

Задача А. Бобровая змея

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

Мистеру Бобру приходится несладко: он должен присматривать за N детёнышами-бобрами, которые одержимы идеей построиться в одну гигантскую «змею».

i -й детёныш начинает движение из уникальной точки (x_i, y_i) на бесконечной декартовой плоскости. Затем все они начинают двигаться. В любой момент времени скорость каждого бобра $\vec{v} = (v_x, v_y)$ должна удовлетворять ограничениям

$$|v_x| \leq 1 \text{ и } |v_y| \leq 1 \text{ (клеток в секунду)}.$$

Они будут довольны, только когда их конечные положения окажутся на некоторой траектории, идущей **только вверх и вправо**, то есть такой, что координаты (x_i, y_i) образуют последовательность, неубывающую по обеим осям. Заметим, что несколько бобров могут оказаться в одной точке.

Мистер Бобер очень устал и хочет поскорее домой. Помогите ему найти **удвоенное минимальное время**, необходимое для того, чтобы привести детёнышей в описанную «змеиную» формацию. Можно доказать, что это значение — целое число.

Формат входных данных

Первая строка содержит целое число T ($1 \leq T \leq 10^4$) — количество тестов.

Каждый тест начинается с целого числа N ($1 \leq N \leq 2 \cdot 10^5$) — количество детёнышей-бобров.

Следующие N строк содержат по два целых числа x_i и y_i ($0 \leq x_i, y_i \leq 10^9$) — начальные координаты i -го бобра.

Все точки (x_i, y_i) внутри одного теста различны.

Сумма N по всем тестам не превышает $2 \cdot 10^5$.

Формат выходных данных

Для каждого теста выведите одно целое число — **удвоенное минимальное время** (в секундах), необходимое для достижения «змеиной» формации.

Система оценки

В этой задаче две подзадачи:

(30 баллов): Сумма по всем N не превосходит 3000.

(70 баллов): Нет дополнительных условий

Пример

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

Задача В. Квантовый бобер

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Мистер Бобер живет на неокрашенном тротуаре, состоящем из N отдельных плиток, пронумерованных $1, 2, \dots, N$ слева направо. Все плитки изначально имеют стандартный заводской цвет 0, но мистер Бобер хочет это изменить! За несколько дней покраски мистер Бобер намерен перекрасить весь тротуар.

А именно, мистер Бобер записал список из Q планов, i -й из которых гласит, что в день d_i ($0 \leq d_i \leq 2^K - 1$) он перекрасит все плитки на отрезке $[x_i, y_i]$ в цвет c_i , перезаписывая их предыдущие цвета. У мистера Бобра запланировано не более одного плана покраски на каждый день, и его интересует сумма конечных цветов всех плиток после всех перекрашиваний.

Однако мистер Бобер не знает, что он — первый Бобер, живущий в квантовом компьютере! К сожалению, это означает, что вселенная работает не так, как он думает. Ночью, пока никто не видит, кубиты подвергаются случайному преобразованию: выбирается целое число z от 0 до $2^K - 1$, и каждый план, изначально запланированный на день i , вместо этого перемещается на день $i \oplus z$, где \oplus обозначает побитовое исключающее ИЛИ (XOR).

Для данного z планы затем выполняются в порядке возрастания (сдвинутых) дней, и каждая операция перекрашивания перезаписывает цвета плиток с x_i по y_i . Пусть $F(z)$ — сумма конечных цветов всех плиток после применения планов со сдвигом z .

Требуется вычислить $\sum_{z=0}^{2^K-1} F(z)$. Поскольку ответ может быть очень большим, выведите его по модулю $10^9 + 7$.

Формат входных данных

Каждый тест содержит несколько тестовых случаев. Первая строка содержит количество тестовых случаев T ($1 \leq T \leq 150$). Далее следуют описания тестовых случаев.

Первая строка каждого тестового случая содержит три целых числа N , K и Q ($1 \leq N \leq 2 \cdot 10^5$, $0 \leq K \leq 60$, $1 \leq Q \leq \min(2^K, 2 \cdot 10^5)$).

Каждая из следующих Q строк содержит четыре целых числа d_i , x_i , y_i и c_i ($0 \leq d_i < 2^K$, $1 \leq x_i \leq y_i \leq N$, $0 \leq c_i \leq 10^9$).

Гарантируется, что сумма N по всем тестовым случаям не превышает $4 \cdot 10^5$, сумма Q по всем тестовым случаям не превышает $4 \cdot 10^5$, и все d_i внутри одного тестового случая различны.

Формат выходных данных

Для каждого тестового случая выведите одно целое число на отдельной строке: сумма $\sum_{z=0}^{2^K-1} F(z)$, взятая по модулю $10^9 + 7$.

Система оценки

В задаче пять подзадач.

- (10 баллов): Сумма N , 2^K и Q по всем тестовым случаям не превышает $5 \cdot 10^3$. Заметим, что речь идет о каждой из отдельных сумм, а не о их совокупности. Также $c_i \leq 5 \cdot 10^3$ для всех i .
- (20 баллов): Для всех тестовых случаев $N = 1$, $Q \leq 2$.
- (20 баллов): Для всех тестовых случаев $N = 1$.
- (20 баллов): Для всех тестовых случаев $K \leq 18$.
- (30 баллов): Дополнительных ограничений нет.

Примеры

стандартный ввод	стандартный вывод
3 6 4 6 6 3 5 4 12 2 2 3 4 1 4 2 5 3 3 4 14 1 1 3 13 1 6 4 6 4 4 4 1 2 0 3 2 6 4 2 2 2 2 5 3 5 0 4 4 6 8 2 4 2 11 1 3 2 9 3 4 4 10 3 3 2 2 3 4 3 3 1 4 5	332 184 220
3 5 4 4 9 1 2 2 12 1 3 1 10 4 5 3 14 1 5 4 6 4 4 5 2 4 3 14 5 6 2 10 3 6 3 1 4 6 5 6 4 5 11 2 4 5 13 1 2 4 14 4 6 5 10 1 6 3 15 2 6 0	224 272 276

Задача С. Рисование линий

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

Мистер Бобер любит рисовать линии на бумаге. Однако ему становится грустно, когда две линии пересекаются. Однажды ночью ему приснился шедевр — рисунок, состоящий из множества лучей, не имеющих пересечений. Но, проснувшись, он запомнил только, где начинаются лучи, а в каком направлении они идут — забыл! Помогите ему воссоздать рисунок.

Формально, вам даны N точек на плоскости, каждая с различными целочисленными координатами x и y в диапазоне $[1, N]$. Каждая точка помечена как UD или LR. Для каждой точки UD вы должны нарисовать бесконечный луч, начинающийся в этой точке и идущий вертикально — либо вверх, либо вниз. Аналогично, для каждой точки LR вы должны нарисовать бесконечный луч, начинающийся в этой точке и идущий горизонтально — либо влево, либо вправо.

Из всех 2^N способов назначить направления для каждой точки определите, существует ли назначение, в котором никакие два луча не пересекаются, и если существует, выведите количество подходящих назначений по модулю $10^9 + 7$.

Для вашего удобства гарантируется, что если x — количество подходящих назначений и $x > 0$, то $x \not\equiv 0 \pmod{10^9 + 7}$.

Формат входных данных

Каждый тест содержит несколько тестовых случаев. Первая строка ввода содержит одно целое число T ($1 \leq T \leq 10^4$) — количество тестовых случаев.

Первая строка каждого тестового случая содержит целое положительное число N ($1 \leq N \leq 10^3$), обозначающее количество точек.

Каждая из следующих N строк содержит два целых числа x_i, y_i и строку s_i , где (x_i, y_i) обозначает местоположение i -й точки, а $s_i \in \{UD, LR\}$ обозначает допустимые направления луча из точки.

Гарантируется, что все x_i являются попарно различными целыми числами в диапазоне $[1, N]$, и то же самое верно для y_i .

Гарантируется, что сумма N по всем тестовым случаям не превышает 10^4 .

Формат выходных данных

Для каждого из T тестовых случаев выведите две строки.

На первой строке выведите "YES" или "NO", обозначая, существует ли подходящее назначение. Вы можете выводить ответ в любом регистре (верхнем или нижнем). Например, строки "YES", "yes", "Yes" и "YES" будут распознаны как положительные ответы.

На второй строке выведите количество подходящих назначений по модулю $10^9 + 7$.

Система оценки

Вы получите 50% баллов за каждую подзадачу, если ответы YES/NO верны, а количество подходящих назначений — нет.

- (10 баллов): $N \leq 10$.
- (40 баллов): $N \leq 100$.
- (50 баллов): Дополнительные ограничений нет.

Пример

стандартный ввод	стандартный вывод
3	YES
2	4
1 1 UD	YES
2 2 UD	3
2	NO
1 2 UD	0
2 1 LR	
7	
1 5 UD	
2 3 UD	
3 4 LR	
4 1 LR	
5 7 LR	
6 2 UD	
7 6 UD	

Задача D. Пианино Бобрик

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	6 секунд
Ограничение по памяти:	256 мегабайт

Мистер Бобер — страстный поклонник пианино и мечтает играть длинные, сложные произведения на своём пианино. Однако, имея всего лишь жалкие 10 пальцев, он не может сделать многого. К счастью, его друг Бобрик может отрастить любое количество пальцев на своей руке, что позволяет ему играть любые произведения.

Бобрик пытается сыграть мелодию, состоящую из N нот, пронумерованных от 1 до N . Чтобы сыграть i -ю ноту, Бобрик должен нажать клавишу k_i в момент времени a_i и отпустить её в момент b_i . Гарантируется, что никакие две ноты с одинаковой клавишей не играют одновременно. Другими словами, для всех пар нот (i, j) , где $k_i = k_j$, выполняется либо $b_i \leq a_j$, либо $b_j \leq a_i$.

Бобрик планирует использовать одну руку с F пальцами, пронумерованными слева направо от 1 до F , чтобы сыграть мелодию. Он хочет назначить палец f_i для i -й ноты. Бобрик может сыграть мелодию, если он может назначить палец для каждой ноты так, чтобы выполнялись следующие критерии:

— Ни в какой момент времени его пальцы не пересекаются. Формально: для всех пар нот (i, j) , для которых существует момент времени t такой, что $a_i < t < b_i$ и $a_j < t < b_j$, если $k_i < k_j$, то $f_i < f_j$.

Бобрику разрешено удалить из мелодии не более M нот, где M равно 0 или 1. Найдите минимальное количество пальцев, которое ему нужно на руке, чтобы сыграть оставшуюся мелодию. Другими словами, найдите наименьшее F , для которого существует назначение пальцев f , удовлетворяющее указанным выше критериям.

Формат входных данных

Каждый тест содержит несколько тестовых случаев. Первая строка содержит количество тестовых случаев T ($1 \leq T \leq 10^4$). Далее следуют описания тестовых случаев.

Первая строка каждого тестового случая содержит два целых числа N и M ($1 \leq N \leq 2 \cdot 10^5$, $M = 0$ или 1).

Каждая из следующих N строк содержит три целых числа k_i , a_i и b_i ($1 \leq k_i \leq 10^9$, $1 \leq a_i < b_i \leq 10^9$) — это означает, что i -я нота использует клавишу k_i с момента времени a_i до момента b_i .

Гарантируется, что никакие две ноты с одинаковой клавишей не играют одновременно, и сумма N по всем тестовым случаям не превышает $2 \cdot 10^5$.

Формат выходных данных

Для каждого тестового случая выведите минимальное количество пальцев, которое нужно Бобрику на руке, чтобы после удаления не более M нот существовало подходящее назначение пальцев.

Система оценки

В задаче три подзадачи.

- (10 баллов): Сумма N по всем тестовым случаям не превышает 10^3 и $M = 0$.
- (20 баллов): $M = 0$.
- (70 баллов): Дополнительных ограничений нет.

Пример

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