минимальное количество операций за которое это можно сделать. В некоторых тестах, также необходимо найти последовательность операций позволяющих этого добиться.

Формат входных данных

Первая строка содержит целое число t ($t = 0$ или $t = 1$). В случае, если $t = 0$ необходимо вывести только минимальное количество операций, а в случае, если $t = 1$ необходимо также вывести сами операции.

Вторая строка содержит три целых числа a, b и c — изначальные координаты точек на числовой прямой ($|a|, |b|, |c| \leqslant 10^9,$ если $t = 0$ и $|a|, |b|, |c| \leqslant 10^5,$ если $t = 1$).

Формат выходных данных

В первой строке выведите Yes или No, в зависимости от того, можно ли сделать координаты всех точек равными.

Во второй строке выведите минимальное количество операций.

Если $t = 1$, то в $(i+2)-$ ой строке выведите u и v, если $i-$ая операция заключалась в выборе пары (u, v).

Если возможных вариантов ответа несколько — выведите любой из них.

Система оценки

В этой задаче 20 тестов, не считая тестов из условия. Каждый тест оценивается независимо в 5 баллов.

Решения, верно работающие при $t = 0$, будут получать не менее 50 баллов.

Примеры

стандартный ввод	стандартный вывод
0 1 4 2	No
1 5 6 7	Yes 1 5 7
0 -10000 0 10000	Yes 10000

Задача B. Коммуникация на высоком уровне

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

В городе в ряд построено n новых небоскребов, которые вы хотите обеспечить современной связью. Для этого вы хотите установить по датчичку на каждом небоскребе. На i-м из них вы можете его установить не ниже a_i и не выше b_i. Задержкой для двух датчиков на высотах h_1 и h_2 называется величина $|h_1 - h_2|$. Вы хотите минимизировать сумму задержек для пар соседних зданий. Более формально, если датчики выставлены на высотах $d_1, d_2, \ldots, d_n$, требуется минимизировать величину $|d_1 - d_2| + |d_2 - d_3| + \ldots + |d_{n-1} - d_n|$.

Формат входных данных

Первая строка входных данных содержит единственное целое число t ($1 \leqslant t \leqslant 10^3$) — количество наборов входных данных. Описание наборов входных данных следует ниже.

Первая строка каждого набора входных данных содержит одно целое число n ($1 \leqslant n \leqslant 10^6$) — длину массивов a, b.

Следующие две строки содержат по n целых чисел: массивы a и b ($0 \leqslant a_i \leqslant b_i \leqslant 10^9$) соответственно.

Гарантируется, что сумма n по всем наборам входных данных не превосходит 10^6. В системе оценки сумма n обозначена как sum_n.

Формат выходных данных

Для каждого набора входных данных выведите две строки. Первая строка должна содержать ответ - минимальную суммарную задержку. Вторая строка должна содержать n целых чисел $d_1, d_2, \ldots, d_n$ - высоты расставленных датчиков. Должно выполняться $a_i \leqslant d_i \leqslant b_i$. Если решений несколько, выведите любое.

Система оценки

Задача состоит из 20 тестов, не считая тестов из условия. Каждый тест оценивается независимо в 5 баллов. Все тесты можно разделить на следующие группы:

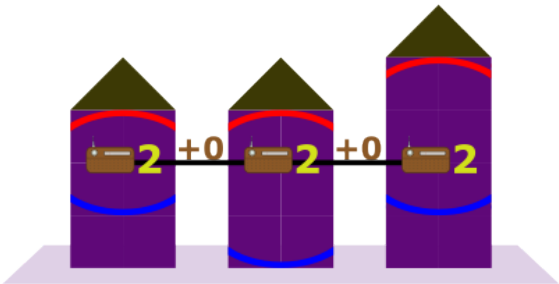
Группа	Макс. балл	Доп. ограничения			Комментарий
		n	sum_n	b_i	
0	0	–	–	–	Тесты из условия
1	20	$n \leqslant 20$	$sum_n \leqslant 2\,000$	$b_i \leqslant 20$	
2	20	$n \leqslant 500$	$sum_n \leqslant 2\,000$	$b_i \leqslant 1000$	
3	30	$n \leqslant 500$	$sum_n \leqslant 2\,000$	–	
4	30	–	–	–	

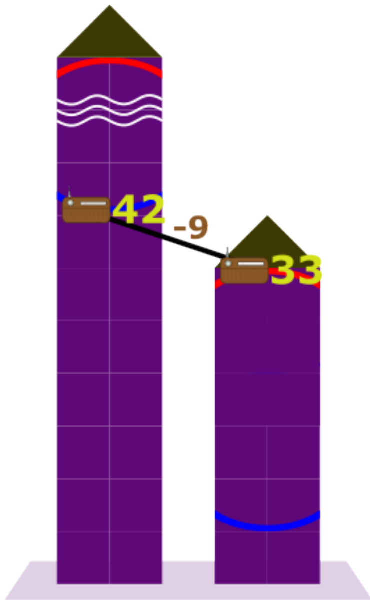
Пример

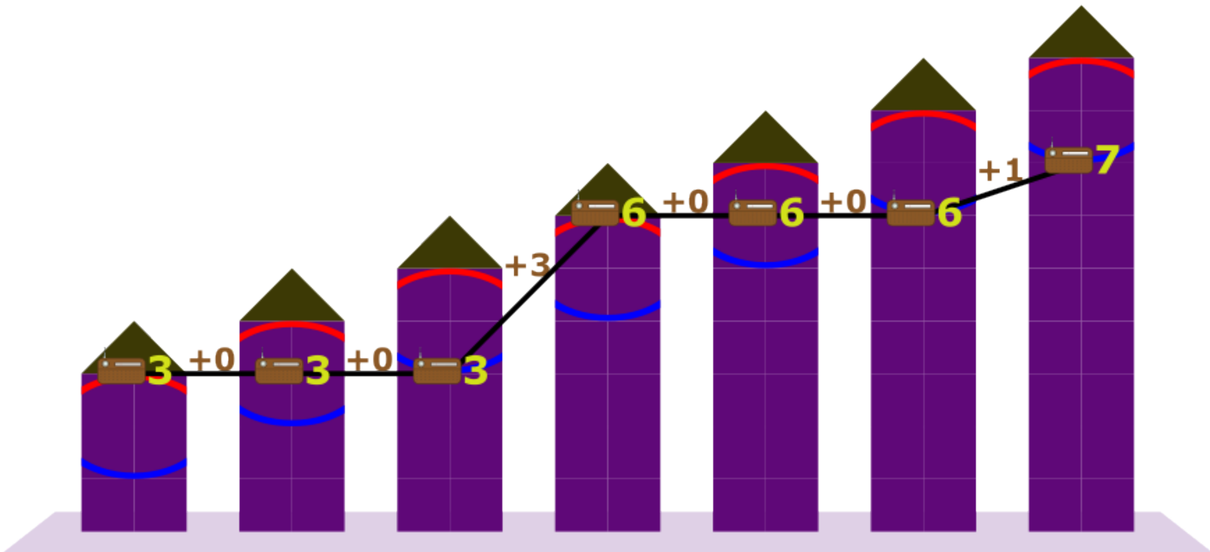
стандартный ввод	стандартный вывод
3 3 1 0 1 3 3 4 2 42 10 239 33 7 1 2 3 4 5 6 7 3 4 5 6 7 8 9	0 2 2 2 9 42 33 4 3 3 3 6 6 6 7

Замечание

Ниже приведены иллюстрации для решений тестовых случаев из примера.







Задача C. Следствие вели

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

Вам дана последовательность бит $a_1, a_2, \ldots, a_n$, где $a_i \in \{0, 1\}$, а также бит r. Вам нужно поставить **максимум одну** пару скобок в выражении $a_1 \Rightarrow a_2 \Rightarrow \ldots \Rightarrow a_n$ так, чтобы выражение равнялось r. Здесь $\Rightarrow$ означает битовую импликацию (следствие). Эта операция задаётся следующей таблицей истинности:

x	y	$x \Rightarrow y$
0	0	1
0	1	1
1	0	0
1	1	1

Выражение из нескольких импликаций вычисляется слева направо. Разрешается ставить скобки даже если выражение изначально равнялось r.

Формат входных данных

В первой строке даётся одно натуральное число n ($2 \leqslant n \leqslant 5 \cdot 10^5$)— количество бит в выражении.

Во второй строке даётся n бит — переменные $a_1, a_2, \ldots, a_n$ ($a_i \in \{0, 1\}$)

В третьей строке даётся один бит r — требуемый результат выражения.

Формат выходных данных

Если поставить пару скобок в выражении так, чтобы оно стало равняться искомому, невозможно, то выведите одно число «-1» (без кавычек).

Если ставить скобки в выражение не требуется, выведите одно число «0».

Иначе выведите 2 числа l и r, где $l < r$ — перед какой по счёту переменной нужно поставить открывающую скобку и после какой переменной нужно поставить закрывающую. Обратите внимание, что такая расстановка скобок некорректна: $0 \Rightarrow (1) \Rightarrow 1$.

Система оценки

Задача состоит из 20 тестов, не считая тестов из условия. Каждый тест оценивается независимо в 5 баллов.

Решения, верно работающие для $n \leqslant 100$ получат не менее 30 баллов.

Решения, верно работающие для $n \leqslant 5000$ получат не менее 60 баллов.

Примеры

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

Замечание

В первом примере после установки открывающей скобки до второго элемента и закрывающей после третьего получается выражение $0 \Rightarrow (1 \Rightarrow 0) = 0 \Rightarrow 0 = 1$. Это единственный ответ для данного теста.

Во втором примере изначальное выражение уже имеет значение 0. Также корректным ответом будет, например, «2 4».

В третьем примере при любой расстановке скобок значение выражения будет 1.

Задача D. Выбор полосы

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	512 мегабайт

Вы решили проехать по платной дороге, которая состоит из K полос. Также на этой дороге стоит $N+1$ терминал для оплаты проезда (один в начале, другой в конце и остальные посередине дороги). Терминалы нумеруются числами от 1 до $N+1$, где 1 — терминал у начала дороги, а $N+1$ — терминал у конца дороги.

Вы знаете, что время проезда между терминалом i и терминалом $i+1$ по полосе j $(1 \leqslant i \leqslant N, 1 \leqslant j \leqslant K)$ равно $A_{i,j}$. Также в любом терминале вы можете сменить полосу, каждое перемещение на соседнюю полосу занимает X минут. Можно сместится на несколько полос.

Вам нужно найти, за какое минимальное время вы сможете добраться от начала дороги (от любой полосы терминала 1) до конца дороги (любой полосы терминала $N+1$).

Кроме этого, в будущем планируется Q ремонтов, занумерованных от 1 до Q. Нужно определить минимальное время проезда во время ремонтов. Во время ремонта i по полосе l_i нельзя проехать между терминалами t_i и t_i+1. Ремонты происходят последовательно, одновременно идёт только один ремонт. Обратите внимание, что в некоторых подзадачах $Q=0$, то есть ремонтов не будет.

Формат входных данных

В первой строке вводятся три целых числа N, K и X $(2 \leqslant N, K \leqslant 10^6, N \cdot K \leqslant 10^6, 1 \leqslant X \leqslant 10^9)$ — число терминалов, полос и время смены полосы на соседнюю соответственно.

В следующих N строках содержится по K целых чисел $A_{i,1}$, $A_{i,2}$, $A_{i,3}$, $\ldots$, $A_{i,K}$ $(1 \leqslant A_{i,j} \leqslant 10^9)$ — времена проезда между терминалами.

В следующей строке вводится одно целое число Q $(0 \leqslant Q \leqslant 10^6)$ — количество ремонтов.

В следующих Q строчках вводится по два целых числа t_i и l_i $(1 \leqslant t_i \leqslant N, 1 \leqslant l_i \leqslant K)$ — параметры ремонта.

Формат выходных данных

В первой строке выведите минимальное время, за которое вы можете добрать от начала до конца дороги.

В следующих Q строках — минимальное время, за которое вы можете добрать от начала до конца дороги во время ремонта.

Система оценки

В этой задаче 25 тестов, кроме тестов из условия. Каждый тест оценивается в 4 балла. Тесты можно разделить на следующие группы:

Номер	Макс. балл	Ограничения		
		N	K	Q
1	12	$N \leqslant 10$	$K \leqslant 2$	$Q = 0$
2	12	$N \leqslant 10$	$K \leqslant 10$	$Q = 0$
3	12	$N \leqslant 100$	$K \leqslant 300$	$Q = 0$
4	12	$N \leqslant 100$	$K \leqslant 300$	$Q \leqslant 100$
5	12	$N \leqslant 100$	$K \leqslant 10^4$	$Q = 0$
6	12	$N \leqslant 10^4$	$K \leqslant 300$	$Q \leqslant 10^4$
7	28	–	–	–

Примеры

стандартный ввод	стандартный вывод
3 3 2 12 2 10 10 10 4 3 7 8 2 1 1 1 2	15 15 21
3 2 5 20 30 10 5 20 10 6 1 1 1 2 2 1 2 2 3 1 3 2	40 45 40 40 45 40 50

Замечание

В первом тестовом примере минимальное время достигается следующим образом:

1. Путь начинается с полосы номер 2. После этого мы доезжаем до терминала номер 2 тратя на это 2 минуты.

2. Далее требуется перейти с полосы номер 2 на полосу с номером 3, затратив на это дополнительно 2 минуты, а время для достижение третьего терминала будет равно 4 минутам. Суммарное время для достижения терминала номер 3 равно $2 + 2 + 4 = 8$ минут.

3. Далее требуется перейти с полосы номер 3 на первую полосу. Для этого потребуется дополнительно $2 \cdot 2 + 3 = 7$ минут. Суммарное время для достижения последнего терминала — $8 + 2 \cdot 2 + 3 = 15$ минут.

Ответ на первый запрос — 15 минут, т. к. наш исходный путь не использует полосу номер 1.

Ответ на второй запрос — 21 минута, потому что оптимальный путь теперь начинается с полосы номер 3.

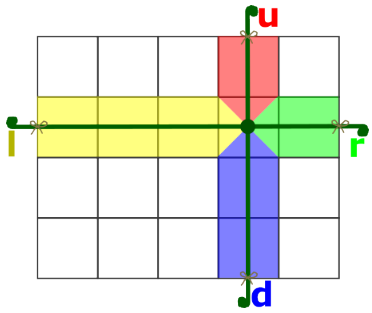
Задача E. Подвязывание малины

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

Ваш дачный участок представляет собой прямоугольное пространство огороженное забором по периметру. Дачный участок разбит на квадратные зоны размера 1×1 в n горизонтальных и m вертикальных рядов.

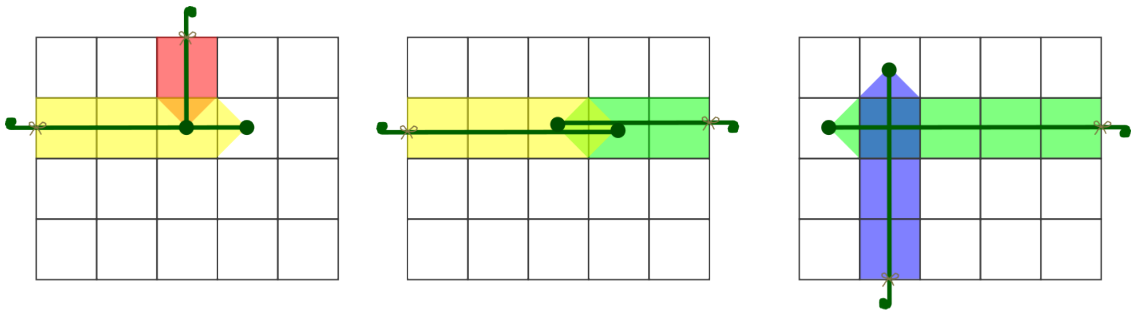
В центре некоторых квадратных зон растет 4 одинаковых стебля малины длиной l_{ij}. Вы хотите подвязать как можно больше стеблей, вырвав остальные. Чтобы подвязать стебель, вы можете протянуть его вдоль земли до забора и привязать к нему. Стеблю должно хватать длины. Протягивать можно только параллельно сторонам забора.

Каждый протянутый стебель занимает все зоны на пути от себя до забора и четверть своей зоны как показано на рисунке(занятое пространство каждым стеблем указано отдельным цветом):



Никакие два стебля не могут занимать одно и то же пространство. То есть для каждой зоны со стеблями вы потенциально можете подвязать любое количество стеблей от 0 до 4, но никакие два из них не могут идти в одном направлении.

Ниже указано три примера, где пространства, занятые стеблями, пересекаются. Такие подвязывания некорректны.



Подвяжите как можно больше стеблей и выведите какие именно из них и как надо привязать.

Формат входных данных

Первая строка входных данных содержит единственное целое число t ($1 \leqslant t \leqslant 10^3$) — количество наборов входных данных. Описание наборов входных данных следует ниже.

Первая строка каждого набора входных данных содержит три целых числа n, m и s ($1 \leqslant n, m \leqslant 10^6$, $1 \leqslant s \leqslant min(10^5, n \cdot m)$) — размер участка и количество зон со стеблями.

Следующие s строк содержат по 3 целых числа r_i, c_i, l_i ($1 \leqslant r_i \leqslant n$, $1 \leqslant c_i \leqslant m$, $1 \leqslant l_i \leqslant 10^6$) — строку, столбец и длину i-го набора стеблей. Гарантируется, что каждая пара (r_i, c_i) встречается в наборе входных данных не более раза.

Гарантируется, что сумма $n \cdot m$ по всем наборам входных данных не превосходит 10^6, сумма s по всем наборам входных данных не превосходит 10^5.

Формат выходных данных

Для каждого набора входных данных первая строка должна содержать ответ t — максимальное количество подвязанных стеблей. Следующие t строк должны содержать описание подвязанных стеблей в следующем формате:

В строке должно содержаться два целых числа r_j, c_j — координаты зоны стебля — и литера $d_j \in \{$«u», «r», «d», «l»$\}$, обозначающая направление подвязывания (в соответствии с пояснением на рисунке в условии выше).

Если решений несколько, выведите любое.

Система оценки

В этой задаче каждый тест оценивается независимо. Все тесты можно разделить на следующие группы:

Группа	Макс. балл	Доп. ограничения	Комментарий
		n, m	
0	0	–	Тесты из условия
1	5	$n \leqslant 1$	
2	10	$n \leqslant 2$	
3	10	$n \leqslant 3$	
4	10	$n \cdot m \leqslant 40$	
5	5	–	Комментарий[1]
6	20	$n \cdot m \leqslant 1\,000$	
7	40	–	

Баллы начисляются за прохождение каждого теста.

Комментарий[1]: Для каждой зоны со стеблями гарантируется, что до забора можно дотянуться не более чем в одном из 4 направлений.

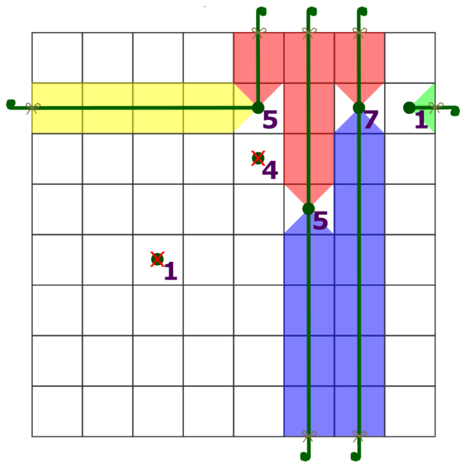
Пример

стандартный ввод	стандартный вывод
2 4 5 1 2 4 9 8 8 6 2 5 5 2 7 7 2 8 1 3 5 4 4 6 5 5 3 1	4 2 4 u 2 4 d 2 4 r 2 4 l 7 2 5 u 2 7 u 4 6 d 2 8 r 2 5 l 4 6 u 2 7 d

Замечание

Первый тестовый случай изображен в условии.

Решение для второго тестового случая изображено ниже.



Обратите внимание, что полностью удаленные стебли не мешают протягиванию стеблей через их зону. Так, например, стебель 2 8 `r` можно заменить на 2 7 `r`, и решение останется корректным. Существуют также другие варианты решения этого тестового случая.