Общая информация по задачам первого тура

Задача	Тип задачи	Ограничения
1. Беспилотная аэрологистика	стандартная	1 с, 512 МБ
2. 2026	стандартная	2 с, 512 МБ
3. Кейс на рейс	стандартная	2 с, 512 МБ
4. Рамазан и капуста	стандартная	4 с, 1024 МБ

Необходимо считывать данные из стандартного потока ввода. Выходные данные необходимо выводить в стандартный поток вывода.

Баллы за подзадачу, если в условии не указано иное, начисляются только если все тесты этой подзадачи пройдены. Решение запускается на тестах для определенной подзадачи, если все тесты всех необходимых подзадач пройдены.

Во всех задачах во всех подзадачах во время тура вам показываются баллы за подзадачу, если все тесты пройдены, либо первая ошибка и номер теста.

Для некоторых подзадач может также требоваться, чтобы были пройдены все тесты из условия. Для таких подзадач указана дополнительно буква У.

Задача 1. Беспилотная аэрологистика

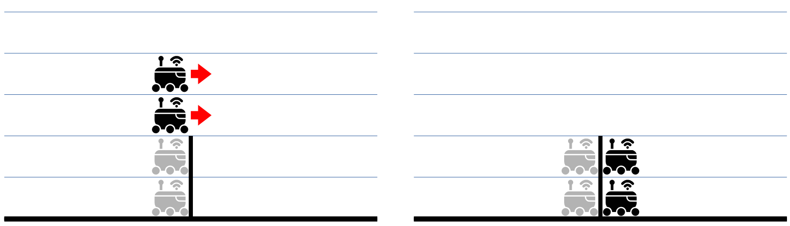
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

На всероссийской олимпиаде по информатике 2224 года, которая проходит в Иннополисе, доставкой занимаются роботы нового поколения, которые способны создавать своих клонов. Доставку можно получить прямо через окно, не выходя из дома.

Изначально есть только один робот-доставщик. В любой момент верхний робот может создать одного или нескольких новых роботов прямо над собой. Так образуется колонна роботов. Высота каждого робота равна высоте одного этажа.

В процессе доставки колонна одинаковых роботов-клонов перемещается вдоль корпусов общежития слева направо. В базе данных у роботов содержится список сделанных заказов, для каждого из которых известно окно, в которое его нужно доставить. Когда колонна роботов проходит мимо окна, соответствующего какому-то заказу, она может произвести доставку, если в колонне есть робот, расположенный на уровне окна.

Во время перемещения конструкция из роботов может натолкнуться на препятствие. После препятствия движение продолжают только те экземпляры роботов, которые находились выше препятствия. Они оказываются на земле непосредственно за препятствием, по прежнему в виде вертикальной колонны, и могут продолжать движение, создавать новых клонов и доставлять заказы.



Расстояние между препятствиями и окнами достаточно большое, поэтому во время переезда через препятствие роботы не будут проезжать мимо окна.

За доставку одного заказа компания-организатор доставки получает p крипторублей. Стоимость создания одного нового робота равна c крипторублей. Итоговая прибыль равна суммарному доходу от доставки заказов за вычетом суммарной стоимости создания всех роботов. Компания хочет максимизировать свою прибыль. При этом, она не обязана выполнить все заказы, а роботы могут в любой момент остановиться, и прекратить процесс доставки.

Определите максимальную прибыль, которую может получить компания.

Формат входных данных

В первой строке входных данных находятся четыре целых числа n, m, c, p ($0 \leqslant n, m \leqslant 100\,000$, $1 \leqslant c, p \leqslant 10^6$) — количество препятствий, количество заказов в базе, стоимость создания клона робота и стоимость доставки одного заказа, соответственно.

В следующих $n + m$ строках идёт описание препятствий и окон, в которые нужно доставить заказы, в порядке следования колонны роботов вдоль общежитий слева направо. Каждая строка содержит два целых числа t_i и h_i ($1 \leqslant t_i \leqslant 2$, $1 \leqslant h_i \leqslant 10^6$) — тип объекта t_i (1 для препятствия и 2 для окна) и h_i — высота препятствия в этажах или этаж, на котором находится окно.

Гарантируется, что ровно n объектов имеют тип 1, и оставшиеся m объектов имеют тип 2.

Формат выходных данных

Выведите одно число — максимальную величину прибыли, которую можно получить.

Система оценки

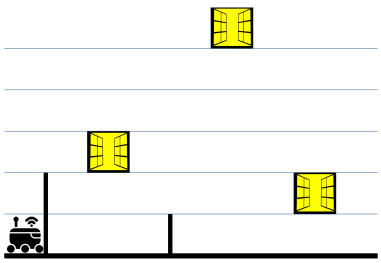
Подз.	Баллы	Ограничения			Необх. подзадачи
		n	m	дополнительно	
1	24	$n \leqslant 100$	$m \leqslant 100$	$h_i \leqslant 100$	
2	12	$n = 0$			
3	14	$n = 1$			
4	15		$m = 1$		
5	17			$c = 1$, $p = 10^6$ высоты всех препятствий равны 1	
6	18				У, 1 – 5

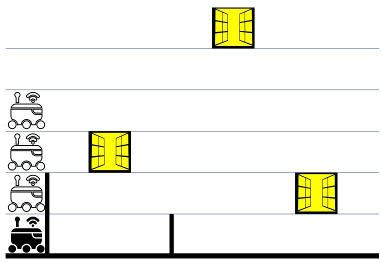
Примеры

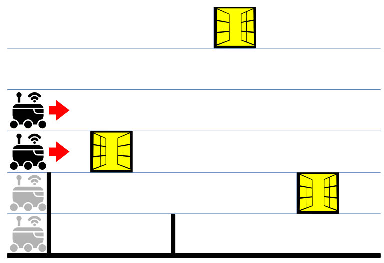
стандартный ввод	стандартный вывод
2 3 2 6 1 2 2 3 1 1 2 6 2 2	4
1 3 1 5 2 2 2 1 1 9 2 1	9

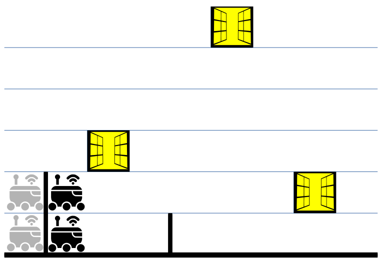
Пояснения к примерам

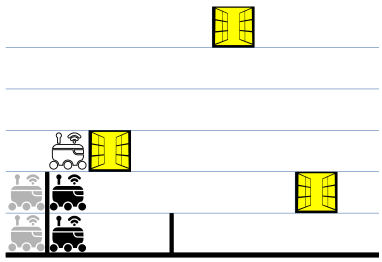
Одна из оптимальных стратегий доставки заказов из первого примера изображена на девяти рисунках ниже, при этом выполнение второго заказа не увеличивает прибыль.

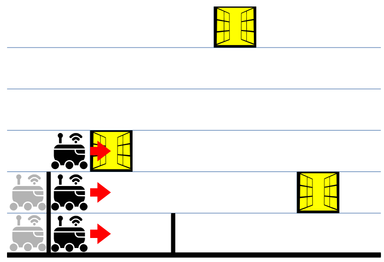

(1)

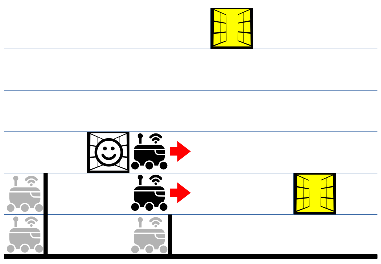

(2)

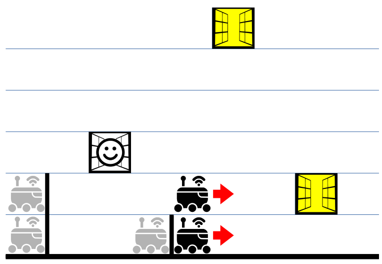

(3)

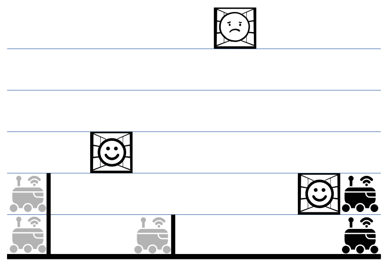

(4)


(5)


(6)


(7)


(8)


(9)

Во втором примере достаточно один раз клонировать робота для доставки первого заказа, полученной системой роботов доставить второй заказ, а производить дополнительное клонирование для доставки третьего заказа экономически невыгодно.

Задача 2. 2026

Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Новая татарская игра «2026» ведется на прямоугольной клетчатой доске, состоящей из m строк и n столбцов. Доска разбита на $m \times n$ единичных клеток размером 1×1. На некоторых клетках стоят квадратные фишки размером 1×1, на каждой фишке написана одна из 26 английских букв.

С фишками производятся q операций. Каждая операция состоит в перемещении всех фишек до упора в одном из четырех направлений. Таким образом, последовательность операций задается строкой s длины q, состоящей из символов, соответствующих направлениям: «`L`» — влево, «`R`» — вправо, «`U`» — вверх и «`D`» — вниз.

Операция выполняется следующим образом: пока на доске есть хотя бы одна фишка, для которой соседняя с ней в заданном направлении клетка является свободной, эта фишка передвигается на эту соседнюю клетку.

Определите, как будет выглядеть доска после выполнения всех операций.

Формат входных данных

Каждый тест состоит из нескольких наборов входных данных. В первой строке теста задано целое число t — количество наборов входных данных в тесте ($1 \leqslant t \leqslant 200\,000$). Далее следуют описания наборов входных данных. Каждый набор входных данных описывается следующим образом:

В первой строке набора заданы целые числа m и n — размеры доски ($1 \leqslant m, n \leqslant 10^6$, $1 \leqslant m \times n \leqslant 10^6$).

В следующих m строках задано изначальное расположение фишек на доске.

В i-й строке ($1 \leqslant i \leqslant m$) находится строка $a_{i1}a_{i2}\ldots a_{in}$ длины n, задающая i-ю строку доски. Каждый символ a_{ij} является либо строчной буквой английского алфавита от «`a`» до «`z`», либо точкой «`.`». Если $a_{ij} =$ «`.`», то клетка в i-й строке и j-м столбце является пустой, иначе в ней находится фишка, на которой написана буква a_{ij}.

В последней строке заданы q символов $s_1s_2\ldots s_q$ без пробелов, задающие последовательность операций ($1 \leqslant q \leqslant 10^6$). Каждый символ s_i является одним из символов «`L`», «`R`», «`U`» или «`D`».

Сумма значений $m \times n$ по всем наборам входных данных не превышает $2 \cdot 10^6$. Сумма значений q по всем наборам входных данных не превышает $2 \cdot 10^6$.

Формат выходных данных

Для каждого набора входных данных выведите итоговое расположение фишек на доске после выполнения всех операций в том же формате, что и во входных данных.

Система оценки

Обозначим через $\sum mnq$ сумму mnq по всем наборам входных данных.

Обозначим через $\sum mq$ сумму mq по всем наборам входных данных.

Назовем расположение фишек *лестницей*, если $m = n$, $a_{ij} =$ «.» для всех $1 \leqslant i \leqslant j \leqslant n$ и $a_{ij} \neq$ «.» для всех $1 \leqslant j < i \leqslant n$. Иными словами, все фишки находятся на клетках ниже главной диагонали доски, и на каждой клетке ниже главной диагонали есть фишка.

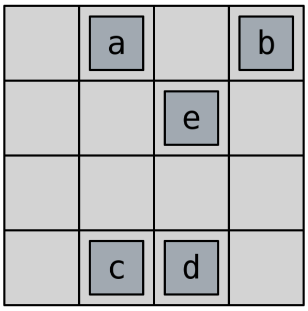
Подзадача	Баллы	Дополнительные ограничения	Необх. подзадачи
1	9	$t = 1$, $q = 1$, $n, m \leqslant 100$	—
2	7	$s_i \neq$ «D», $s_i \neq$ «U»	—
3	13	$\sum mnq \leqslant 10^7$	1
4	14	$s_i \neq$ «D»	2
5	12	На всех фишках написана буква «a», $\sum mq \leqslant 10^7$	—
6	11	На всех фишках написана буква «a»	5
7	9	Изначальное расположение фишек образует *лестницу*	—
8	14	s является строкой «LURD», повторенной несколько раз	—
9	11		1–8

Пример

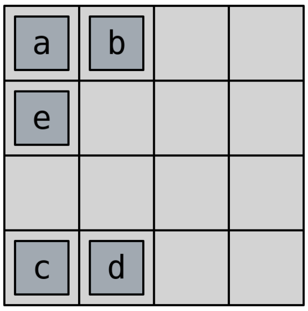
стандартный ввод	стандартный вывод
4 4 4 .a.b ..e. .... .cd. LRU 1 1 . UULLRRDD 1 6 .a.aa. LLURDDD 5 7 .ba.b.. ac..c.d e...... ....da. d.eae.. DLDDRULRRR	..ab ..ce ...d ....aaa dceebab ...aeac .....ad ......d ....

Пояснения к примерам

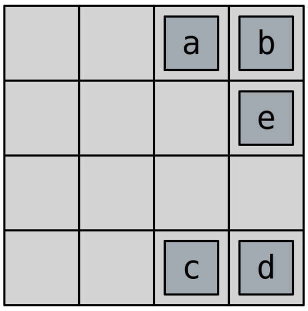
В первом наборе входных данных из примера доска изначально выглядит так:



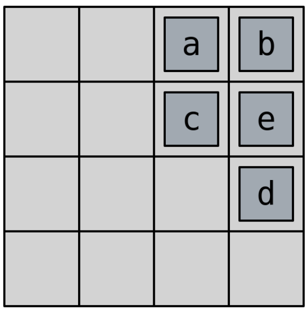
Первая операция сдвигает все фишки влево, так как $s_1 =$ «L». После ее выполнения доска будет выглядеть следующим образом:



Вторая операция сдвигает все фишки вправо, так как $s_2 =$ «R». После ее выполнения доска будет выглядеть следующим образом:



Третья и последняя операция сдвигает все фишки наверх, так как $s_3 =$ «U». После ее выполнения доска будет выглядеть следующим образом:



Задача 3. Кейс на рейс

Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Авиакомпания «Флагманский Флот Татарстана» предлагает в своих самолётах новый вид бизнес-класса. Салон самолёта состоит из n мест, расположенных в один ряд вдоль прохода. Введём координатную прямую вдоль салона так, что расстояние между креслами будет равно 1, и места будут иметь координаты от 1 до n.

Во время полёта стюарду нужно пройти по самолету и раздать всем пассажирам напитки. Напитки бывают k разных видов, пронумерованных числами от 1 до k. Каждый пассажир получает одну порцию одного напитка, пассажир заказывает предпочитаемый вид напитков при бронировании билета, поэтому все предпочтения пассажиров известны заранее.

Напитки разлиты по бутылкам, каждая бутылка вмещает p порций одного напитка. В тележку для напитков можно загрузить не более m бутылок с любыми видами напитков, гарантируется, что $m \geqslant k$.

Пассажиры обслуживаются в порядке возрастания номеров их мест. Первоначально тележка находится в начале салона в точке 0, и её можно заполнить любыми видами напитков перед обслуживанием. После завершения обслуживания тележка должна приехать в точку $n+1$. При этом в точках 0 и $n+1$ могут находиться кладовые: или одна кладовая в одном из концов салона или две кладовые в двух концах, в которых имеется достаточный запас напитков каждого вида. В этих кладовых можно выгрузить из тележки пустые бутылки и погрузить полные бутылки.

По ходу обслуживания напитки будут расходоваться, поэтому время от времени возникает необходимость пополнить запас напитков на тележке в одной из кладовых. Если в текущий момент тележка находится напротив кресла номер i, то для того, чтобы доехать до кладовой в точке 0 необходимо проехать расстояние i, а для того, чтобы доехать до кладовой в точке $n+1$ необходимо проехать расстояние $n+1-i$. В кладовых можно выгрузить пустые бутылки из тележки и загрузить на свободные места бутылки с напитками любых видов. Выгружаемые бутылки должны быть пустыми, нельзя выгружать бутылки, в которых остались напитки, или выливать напитки. Нельзя переливать остатки напитков между разными бутылками. Можно загружать на тележку более одной бутылки одного вида. После этого тележка должна проехать расстояние от кладовой до кресла первого необслуженного пассажира, чтобы продолжить обслуживание.

Определите, какое минимальное расстояние должна проехать тележка, чтобы переместиться из точки 0 в точку $n+1$ и обслужить всех пассажиров.

Формат входных данных

Первая строка входных данных содержит четыре целых числа n, m, k, p ($3 \leqslant n \leqslant 10^6$, $1 \leqslant p \leqslant 10^6$, $1 \leqslant k \leqslant m \leqslant 10^6$) — количество мест в салоне, вместимость тележки, количество типов напитков и вместимость каждой бутылки соответственно.

В следующей строке содержится целое число c ($1 \leqslant c \leqslant 3$) — параметр, описывающий наличие кладовых в салоне. Если $c = 1$, то кладовая находится только в точке $n+1$. Если $c = 2$, то кладовая находится только в точке 0. Если $c = 3$, то кладовые находятся в обоих концах салона.

В следующей строке содержатся n целых чисел a_i ($1 \leqslant a_i \leqslant k$) — типы напитков, которые заказали пассажиры.

Формат выходных данных

Программа должна вывести одно целое число — минимальное расстояние, которое должна проехать тележка.

Система оценки

Подзадача	Баллы	Ограничения			Необх. подзадачи
		c	n	Дополнительные ограничения	
1	5	$c = 1$	$n \leqslant 15$	$k \leqslant 15$	
2	8	$c = 1$	$n \leqslant 2000$		1
3	4	$c = 1$	—	$p = 1$	
4	7	$c = 1$	—		1, 2, 3
5	8	$c = 2$	$n \leqslant 15$	$k \leqslant 15$	
6	10	$c = 2$	$n \leqslant 2000$		5
7	6	$c = 2$	—	$p = 1$	
8	9	$c = 2$	—		5, 6, 7
9	10	$c = 3$	$n \leqslant 15$	$k \leqslant 15$	
10	13	$c = 3$	$n \leqslant 2000$		9
11	9	$c = 3$	—	$p = 1$	
12	11	$c = 3$	—		9, 10, 11

Примеры

стандартный ввод	стандартный вывод

`5 2 2 1`
`1`
`1 2 1 2 1`
	`14`

`8 3 2 2`
`2`
`1 1 1 1 1 2 2 2`
	`17`

`8 3 3 2`
`3`
`1 2 2 3 2 3 2 1`
	`15`

`8 6 6 2`
`2`
`1 2 3 4 3 5 6 1`
	`9`

`7 3 3 1`
`3`
`1 2 3 2 2 1 3`
	`16`

Пояснения к примерам

В первом примере в тележку вмещается $m = 2$ бутылки по $p = 1$ порции в каждой. Кладовая находится в конце салона. Первоначально тележку нужно загрузить бутылками с напитками вида 1 и 2, которые будут налиты пассажирам на местах 1 и 2, тележка проедет расстояние 2 от точки 0 до точки 2. После этого тележке нужно будет проехать до кладовой в конце салона (расстояние 4), загрузить тележку бутылками вида 1 и 2 и вернуться к креслу номер 3 (тележка проедет расстояние 3). Пассажирам на местах 3 и 4 выдаются напитки вида 1 и 2 (тележка проезжает расстояние 1 от места 3 до места 4). После этого тележке понадобится ещё раз съездить в кладовую (от кресла 4 до кладовой расстояние 2), вернуться из кладовой до кресла 5 (расстояние 1), и проехать ещё 1 до конца салона. Общее расстояние равно $2 + 4 + 3 + 1 + 2 + 1 + 1 = 14$.

Во втором примере в тележку вмещаются $m = 3$ бутылки по $p = 2$ порции в каждой. Кладовая находится в начале салона. Необходимо загрузить тележку тремя бутылками вида 1, обслужить пассажиров на местах с номерами от 1 до 4. После этого опустошатся две бутылки вида 1, нужно будет сразу съездить в кладовую, чтобы загрузить две бутылки вида 2, затем обслужить пассажиров на местах с номерами от 5 до 8.

В третьем примере в тележку вмещаются $m = 3$ бутылки по $p = 2$ порции в каждой, кладовые находятся в обоих концах салона. Для обслуживания пассажиров нужны две бутылки вида 2 и по одной бутылке видов 1 и 3, поэтому понадобится один раз съездить в кладовую для того, чтобы заменить пустую бутылку вида 2 на полную. Это лучше сделать после обслуживания пассажира на месте 3, тележка должна съездить в кладовую в начале салона.

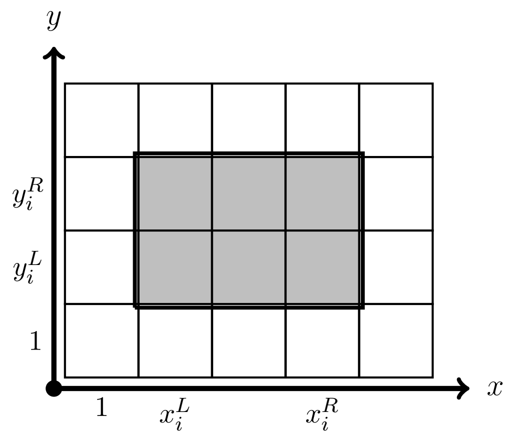
В четвёртом примере в тележку нужно загрузить по одной бутылке каждого вида, и поскольку каждая бутылка вмещает по две порции напитков, это позволит обслужить всех пассажиров без дополнительного пополнения тележки.

В пятом примере понадобится два пополнения тележки, один раз тележке придётся вернуться в кладовую в начало салона после обслуживания пассажира 3, второй раз — в конец салона после обслуживания пассажира 6.

Задача 4. Рамазан и капуста

Ограничение по времени: 4 секунды
Ограничение по памяти: 1024 мегабайта

Рамазан решил заняться серьезным бизнесом — выращиванием капусты.



Поле для выращивания капусты представляет собой бесконечное клетчатое поле. В каждой клетке поля может быть посажен один кочан капусты.

Рамазан засадил только часть поля. Он запланировал использовать несколько прямоугольных участков поля, причём оказалось, что некоторые из них могут пересекаться. Клетка поля принадлежит посадкам, если она лежит хотя бы в одном из прямоугольников.

Формально, Рамазан выбрал n прямоугольных участков $(x_i^L, y_i^L, x_i^R, y_i^R)$ $(x_i^L \leqslant x_i^R,\ y_i^L \leqslant y_i^R,\ 1 \leqslant i \leqslant n)$. Клетка (x, y) содержит капусту, если существует хотя бы один выбранный прямоугольник i $(1 \leqslant i \leqslant n)$, такой что $x_i^L \leqslant x \leqslant x_i^R$ и $y_i^L \leqslant y \leqslant y_i^R$.

В прошлом Рамазан был программистом (и победителем), поэтому он решил использовать роботов с искусственным интеллектом для периодической обработки посадок. Один робот может обслуживать произвольный горизонтальный участок клеток $(x_1^{robot}, x_2^{robot}, y^{robot})$, то есть все клетки (x, y), такие что $x_1^{robot} \leqslant x \leqslant x_2^{robot}$ и $y = y^{robot}$.

Важно, чтобы роботы ездили только по участкам с посадками. Он понял, что для минимизации количества роботов важно использовать горизонтальные участки, которые нельзя расширить. Рамазан будет использовать робота на участке клеток $(x_1^{robot}, x_2^{robot}, y^{robot})$, если:

- Все клетки (x, y), такие что $x_1^{robot} \leqslant x \leqslant x_2^{robot}$ и $y = y^{robot}$ принадлежат посадкам;
- Клетка $(x_1^{robot} - 1, y^{robot})$ не принадлежит посадкам;
- Клетка $(x_2^{robot} + 1, y^{robot})$ не принадлежит посадкам.

Ваша задача собрать важную статистику о роботах, которые будут работать на плантации. Будем говорить, что пара (x_1, x_2) *обслуживается* в ряду y, если существует робот, работающий ровно на участке (x_1, x_2, y).

- Найдите все пары (x_1, x_2), которые обслуживаются в каком-нибудь ряду.
- Для каждой такой пары (x_1, x_2) найдите количество рядов, в которых она обслуживается.
- Для каждой такой пары (x_1, x_2) найдите максимальное количество подряд идущих рядов, в которых она обслуживается. Другими словами, найдите максимальное число k, такое что существует отрезок k подряд идущих рядов $[y_1, y_2]$ $(y_2 - y_1 + 1 = k)$, такой что для любого ряда $y_1 \leqslant y \leqslant y_2$, пара (x_1, x_2) обслуживается в ряду y.

Формат входных данных

Каждый тест состоит из нескольких наборов входных данных. В первой строке дано одно целое число t $(1 \leqslant t \leqslant 200\,000)$ — количество наборов входных данных. Далее следуют описания наборов входных данных.

В первой строке каждого набора входных данных дано единственное целое число n $(1 \leqslant n \leqslant 200\,000)$ — количество выбранных прямоугольных участков.

В следующих n строках дано по четыре целых числа x_i^L, y_i^L, x_i^R, y_i^R $(1 \leqslant x_i^L \leqslant x_i^R \leqslant 10^9$, $1 \leqslant y_i^L \leqslant y_i^R \leqslant 10^9)$ — описания выбранных прямоугольных участков.

Обозначим за N сумму n по всем наборам входных данных в одном тесте. Гарантируется, что $N \leqslant 200\,000$.

Формат выходных данных

Для каждого набора входных данных сначала выведите единственное целое число p ($p \geqslant 1$) — количество пар (x_1, x_2), которые обслуживаются в каком-нибудь ряду.

В следующих p строках выведите по четыре целых числа x_1, x_2, cnt, k ($1 \leqslant x_1 \leqslant x_2 \leqslant 10^9$, $0 \leqslant cnt, k \leqslant 10^9$). Число cnt должно быть равно количеству рядов, в которых обслуживается пара (x_1, x_2). Число k должно быть равно максимальному количеству подряд идущих рядов, в которых обслуживается пара (x_1, x_2).

Все пары (x_1, x_2) должны быть различны. Каждая пара, которая обслуживается в каком-нибудь ряду, должна быть выведена ровно один раз. Можно вывести пары в произвольном порядке.

Система оценки

Для набора входных данных обозначим за w ширину поля, то есть $w = \max\limits_{i=1}^{n} x_i^R$, за h высоту поля, то есть $h = \max\limits_{i=1}^{n} y_i^R$.

Подз.	Баллы	Ограничения: n, N	Ограничения: w, h	Ограничения: дополнительно	Необх. подзадачи
1	4	$n = 1$			
2	8		$h = 1$		
3	8	$n \leqslant 30$, $N \leqslant 3000$	$w, h \leqslant 10$	$t \leqslant 100$	У
4	4		$w, h \leqslant 5000$, $\sum wh \leqslant 25 \cdot 10^6$		У, 3
5	8	$N \leqslant 3000$			У, 3
6	4	$N \leqslant 10\,000$			У, 3, 5
7	8			все $[x_i^L, x_i^R]$ пересекаются	1
8	8			$y_i^L = 1$	2
9	8			прямоугольники не пересекаются	1
10	8			$\forall 1 \leqslant i, j \leqslant n$ $\forall y \in [y_i^L, y_i^R] \cap [y_j^L, y_j^R]$ выполнено $[x_i^L, x_i^R] \not\subseteq [x_j^L, x_j^R]$	1, 9
11	8			все отрезки $[x_i^L, x_i^R + 1]$ либо вложены, либо не пересекаются	1
12	8	$N \leqslant 50\,000$			У, 3, 5 – 6
13	8	$N \leqslant 100\,000$			У, 3, 5 – 6, 12
14	8	$N \leqslant 200\,000$			У, 1 – 13

- Если для теста ваше решение неправильно находит множество пар (x_1, x_2), которые обслуживаются в каком-нибудь ряду, решение получает вердикт «Неправильный ответ».
- Если во всех тестах подзадачи и необходимых подзадач решение

– правильно находит множество, но не все cnt верны, оно получает 50% баллов за подзадачу.
– правильно находит множество и все cnt, но не все k верны, оно получает 75% баллов за подзадачу.
– правильно находит множество, все cnt и все k, оно получает 100% баллов за подзадачу.

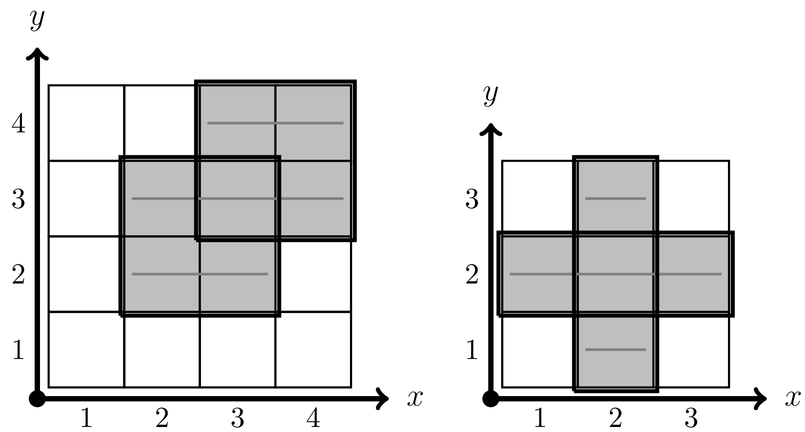
Обратите внимание, что для получения частичных баллов за подзадачу, все равно необходимо вывести какие-нибудь значения cnt и k для каждой пары (x_1, x_2), но не обязательно верные.

Пример

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

Пояснения к примерам

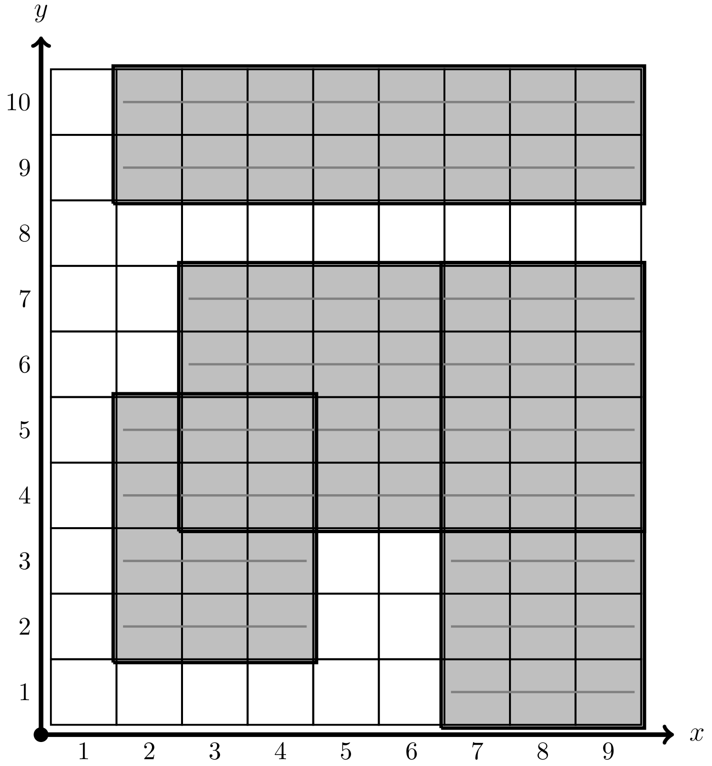
Первый и второй наборы входных данных для теста из условия

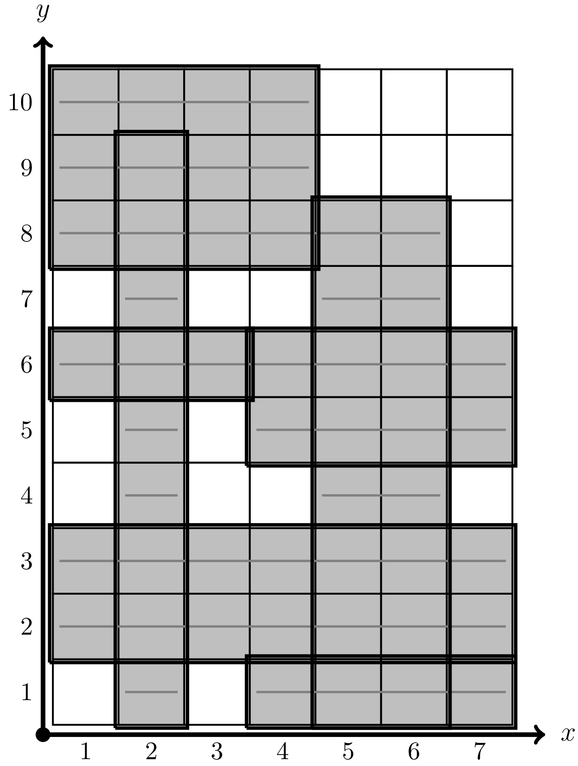


В первом наборе входных данных будут использоваться роботы на участках $(2, 3, 2)$, $(2, 4, 3)$, $(3, 4, 4)$. Таким образом, пары $(2, 3)$, $(2, 4)$, $(3, 4)$ обслуживаются в каком-нибудь ряду, причем каждая из них обслуживается ровно в одном ряду.

Во втором наборе входных данных будут использоваться роботы на участках $(2, 2, 1)$, $(2, 4, 2)$, $(2, 2, 3)$. Таким образом, пары $(2, 2)$, $(2, 4)$ обслуживаются в каком-нибудь ряду. Пара $(2, 2)$ обслуживается в рядах $1, 3$, пара $(2, 4)$ обслуживается ряду 2.

Третий и четвертый наборы входных данных для теста из условия





Общая информация по задачам второго тура

Задача	Тип задачи	Ограничения
5. Восстание газонокосилок	стандартная	1 с, 512 МБ
6. Интерактивные переходы	стандартная	1 с, 512 МБ
7. Гонка дронов	стандартная	2 с, 512 МБ
8. За связь без перебоев	стандартная	3 с, 512 МБ

Необходимо считывать данные из стандартного потока ввода. Выходные данные необходимо выводить в стандартный поток вывода.

Баллы за подзадачу, если в условии не указано иное, начисляются только если все тесты этой подзадачи пройдены. Решение запускается на тестах для определенной подзадачи, если все тесты всех необходимых подзадач пройдены.

Во всех задачах во всех подзадачах во время тура вам показываются баллы за подзадачу, если все тесты пройдены, либо первая ошибка и номер теста.

Для некоторых подзадач может также требоваться, чтобы были пройдены все тесты из условия. Для таких подзадач указана дополнительно буква У.

Задача 5. Восстание газонокосилок

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Газоны в Иннополисе косят электрические роботы-газонокосилки. Будем считать, что газон представляет собой отрезок числовой прямой, на котором в некоторых точках расположены роботы-газонокосилки. Размером роботов можно пренебречь. Один из роботов стоит в начале газона (левее него газона нет), и один — в конце (правее него газона нет). Каждый робот изначально ориентирован в одном из двух направлений: либо направо, либо налево.

Заряда i-го робота хватает для обработки p_i метров газона. После ночной зарядки все роботы начинают работать одновременно и движутся с одинаковой скоростью. Каждый робот движется в своём направлении вдоль прямой. Робот останавливается в одном из трёх случаев:

1. Если у робота закончился заряд. Иными словами, если i-й робот проехал p_i метров от точки старта.
2. Если робот доехал до начала или конца газона.
3. Если робот встретился в одной точке с другим роботом, который двигался ему навстречу или остановился в этой точке ранее.

Перед запуском роботов вы можете поменять направление некоторых из них на противоположное. Требуется скосить траву на всём газоне.

Определите, у какого минимального количества роботов нужно поменять направление, чтобы в итоге вся трава на газоне оказалась скошена. Иначе сообщите, что всю траву скосить невозможно.

Формат входных данных

В первой строке содержится целое число n ($2 \leqslant n \leqslant 10^5$) — количество роботов.

В следующих n строках содержатся описания роботов в порядке их расположения на прямой слева направо. Каждый робот характеризуется тремя целыми числами x_i, p_i, d_i: начальной позицией робота, количеством метров, которые он может проехать, и направлением движения ($0 = x_1 < x_2 < \ldots < x_n \leqslant 10^9$, $1 \leqslant p_i \leqslant 10^9$, значение $d_i = -1$ обозначает движение налево, в направлении уменьшения координаты, $d_i = 1$ обозначает движение направо, в направлении увеличения координаты). Начало и конец газона находятся в точках $x_1 = 0$ и x_n соответственно.

Формат выходных данных

В единственной строке необходимо вывести -1, если скосить всю траву на газоне невозможно. Иначе, нужно вывести одно число — количество роботов, у которых нужно изменить направление на противоположное, чтобы газон был скошен.

Система оценки

Подз.	Баллы	Ограничения		Необх. подзадачи
		n	дополнительно	
1	23	$n \leqslant 10$		У
2	16		изначально все роботы ориентированы направо ($d_i = 1$)	
3	17	$n \leqslant 1000$		У, 1
4	13		$x_i = i - 1$, $p_i = 1$	
5	14		$p_i = 10^9$	
6	17		без дополнительных ограничений	У, 1 – 5

Примеры

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

Пояснения к примерам

Первый пример изображен на рисунке. Для того, чтобы скосить всю траву, можно, например, развернуть робота, который стоит посередине.

Задача 6. Интерактивные переходы

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Кампус Иннополиса состоит из n корпусов, соединённых m переходами. Каждый переход соединяет два различных корпуса, никакие два корпуса не соединены более чем одним переходом.

Известно, что каждый корпус имеет подсветку, которая может быть включена или выключена. Изначально подсветка всех корпусов выключена. Диспетчер кампуса может за одно *действие* включить или выключить подсветку любого корпуса. Диспетчер может также нажимать кнопку включения подсветки, если она уже включена, или нажимать кнопку выключения, если она выключена. Данные действия не приводят к изменению состояния подсветки корпуса.

Аналогично, каждый переход имеет подсветку, которая может быть включена или выключена. Исходно подсветка всех переходов выключена. Однако в отличие от подсветки корпусов, подсветка переходов изменяется автоматически: если после очередного действия диспетчера состояние подсветки соединённых переходом корпусов оказывается одинаковым, то подсветка перехода также переходит в это состояние, а иначе она не меняется.

Другими словами, если после очередного действия диспетчера подсветка обоих корпусов, соединённых переходом, оказывается выключена, то подсветка перехода также выключается. Если подсветка обоих корпусов, соединённых переходом, оказывается включена, то подсветка перехода также включается. Если подсветка одного из корпусов оказывается включена, а другого — выключена, то состояние подсветки перехода не меняется.

Перед приездом участников олимпиады по информатике директор кампуса для каждого корпуса и каждого перехода определил, должен ли этот корпус или переход быть подсвечен.

Проверьте, может ли диспетчер выполнить пожелание директора, выполнив произвольное число действий. Если это возможно, то найдите любую такую последовательность действий. Решения, корректно определяющие возможность получить желаемое состояние подсветки, но не предъявляющие искомую последовательность действий, будут получать частичные баллы.

Формат входных данных

Каждый тест содержит один или несколько наборов входных данных. В первой строке теста задано целое число t — количество наборов входных данных в тесте ($1 \leqslant t \leqslant 50\,000$). Далее следуют описания наборов. Каждый набор описывается следующим образом.

В первой строке заданы целые числа: n — количество корпусов, и m — количество переходов ($1 \leqslant n \leqslant 10^5$, $0 \leqslant m \leqslant 2 \cdot 10^5$).

В следующих m строках следуют описания переходов.

В i-й строке находятся целые числа a_i, b_i, c_i — номера корпусов, соединённых i-м переходом, и требуемое состояние подсветки i-го перехода, соответственно ($1 \leqslant a_i, b_i \leqslant n$, $a_i \neq b_i$, $0 \leqslant c_i \leqslant 1$). Если $c_i = 0$, то подсветка i-го перехода в результате должна быть выключена, а если $c_i = 1$, то включена.

В последней строке заданы n целых чисел $d_1, d_2, \ldots, d_n$ — требуемое состояние подсветки корпусов ($0 \leqslant d_i \leqslant 1$). Если $d_v = 0$, подсветка корпуса v в итоге должна быть выключена, а если $d_v = 1$, то включена.

Сумма значений n по всем наборам входных данных не превышает 10^5. Сумма значений m по всем наборам входных данных не превышает $2 \cdot 10^5$.

Формат выходных данных

Для каждого набора входных данных:

- Если не существует последовательности действий, получающей требуемое состояние подсветки корпусов и переходов, то выведите «NO».
- Если последовательность действий существует, то выведите «YES». Если вы не хотите предъявлять саму последовательность действий, то выведите в следующей строке число -1 и перейдите к следующему набору входных данных. Если вы хотите предъявить последовательность действий, то выведите в следующей строке целое число s — количество действий ($0 \leqslant s \leqslant 10^6$,

сумма значений s по всем наборам входных данных не должна превышать 10^6), а в следующих s строках выведите сами действия.

В i-й строке ($1 \leqslant i \leqslant s$) выведите два целых числа: v_i — номер корпуса, в котором изменяется состояние подсветки, и x_i — новое состояние подсветки ($1 \leqslant v_i \leqslant n$, $0 \leqslant x_i \leqslant 1$, если $x_i = 0$, то подсветка корпуса v_i выключается, если $x_i = 1$, то подсветка корпуса v_i включается).

Система оценки

Обозначим за N сумму значений n во всех наборах входных данных в одном тесте, за M — сумму значений m во всех наборах тестовых данных в одном тесте.

Если решение выводит неправильную последовательность действий на одном из тестов подзадачи, то оно получает 0 баллов за подзадачу. Если на хотя бы одном тесте подзадачи решение выводит -1 и на каждом тесте подзадачи выводит либо верную последовательность действий, либо -1, то оно получает половину баллов за подзадачу. Если на каждом тесте подзадачи решение выводит верную последовательность действий, то оно получает полный балл за подзадачу.

Подз.	Баллы	Ограничения			Необх. подзадачи
		N, n	M, m	Дополнительные ограничения	
1	4	$n \leqslant 3$	—	$t \leqslant 230$	
2	10	$N \leqslant 2000$	$M \leqslant 2000$	$n + m \leqslant 14$	
3	8	—	—	$c_i = 1$	
4	6	—	—	$m = n - 1$, $a_i = 1$, $b_i = i + 1$	
5	6	—	—	$d_{a_i} = c_i$, $a_i < b_i$	
6	8	$N \leqslant 2000$	—	$m = n - 1$, $a_i = i$, $b_i = i + 1$	
7	8	—	—	$m = n - 1$, $a_i = i$, $b_i = i + 1$	6
8	10	$N \leqslant 2000$	—	$m = n - 1$, из любого корпуса можно добраться до любого другого по переходам	6
9	6	—	—	$m = n - 1$, из любого корпуса можно добраться до любого другого по переходам	4, 6–8
10	6	—	—	$m = n$, $a_i = i$, $b_i = i\%n + 1$	
11	10	$N \leqslant 2000$	$M \leqslant 2000$	—	2, 6, 8
12	18	—	—	—	1–11

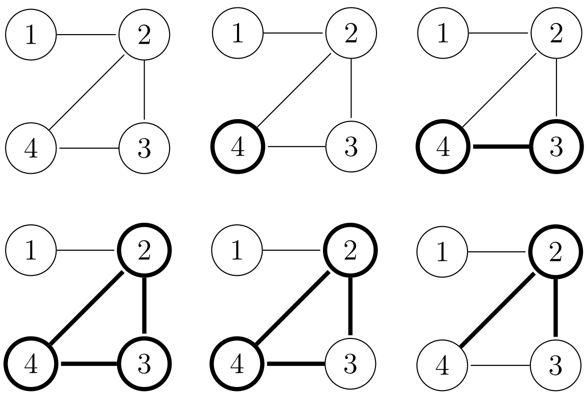
Пример

стандартный ввод	стандартный вывод
5 4 4 1 2 0 2 3 1 3 4 0 2 4 1 0 1 0 0 4 4 1 2 0 2 3 1 3 4 0 4 1 1 0 1 0 1 1 0 1 1 0 0 2 1 1 2 0 0 0	YES 5 4 1 3 1 2 1 3 0 4 0 NO YES 1 1 1 YES 1 1 0 YES 0

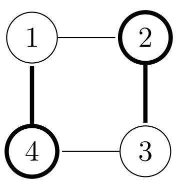
Пояснения к примерам

В примере из условия пять тестовых наборов данных.

В первом наборе даны 4 корпуса, обозначенных кружками, и 4 перехода, обозначенных линиями. Наличие подсветки корпуса или перехода обозначено жирной линией. Получить нужную подсветку можно за 5 действий. На рисунках ниже изображены начальное состояние подсветки и состояние после каждого действия.



Во втором наборе нужно получить следующую конфигурацию из 4 корпусов и 4 переходов, но сделать это невозможно.



В третьем наборе один корпус, в котором нужно включить подсветку. Это можно сделать за одно действие.

В четвёртом примере один корпус, в котором подсветка должна быть выключена. Возможной последовательностью действий является единственное действие выключения подсветки в корпусе. Это является корректной действием, несмотря на то, что подсветка уже была выключена.

В пятом примере два корпуса и один переход, подсветка везде должна быть выключена. Пустая последовательность действий является корректной последовательностью, приводящей к такой конфигурации.

Задача 7. Гонка дронов

Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

В Иннополисе проводятся гонки дронов.

В гонке могут принять участие n дронов, i-й дрон пролетает единицу расстояния за t_i секунд. Гонка проводится на прямой, на которой расположены m ворот, пронумерованных от 1 до m, i-е ворота находятся на расстоянии s_i от стартовой позиции гонки.

В гонке примут участие первые k дронов с номерами от 1 до k. Величину k судьи объявляют непосредственно перед гонкой, поэтому вам необходимо проанализировать гонку для всех k от 1 до n.

Гонка проводится следующим образом.

Дроны начинают движение из точки 0 в сторону ворот, каждый со своей скоростью. У каждого дрона есть *точка восстановления* — последние ворота, в которых он выполнял *сохранение позиции*. Изначально точка восстановления каждого дрона — точка 0. Дроны каждый раз начинают двигаться из своих точек восстановления и продолжают движение, пока один или несколько дронов не оказываются в точке, где расположены ворота (возможно, различные для разных дронов). В этот момент среди всех дронов, которые оказались в каких-либо воротах, выбирается дрон с наименьшим номером. Для этого дрона производится сохранение позиции, его точка восстановления переносится в его текущую позицию. Остальные дроны мгновенно телепортируются в свои точки восстановления. После этого гонка продолжается таким же образом.

Как только дрон сохраняет позицию в последних воротах с номером m, он финиширует. Не финишировавшие пока дроны, как обычно, телепортируются в свои точки восстановления и продолжают гонку. Когда все дроны финишируют, гонка завершается.

Телепортация — очень энергоемкий процесс. Для подготовки к гонке необходимо понять, сколько суммарно телепортаций совершат все дроны до её завершения. Обозначим как c_k суммарное число телепортаций, которое совершат все дроны, если в гонке будут участвовать первые k дронов. Найдите значения $c_1, c_2, \ldots, c_n$.

Формат входных данных

В первой строке даны два целых числа n и m — количество дронов и ворот, соответственно ($2 \leqslant n \leqslant 150\,000$, $1 \leqslant m \leqslant 150\,000$).

Во второй строке даны n положительных целых чисел $t_1, t_2, ..., t_n$, где t_i — количество секунд, за которое i-й дрон пролетает единицу расстояния ($1 \leqslant t_i \leqslant 10^9$).

В третьей строке даны m положительных целых чисел $s_1, s_2, ..., s_m$, где s_i — позиция i-х ворот на прямой ($1 \leqslant s_1 < s_2 < \ldots < s_m \leqslant 150\,000$).

Формат выходных данных

Выведите n целых чисел $c_1, c_2, \ldots, c_n$.

Система оценки

Подз.	Баллы	Ограничения				Необх. подзадачи
		n	m	t_i, s_i	Доп. ограничения	
1	5	$n = 2$	$m \leqslant 50$	$t_i, s_i \leqslant 100\,000$		
2	7	$n \leqslant 50$	$m \leqslant 50$	$t_i, s_i \leqslant 100\,000$		У, 1
3	13	$n \leqslant 1000$	$m \leqslant 5$	$t_i, s_i \leqslant 100\,000$		У
4	9	$n \leqslant 100\,000$	$m \leqslant 100\,000$	$t_i, s_i \leqslant 100\,000$	$s_{i+1} - s_i = s_1$ для всех $1 \leqslant i < m$	
5	8	$n \leqslant 100\,000$	$m \leqslant 100\,000$	$t_i, s_i \leqslant 100\,000$	все t_i равны	
6	10	$n \leqslant 100$	$m \leqslant 100\,000$	$t_i, s_i \leqslant 100\,000$		У, 1 – 2
7	5	$n \leqslant 100\,000$	$m \leqslant 100\,000$	$t_i \leqslant 2,\ s_i \leqslant 100\,000$		
8	7	$n \leqslant 100\,000$	$m = 2$	$t_i, s_i \leqslant 100\,000$		
9	6	$n \leqslant 10\,000$	$m \leqslant 100\,000$	$t_i, s_i \leqslant 100\,000$		У, 1 – 3, 6
10	6	$n \leqslant 50\,000$	$m \leqslant 100\,000$	$t_i, s_i \leqslant 100\,000$		У, 1 – 3, 6, 9
11	8	$n \leqslant 100\,000$	$m \leqslant 100\,000$	$t_i, s_i \leqslant 100\,000$		У, 1 – 10
12	8	$n \leqslant 100\,000$				У, 1 – 11
13	8	без дополнительных ограничений				У, 1 – 12

Примеры

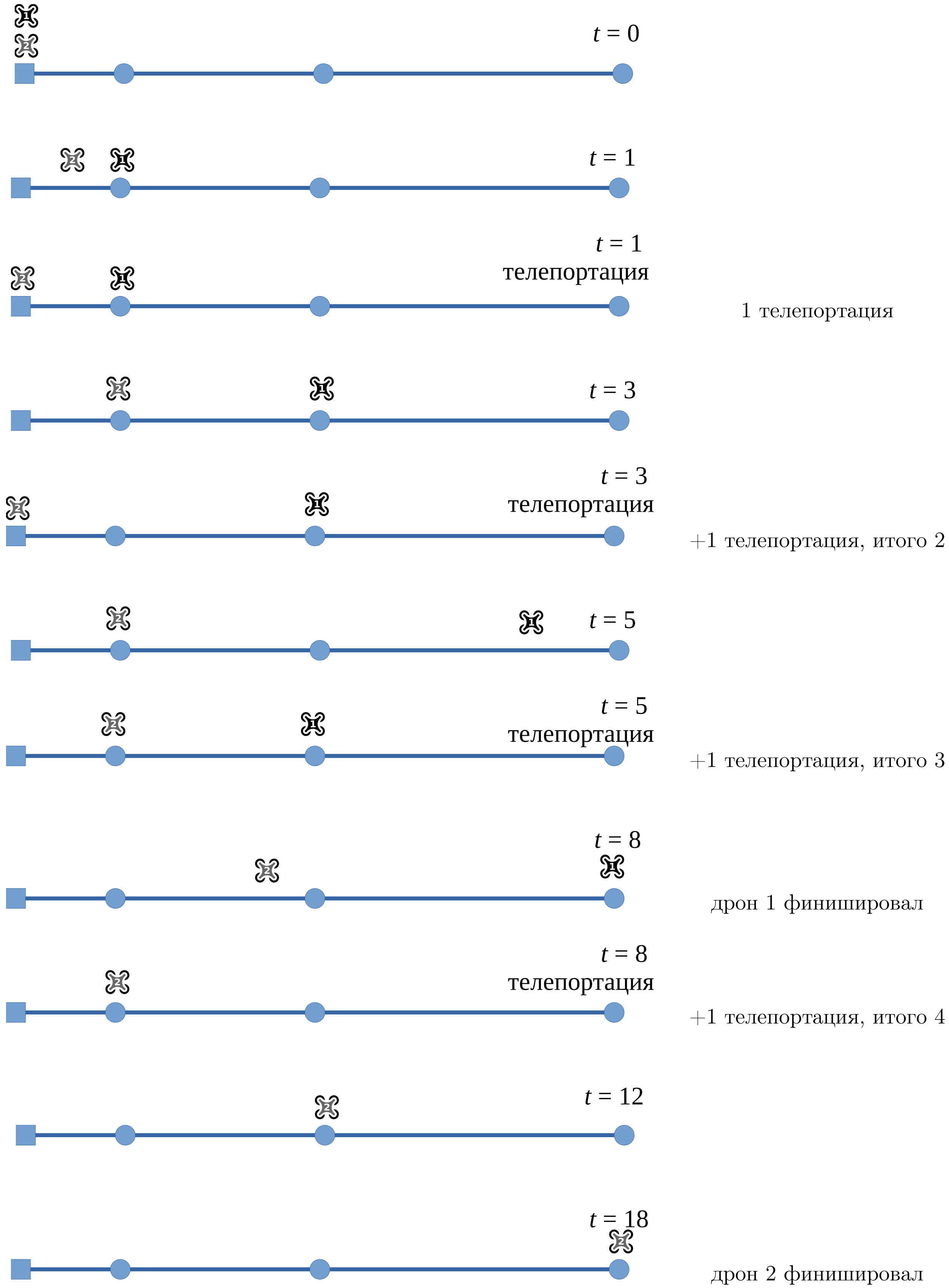
стандартный ввод	стандартный вывод
3 3 1 2 3 1 3 6	0 4 11
3 3 3 2 1 1 3 6	0 5 13
2 5 2 1 1 3 4 6 7	0 6

Пояснения к примерам

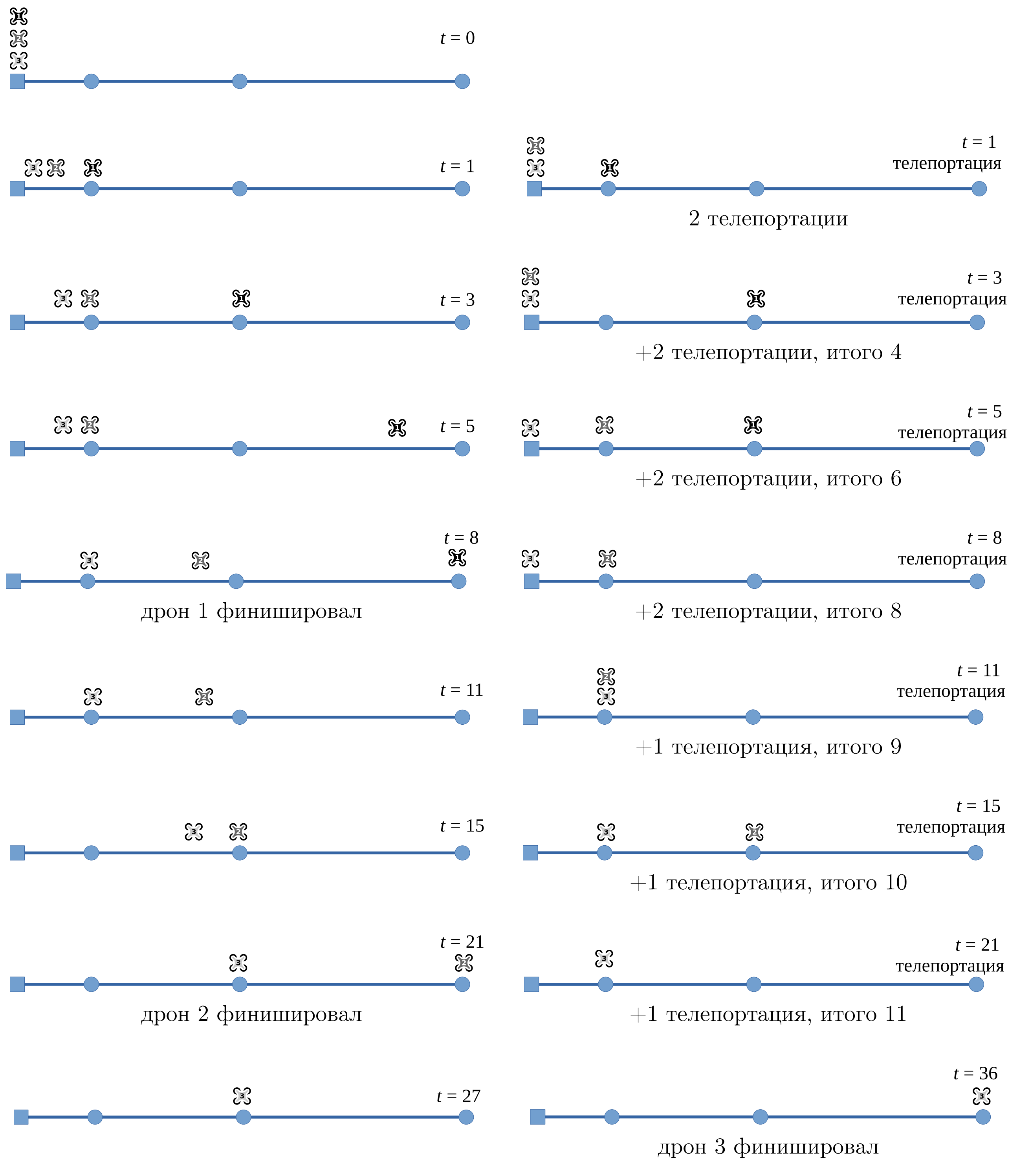
Рассмотрим первый пример.

Если $k = 1$, то телепортаций не происходит.

Если $k = 2$, то гонка происходит следующим образом. На рисунках показаны моменты, когда дроны оказываются в воротах и происходит телепортация.



Если $k = 3$, то гонка происходит следующим образом. На рисунках показаны моменты, когда дроны оказываются в воротах и происходит телепортация.



Задача 8. За связь без перебоев

Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Вдоль прямой дороги, на которой происходят испытания беспилотных грузовиков, расположены n городов, i-й город находится в точке, имеющей координату i. В i-м городе установлена антенна мощностью a_i, покрывающая все города от $L_i = \max(1, i - a_i)$ до $R_i = \min(n, i + a_i)$ включительно.

Беспилотный грузовик перемещается вдоль дороги от города s к городу t, где $s < t$. В каждом городе по пути следования грузовик подключён к одной из антенн. Подключение к антеннам происходит следующим образом.

- В начальном городе грузовик подключается к антенне, покрывающей этот город, у которой значение R_i максимально. Если таких антенн несколько, выбирается любая из них.
- После перемещения грузовика из города v в город $v + 1$, если антенна, к которой он был подключен в городе v, покрывает также и город $v + 1$, грузовик остаётся подключен к этой антенне. Иначе, если антенна, к которой он был подключён, не покрывает город $v+1$, грузовик переподключается к антенне, покрывающей город $v+1$, для которой значение R_i максимально. Если таких антенн несколько, выбирается любая из них.

Обозначим как $f(s, t)$ количество переподключений между антеннами для грузовика, который начинает свой маршрут в городе s и заканчивает свой маршрут в городе t $(s < t)$. Начальное подключение к антенне в городе s переподключением не считается.

Нестойкостью покрытия дороги антеннами назовем сумму значений $f(s, t)$ по всем допустимым парам городов, то есть величину

$$F = \sum_{s=1}^{n-1} \sum_{t=s+1}^{n} f(s, t).$$

В распоряжении оператора дороги есть одна запасная антенна с мощностью x. Для снижения нестойкости покрытия можно заменить одну из антенн на запасную. Требуется определить минимальное значение нестойкости покрытия дороги F, если не более одной антенны можно заменить на запасную антенну мощности x.

Формат входных данных

Первая строка содержит два целых числа n и x $(1 \leqslant n \leqslant 10^6,\ 0 \leqslant x \leqslant n)$ — количество городов и мощность запасной антенны.

Вторая строка содержит n целых чисел $a_1, a_2, \ldots, a_n$ $(0 \leqslant a_i \leqslant n)$ — мощности антенн.

Формат выходных данных

Выведите минимальное возможное значение нестойкости покрытия дороги, если не более одной антенны можно заменить на запасную антенну мощности x.

Система оценки

Подзадачи	Баллы	Ограничения			Необх. подзадачи
		n	x	a_i	
1	7	$n \leqslant 100$			У
2	8	$n \leqslant 500$			У, 1
3	6	$n \leqslant 5000$			У, 1, 2
4	12		$x = 0$		
5	5			$a_i = 0$	
6	16			$a_i \leqslant 1$	5
7	14			$a_i \geqslant \frac{n}{20}$	
8	32				У, 1 – 7

Примеры

стандартный ввод	стандартный вывод
3 1 1 0 0	0
5 0 2 1 0 0 1	6

Пояснения к примерам

В первом примере мы можем заменить вторую антенну на запасную. Тогда грузовик, стартующий в любой точке, будет подключаться к ней и переподключаться никакому грузовику не понадобится.

Во втором примере использовать запасную антенну не нужно. Грузовикам, стартующим в одном из первых трёх городов и финиширующим в одном из двух последних городов придётся один раз переподключиться к последней антенне, поэтому нестойкость покрытия дороги равна 6.