Задача A. Круглая карта

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

Новая карта в Одиночном столкновении представляет собой круг с центром в координате (0, 0) и радиусом 1. По периметру стоит стена. Возле края карты, у стены в точке (x_0, y_0) стоит Рико и делает выстрел пулями, которые, как известно, могут отскакивать от стен. При этом угол падения пули равен углу отражения (для отражения от кривых поверхностей проводится касательная в точке и сравниваются углы падения и отражения относительно неё).

Известно, что первый отскок пули от стены был в точке (x_1, y_1). Определите, в какой точке будет находиться пуля в момент n-го удара о стену.

Формат входных данных

В первой строке вводятся 2 вещественных числа (x_0, y_0) координаты Рико $(x_0^2 + y_0^2 = 1)$.

Во второй строке тоже 2 вещественных числа (x_1, y_1) – координаты точки первого удара о стену $(x_1^2 + y_1^2 = 1)$.

В третьей строке вводится целое число n $(1 \leqslant n \leqslant 10^9)$.

Формат выходных данных

Выведите пару чисел (x, y) – координаты точки в момент n-го удара пули о стену.

Абсолютная или относительная погрешность ответа не должна превышать 10^{-6} по обеим координатам. Формально, если координата в вашем ответе равна a, а в ответе жюри – b, то должно выполняться $\frac{|a-b|}{\max(b,1)} \leqslant 10^{-6}$.

Примеры

стандартный ввод	стандартный вывод
1.0 0.0 0.0 1.0 5	0.00000000 1.00000000
1.0 0.0 -1.0 0.0 9	-1.00000000 0.00000000
1.0 0.0 0.70710678 0.70710678 7	0.70710678 -0.70710678

Задача B. Очень большая игра

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

В событии «Очень большая игра» команда из n бойцов пытается победить одного Мегабойца. В команде есть несколько типов бойцов, и перед боем они выстроились в линию, готовясь атаковать по очереди.

Однако прямо перед началом выяснилось, что Мегабоец восстанавливает всё здоровье каждый раз, когда тип атакующего бойца меняется. То есть суммарный урон, который команда сможет нанести до следующего восстановления, равен длине самой длинной последовательности подряд стоящих бойцов одного типа.

Чтобы увеличить урон, бойцы решили слегка изменить порядок построения. У них есть время ровно на одну операцию: можно взять подряд несколько бойцов (отрезок с позиций l до r) и вставить их в другое место строя. Например, если изначально порядок был «$ABCD$», то после операции можно получить «$ACBD$», где A, B, C, D — это группы подряд идущих бойцов. Какой наибольший урон команда может нанести Мегабойцу, если успеют сделать такую операцию?

Формат входных данных

В первой строке вводится целое число n – количество бойцов ($1 \leqslant n \leqslant 10^6$).

Во второй строке вводятся целые числа $a_1, a_2, ...a_n$ – типы бойцов, в том порядке, в котором они стоят в строю ($1 \leqslant a_i \leqslant 10^9$).

Формат выходных данных

Выведите одно челое число – наибольший урон, который команда может нанести Мегабойцу.

Примеры

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

Замечание

В первом примере можно взять отрезок с 3-го до 4-го бойца и переставить их в начало, то есть из 1 1[2 1]2 получится [2 1]1 1 2, то есть 2 1 1 1 2. Значит итоговый урон будет равен 3.

Во втором примере переставим отрезок с 1-го до 2-го бойца на 2 элемента вправо, получится последовательность 1 3 1 2 2 2 1.

Задача C. Расстановка бойцов

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

На арене вам нужно расставить n бойцов так, чтобы каждый контролировал определенный участок карты. Карту в этой задаче можно представить как координатную прямую. Боец номер i контролирует отрезок $[l_i, r_i]$ на координатной прямой, где l_i и r_i — целые числа в диапазоне от -10^9 до 10^9.

Вам также дано число k — максимальная плотность скопления бойцов. После расстановки всех n бойцов для каждой целочисленной точки x на карте подсчитывается, сколько бойцов контролируют эту точку. Вы хотите достичь идеального баланса:

- Должна быть ровно одна точка, которую контролирует ровно один боец
- Должна быть ровно одна точка, которую контролируют ровно два бойца
- ...
- Должна быть ровно одна точка, которую контролируют ровно k бойцов
- Не должно быть ни одной точки, которую контролируют более k бойцов

Сможете ли вы найти такую расстановку бойцов на карте?

Формат входных данных

В единственной строке содержатся два целых числа n и k — количество бойцов и максимальная плотность скопления ($1 \leqslant n, k \leqslant 10^5$).

Формат выходных данных

В первой строке выведите «YES», если возможна требуемая расстановка бойцов, и «NO» в противном случае.

Если решение существует, в следующих n строках выведите по два целых числа l_i, r_i — координаты участков, которые контролируют бойцы ($-10^9 \leqslant l_i \leqslant r_i \leqslant 10^9$).

Примеры

стандартный ввод	стандартный вывод
1 1	YES 1 1
2 1	NO
1 2	NO
4 3	YES 10 10 3 4 3 4 3 3

Задача D. Сбор команды

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

В мире Brawl Stars существует иерархическая система бойцов. Эта система представляет собой корневое дерево, где вершинами являются бойцы, рёбрами — отношения подчинения, а в корне дерева находится боец номер 1.

Каждый боец i обладает типом способности b_i ($1 \leqslant b_i \leqslant n$).

Вам дан целевой набор способностей $a_1, a_2, \ldots, a_m$, который необходим для выполнения специальной миссии.

Командой называется любой отряд бойцов, который включает некоторого бойца и всех его подчинённых (то есть поддерево в иерархической структуре).

Требуется найти количество таких команд, в составе которых имеются все перечисленные в целевом наборе типы способностей. Команда считается подходящей, если мультимножество типов способностей её бойцов содержит все типы из целевого набора (с учётом кратностей).

Формат входных данных

Первая строка содержит целое число n — количество бойцов в системе ($1 \leqslant n \leqslant 10^5$).

Вторая строка содержит n целых чисел $b_1, \ldots, b_n$ — типы способностей бойцов ($1 \leqslant b_i \leqslant n$).

Следующие $n-1$ строк содержат пары целых чисел u_i, v_i — связи подчинения между бойцами ($1 \leqslant u_i, v_i \leqslant n$).

Гарантируется, что структура связей образует дерево с корнем в позиции 1.

Следующая строка содержит целое число m — размер целевого набора способностей ($1 \leqslant m \leqslant 10^5$).

Последняя строка содержит m целых чисел $a_1, \ldots, a_m$ — требуемые типы способностей ($1 \leqslant a_i \leqslant n$).

Формат выходных данных

Выведите одно целое число — количество подходящих команд.

Примеры

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

Задача E. Оглуши меня полностью

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

В игре под названием «Brawl Stars» есть два персонажа, которые могут оглушать противников своими способностями — Фрэнк и Булл. Недавно в игру добавили возможность перезарядки способностей. Также добавили артефакты, которые позволяют сокращать перезарядки. Перезарядка начинается сразу после начала применения способности.

После данного обновления два друга, Витя и Антон, придумали стратегию - бесконечное оглушение. Суть в том, чтобы сократить на максимум свои перезарядки и держать в оглушении своего противника, пока у него не закончатся очки здоровья. Это невероятно весёлая стратегия!

Но суть в том, что артефакты не бесплатные, а Витя и Антон школьники, у которых не так много денег. Помогите им узнать, можно ли реализовать данную стратегию и какое минимальное количество денег на артефакты надо потратить для реализации.

Формат входных данных

В первой строке даны целые числа a, b, c, d — время оглушения Фрэнка и Булла, затем перезарядки их способностей. ($1 \leqslant a, b, c, d \leqslant 10^4$)

Во второй строке даны два числа n, m — количество артефактов для Фрэнка и Булла соответственно. ($0 \leqslant n, m \leqslant 100$)

В следующих n строках идут две пары чисел — время сокращения перезарядки p и стоимость артефакта q. ($0 \leqslant p, q \leqslant 10^4$)

Аналогично для следующих m строк.

Формат выходных данных

В первой строке выведите «YES», если такое возможно, иначе «NO».

Во второй строке выведите минимальную суммарную стоимость артефактов, если это возможно.

Примеры

стандартный ввод	стандартный вывод
2 2 7 6 3 1 3 10 2 1 1 1 2 4	YES 6
1 1 3 2 1 0 1 10	YES 10

Задача F. Прагающая Пайпер

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

Пайпер использует свою ульту, чтобы прыгать по клеткам на числовой прямой. Клетки пронумерованы целыми числами от $-n$ до n, центр — в клетке 0. На первом прыжке она может переместиться на 1 клетку, на втором — на 2, на третьем — на 4, затем на 8 и так далее: длина k-го прыжка равна 2^{k-1}.

Перед каждым прыжком Пайпер может выбрать направление — влево или вправо. Прыгать обязательно (нельзя пропускать прыжки) и очевидно, что Пайпер не может выпрыгнуть за границы карты, которая размером $1 \times (2 \cdot n + 1)$. Изначально Пайпер стоит в клетке 0, которая является центром поля: $[-n, n]$.

Ваша задача: для каждой заданной клетки a ($-n \leqslant a \leqslant n$) определить, за какое минимальное количество прыжков Пайпер может попасть в неё, начиная из клетки 0. Если попасть в клетку a невозможно, выведите -1.

Формат входных данных

Первая строка содержит два целых числа n и q ($1 \leqslant n \leqslant 10^{18}$, $1 \leqslant q \leqslant 10^6$).

Вторая строка содержит q целых чисел $a_1, a_2 \ldots, a_q$ — клетки, в которые Пайпер хочет попасть, ($-n \leqslant a_i \leqslant n$).

Формат выходных данных

Для каждого запроса выведите одно число — минимальное количество прыжков, необходимых Пайпер, чтобы попасть в клетку a_i, или -1, если попасть туда невозможно.

Примеры

стандартный ввод	стандартный вывод
3 4 -1 -2 3 3	1 -1 2 2
77 9 -40 -43 9 -18 21 -73 -61 63 -29	-1 6 4 -1 5 7 6 6 5

Задача G. Лава в Джейлбрейке

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

После большого взрыва в Джейлбрейке на арене появились опасные вулканы! Тебе, как бойцу Brawl Stars, нужно уметь предсказывать распространение лавы, чтобы избежать поражения.

На прямой расположены n вулканов. i-й вулкан начинает извергаться в момент времени t_i из точки p_i с максимальной высотой лавы h_i.

Лава распространяется по следующим правилам:

- В момент извержения (t_i) в точке p_i высота лавы равна 0.
- За каждую единицу времени лава распространяется на 1 единицу расстояния в обе стороны.
- Высота лавы в точке $p_i + y$ в момент времени $t_i + x$ после извержения вычисляется как $\min(\max(0, x - |y|), h_i)$.
- Если лава из нескольких вулканов достигает одной точки, берется максимальная высота.

Тебе нужно ответить на q запросов: для момента времени T и точки X определи высоту лавы.

Формат входных данных

Первая строка содержит два целых числа n и q — количество вулканов и запросов ($1 \leqslant n, q \leqslant 2 \cdot 10^5$).

Следующие n строк содержат по три целых числа t_i, p_i, h_i — параметры вулканов ($0 \leqslant t_i \leqslant 10^9$, $-10^9 \leqslant p_i \leqslant 10^9$, $1 \leqslant h_i \leqslant 10^9$).

Следующие q строк содержат по два целых числа T_j, X_j — параметры запросов ($0 \leqslant T_j \leqslant 10^9$, $-10^9 \leqslant X_j \leqslant 10^9$).

Формат выходных данных

Для каждого запроса выведите одно целое число — высоту лавы в точке X_j в момент времени T_j.

Примеры

стандартный ввод	стандартный вывод
2 3 1 0 3 3 5 2 2 1 4 2 5 6	0 1 1
1 2 0 0 5 1 1 2 3	0 0

Задача H. Трофейная дорога Эдгара

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

Эдгар решил собрать максимальную трофейную дорогу в Brawl Stars! У него есть n специальных карточек с наградами. Каждая карточка имеет два значения: количество трофеев слева l_i и количество трофеев справа r_i (каждое значение от 1 до 6).

Эдгар может выкладывать карточки в цепочку по особым правилам: следующую карточку можно положить рядом с предыдущей только если соседние значения совпадают. Например, если у первой карточки правое значение равно 3, то левое значение следующей карточки тоже должно быть равно 3. Также карточки можно переворачивать, то есть менять местами значения l_i и r_i.

Формально, цепочка карточек с индексами $i_1, i_2, \ldots, i_k$ является корректной, если:

- Все карточки различны (каждая карточка используется не более одного раза)
- Для каждой пары соседних карточек выполнено: $r_{i_j} = l_{i_{j+1}}$ для всех $1 \leqslant j < k$

Помогите Эдгару найти максимальную длину цепочки, которую он может собрать!

Формат входных данных

В первой строке вводится целое число n — количество карточек ($1 \leqslant n \leqslant 10^5$).

В следующих n строках вводится по два целых числа l_i и r_i — значения на левой и правой части i-й карточки ($1 \leqslant l_i, r_i \leqslant 6$).

Формат выходных данных

Выведите одно целое число — максимальную длину цепочки карточек, которую можно собрать.

Примеры

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

Замечание

В первом примере можно выложить цепочку: карточка 1 (1-2), карточка 2 (2-3), карточка 3 (3-4). Длина цепочки равна 3.

Во втором примере можно выложить цепочку: карточка 2 (2-3), карточка 3 (3-2), карточка 1 (2-1). Длина цепочки равна 3.

В третьем примере все карточки одинаковые (1-1), поэтому можно выложить их все подряд. Длина цепочки равна 4.

Задача I. Безопасный прыжок

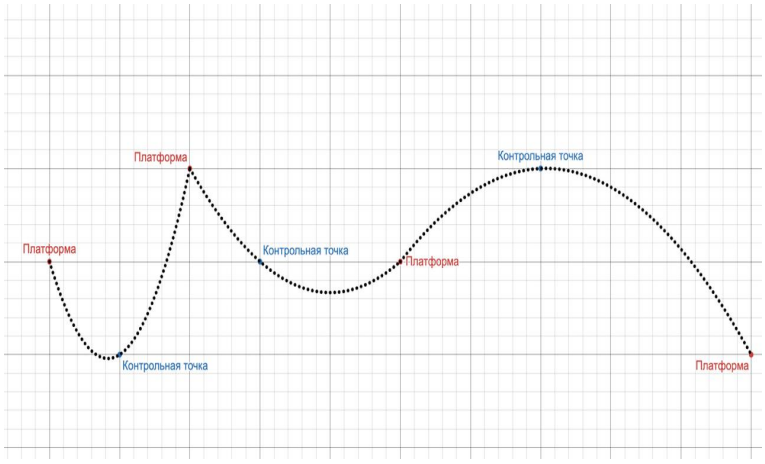
Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

В мире **Brawl Stars** стартует новое испытание — *«Безопасный прыжок»*! Бойцы должны перепрыгивать пропасти, но их траектории описываются параболами. Если хотя бы один из прыжков опустится ниже земли, боец выбывает

Есть n платформ, расположенных по возрастанию координаты x. Между каждой парой соседних платформ есть **контрольная точка**, через которую также проходит траектория прыжка.

Траектория каждого прыжка между платформами i и $i+1$ описывается параболой, проходящей через три точки:

- платформу i;
- платформу $i+1$;
- контрольную точку между ними.



Твоя задача — найти минимальную высоту, до которой опустится хоть одна из траекторий.

Если минимальная высота $\geqslant 0$, выведи это значение (с точностью до трёх знаков после запятой). Если хотя бы одна траектория опускается ниже нуля, то выведи 0, так как ниже пола опуститься нельзя

Формат входных данных

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

В следующих n строках заданы координаты и высоты платформ: x_i y_i, где $x_1 < x_2 < \cdots < x_n$ ($0 \leqslant x_i, y_i \leqslant 10^6$).

Далее следуют $n-1$ строк с координатами контрольных точек: x'_j y'_j, где $x_i < x'_i < x_{i+1}$ ($0 \leqslant x'_i, y'_i \leqslant 10^6$).

Формат выходных данных

Выведи одно число — минимальную высоту прыжка

Пример

стандартный ввод	стандартный вывод
6 0 5 8 6 16 8 28 9 42 11 58 10 4 7 12 4 22 10 35 13 50 9	3.9166666667

Задача J. Зеркальная Галерея

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

В секретной лаборатории Старр Парка Брок строит зеркальную галерею для тренировок бойцов. Галерея представляет собой длинный коридор, в начале которого установлен прожектор, а в конце — идеальное зеркало.

Когда Брок включает прожектор, луч света летит к зеркалу, отражается от него, возвращается к прожектору, снова отражается и так продолжает бесконечно двигаться туда-обратно. Каждый раз, когда луч достигает прожектора или зеркала, происходит отражение.

Расстояние между прожектором и зеркалом составляет ровно 1 метр. Брок хочет знать: сколько отражений произойдет до того момента, когда луч света пройдет суммарное расстояние не менее n метров?

Формат входных данных

В единственной строке вводится целое число n — минимальное суммарное расстояние, которое должен пройти луч ($1 \leqslant n \leqslant 10^{18}$).

Формат выходных данных

Выведите одно целое число — количество отражений, которые произойдут до того, как луч пройдет суммарное расстояние не менее n метров.

Примеры

стандартный ввод	стандартный вывод
1	1
2	2
3	3

Замечание

В примерах:

- При $n = 1$: луч проходит 1 метр (до зеркала) — 1 отражение.

- При $n = 2$: луч проходит 1 метр до зеркала (1 отражение), затем 1 метр обратно до прожектора (2 отражения) — итого 2 отражения при пройденном расстоянии 2 метра.

- При $n = 3$: луч проходит 2 метра (2 отражения), затем еще 1 метр до зеркала (3 отражения) — итого 3 отражения при пройденном расстоянии 3 метра.