

Задача А. Разминочка

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

Вам даны n точек на плоскости. Точка заданы своими координатами: $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$.

Найдите точку (x, y) с целыми координатами, которая минимизирует суммарное манхэттенское расстояние до данных точек, т. е. $\sum_{i=1}^n (|x - x_i| + |y - y_i|)$.

Формат входных данных

В первой строке вводится число n ($1 \leq n \leq 10^5$).

В следующий n строках через пробел вводятся пары чисел x_i, y_i — координаты данных точек.

Формат выходных данных

Выведите через пробел пару чисел x, y — координаты искомой точки. Если оптимальных ответов несколько выведите любой из них.

Система оценки

В этой задаче 6 подзадач:

Подзадача	Баллы	N	Доп. ограничения
0	0	—	тесты из условия
1	3	$N = 2$	$0 \leq X_i, Y_i \leq 10^9$
2	20	$2 \leq N \leq 1000$	$0 \leq X_i \leq 1000, Y_i = 0$
3	28	$2 \leq N \leq 100\,000$	$0 \leq X_i \leq 10^9, Y_i = 0$
4	13	$2 \leq N \leq 100$	$0 \leq X_i, Y_i \leq 100$
5	17	$2 \leq N \leq 1000$	$0 \leq X_i, Y_i \leq 10^9$
6	19	$2 \leq N \leq 100\,000$	$0 \leq X_i, Y_i \leq 10^9$

Примеры

стандартный ввод	стандартный вывод
2 1 0 4 0	4 0
6 1 0 3 0 5 0 7 0 9 0 11 0	7 0
9 1 16 3 12 5 6 7 10 9 8 11 4 13 14 15 2 17 18	9 10

Задача В. Нападение!

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

О ужас! Инопланетяне напали на Яндекс!

Они выстроили всех сотрудников в ряд. Также они нашли базу данных Яндекса, в которой для каждого сотрудника записана его полезность v_i (v_i — это полезность i -го в ряду сотрудника).

Инопланетяне могут применить луч НЛО, который заберет некоторый подотрезок работников. Однако инопланетяне не хотят менять баланс сил на Земле, поэтому они могут применить луч НЛО только на отрезок сотрудников, суммарная полезность которых равна нулю.

У инопланетян есть q планов. Согласно i -му плану луч НЛО может забирать только работников от l_i -го до r_i -го включительно. Для каждого плана нужно посчитать, какое максимальное количество лучей НЛО инопланетяне смогут применить, что никакие два отрезка лучей НЛО не пересекаются.

Вы выбрали правильную сторону и решили помочь инопланетянам.

Формат входных данных

В первой строке входных данных записано целое число n ($1 \leq n \leq 4 \cdot 10^5$) — количество сотрудников.

В следующей строке записаны n целых чисел v_1, v_2, \dots, v_n ($-10^9 \leq v_i \leq 10^9$) — полезности сотрудников.

В следующей строке записано целое число q ($1 \leq q \leq 4 \cdot 10^5$) — количество планов.

В каждой из следующих q строк записаны два целых числа l_i и r_i ($1 \leq l_i \leq r_i \leq n$) — границы i -го плана.

Формат выходных данных

Для каждого плана в отдельной строке выведите целое число — максимальное количество лучей НЛО, которые инопланетяне могут применить.

Система оценки

В этой задаче 3 подгруппы:

Подзадача	Баллы	n	q	Доп. ограничения
0	0	—	—	тесты из условия
1	22	$1 \leq n \leq 5\,000$	$1 \leq q \leq 5\,000$	—
2	39	$1 \leq n \leq 100\,000$	$1 \leq q \leq 100\,000$	—
3	39	—	—	нет дополнительных ограничений

Пример

стандартный ввод	стандартный вывод
10	4
1 2 -3 0 1 -4 3 2 -1 1	2
3	2
1 10	
1 5	
2 9	

Задача С. NP полная задача?

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

Дан неориентированный граф G на n вершинах (вершины занумерованы от 1 до n) и целое число K ($1 \leq K \leq 10$). Граф задаётся списком рёбер. Известно, что в любом непустом подмножестве вершин найдётся вершина, степень которой в подграфе, порождённом этим подмножеством, строго меньше K . Ясно, что в таком графе не может быть клики размера $K + 1$ или больше.

Найдите размер максимальной клики в графе G . **Клик** называется множество попарно смежных вершин.

Формат входных данных

В первой строке записаны три целых числа n , K и m ($1 \leq n \leq 50\,000$, $1 \leq K \leq 10$, $0 \leq m \leq 10^6$) — число вершин графа, параметр K и число рёбер.

В каждой из следующих m строк записаны два целых числа u и v ($1 \leq u, v \leq n$) — номера концов ребра. Гарантируется, что в графе нет петель и кратных ребер.

Формат выходных данных

Выведите одно целое число — размер максимальной клики в графе G .

Система оценки

Подзадача	Баллы	n	Доп. ограничения
0	0	—	тесты из условия
1	4	$n \leq 5\,000$	$K \leq 2$
2	12	$n \leq 5\,000$	$K \leq 3$
3	23	—	степень каждой вершины не превосходит 10
4	38	$n \leq 5\,000$	—
5	23	—	$K \leq 5$

Примеры

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

Задача D. Пути в дереве

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

Дано дерево на n вершинах. Вершины пронумерованы от 1 до n . Требуется добавить минимальное число рёбер так, чтобы граф оставался связным при удалении любого одного ребра (исходного или добавленного). Иначе говоря, после добавления рёбер в графе не должно быть мостов.

Формат входных данных

В первой строке записано целое число n ($3 \leq n$) — число вершин. В каждой из следующих $n - 1$ строк записаны два целых числа a и b ($1 \leq a, b \leq n, a \neq b$) — концы ребра дерева.

Формат выходных данных

В первой строке выведите целое число k — минимальное число добавляемых рёбер. В каждой из следующих k строк выведите два целых числа a и b ($1 \leq a, b \leq n, a \neq b$) — концы добавляемого ребра. Рёбра можно выводить в любом порядке. Если решений несколько, выведите любое.

Система оценки

Подзадача	Баллы	Ограничения
1	18	$n \leq 10$
2	45	$n \leq 2\,000$
3	37	$n \leq 500\,000$

Примеры

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

Задача Е. Центральный парк

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

Город Нью-Йорк можно представить в виде таблицы из квадратиков $N \cdot N$, где каждый квадратик - отдельный квартал.

Клетка в i -й строке и j -м столбце имеет координаты (i, j) .

Где-то в городе расположен Центральный парк. Парк представляет из себя прямоугольную или квадратную область со сторонами, параллельными осям координат, и занимает какие-то квадраты города. Где расположен парк неизвестно, но вы точно знаете что парк достаточно большой, а именно что он занимает не менее 4% площади всего города (т. е. если стороны парка имеют длины H и W , то выполнено $H \cdot W \geq 0.04 \cdot N \cdot N$).

Вы хотите узнать является парк квадратом или нет. Вы можете прийти в определенный квартал города и узнать, расположен ли там парк.

Вы можете задать не более Q запросов.

Ниже приведены два примера: слева парк является квадратом, справа — прямоугольником. Клетки парка закрашены.

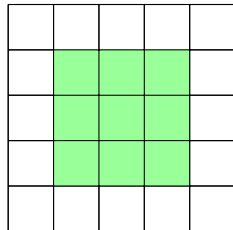


Рис. 1: квадрат

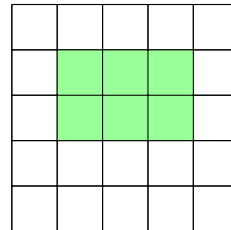


Рис. 2: прямоугольник

Протокол взаимодействия

В первой строке считайте три целых числа T, Q, N ($1 \leq T \leq 200, 1 \leq Q \leq 10^4, N = 100$) — количество независимых тестовых случаев, ограничение на количество запросов и размер города соответственно. **Обратите внимание, что во всех тестах $N = 100$.**

Затем начинаются T тестовых случаев.

В каждом тестовом случае вы можете задать не более Q запросов.

Для того, чтобы задать запрос, выведите строку «? x y» ($1 \leq x, y \leq N$) — координаты квадрата, который вы хотите проверить.

После этого считайте ответ. Ответ будет равен «park» или «town» в зависимости от того, расположен ли парк в этом квадрате или нет.

Когда вы будете готовы вывести ответ, выведите строку «! square» если парк является квадратом, и «! rectangle» иначе.

Система оценки

В этой задаче 4 подзадачи.

Подзадача	Баллы	Доп. ограничения
0	0	тесты из условия
1	14	$Q = 10^4$
2	19	$Q = 100$
3	18	$Q = 40$, фигура занимает не менее 25% площади сетки
4	49	$Q = 50$. Подробности см. в разделе «Scoring»

Во всех подзадачах, кроме 4, вы получите полный балл, если решите все тестовые случаи за не более Q запросов.

Подзадача 4 — специальная подзадача. Ваш балл зависит от максимального количества запросов q , которые вы сделали в любом тестовом случае.

- Если $q > 50$, то вы получите 0 баллов.
- Если $34 \leq q \leq 50$, то вы получите $40 - 30 \cdot \frac{q-34}{17}$ баллов.
- Если $q \leq 33$, то вы получите 49 баллов.

Пример

стандартный ввод	стандартный вывод
2 10000 100	
	? 20 20
park	
	? 19 20
town	
	? 20 19
town	
	? 40 40
park	
	? 41 40
town	
	? 40 41
town	
	! square
park	
	? 20 20
town	
	? 19 20
town	
	? 20 19
park	
	? 40 39
town	
	? 40 40
town	
	? 41 39
	! rectangle

Замечание

В примере пустые строки между выводами оставлены для наглядности, вашему решению их выводить не нужно.

Задача F. Поступление в ВУЗ

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

После окончания школы лучшие программисты выбирают, в какой из двух вузов поступить: MSE или НИПТ. В каждом вузе есть n мест. Всего ребят $2n$, они пронумерованы от 1 до $2n$.

В каждом из вузов есть большая аудитория, в которой будут учиться первокурсники. В обоих вузах в этой аудитории есть n парт, за каждой партой может сидеть только один ученик.

Так как олимпиадники привередливы, то каждый из них решил, что если он будет учиться в MSE, то будет сидеть за партой номер s_i , а если в НИПТ, то за партой h_i .

Также у каждого из программистов есть показатель силы. Так как программисты разумны (а также они хотят честного соревнования на ICPC), они решили распределиться по вузам так, чтобы разница суммарной силы между двумя вузами не превосходила k .

Вам нужно определить, существует ли корректное распределение по вузам, при котором за каждой партой сидит ровно один ученик и разница суммарной силы между вузами не превосходит k .

Формат входных данных

В первой строке записаны целое число n ($1 \leq n \leq 30\,000$) и целое число k ($0 \leq k \leq 20n$) — число мест в каждом вузе и максимально допустимая разница суммарных сил.

В каждой из следующих $2n$ строк записаны три целых числа m_i , h_i и s_i ($1 \leq m_i, h_i \leq n$, $1 \leq s_i \leq 20$) — номер желаемого места в MSE, номер желаемого места в НИПТ и сила программиста i .

Формат выходных данных

Выведите одну строку: YES, если искомое распределение возможно, и NO иначе.

Система оценки

Подзадача	Баллы	Ограничения
1	18	$n \leq 10$
2	30	$n \leq 2\,000$
3	23	$n \leq 30\,000$, все $s_i = 1$
4	29	$n \leq 30\,000$

Примеры

стандартный ввод	стандартный вывод
4 1 1 1 1 2 1 2 2 2 8 1 2 2 3 3 5 3 3 2 4 4 1 4 4 2	YES
2 5 1 1 1 1 2 4 2 2 1 2 1 4	NO

Задача G. Римская республика

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

После изгнания последнего царя из Рима, жители этого города решили разделить власть между двумя консулами.

Всего в городе живет n граждан, они пронумерованы от 1 до n . Во время выборов i -й житель проголосует за a_i -го жителя.

Римляне хотят, чтобы консулы представляли разные слои населения, поэтому они хотят максимизировать разность между номерами консулов.

Выборы будут проходить следующим образом:

- Жители разделяются на две непустые группы.
- Первым консулом станет житель, за которого проголосовало наибольшее количество жителей из первой группы. Если таких жителей несколько, то первым консулом может стать любой из них.
- Вторым консулом станет житель, за которого проголосовало наибольшее количество жителей из второй группы. Если таких жителей несколько, то вторым консулом может стать любой из них.

Римляне хотят узнать, какая максимальная разность между номерами консулов может быть после выборов.

Однако жители могут менять свои голоса. А именно произойдет Q изменений: каждое изменение описывается парой чисел i, x и означает, что i -й житель теперь проголосует за x -го жителя. Изменения сохраняются.

Вам нужно после каждого изменения ответить на вопрос: какая может быть максимальная разность между номерами консулов.

Формат входных данных

В первой строке записаны целые числа N и Q ($2 \leq N \leq 2 \cdot 10^5$, $1 \leq Q \leq 2 \cdot 10^5$) — число жителей и число изменений голосов.

Во второй строке записаны N целых чисел a_1, a_2, \dots, a_N ($1 \leq a_i \leq N$) — за кого изначально голосует каждый житель.

В каждой из следующих Q строк записаны два целых числа i и x ($1 \leq i \leq N$, $1 \leq x \leq N$) — i -й житель меняет голос и теперь голосует за x -го жителя.

Формат выходных данных

Выведите Q строк. В j -й строке выведите одно целое число — максимальную разность между номерами консулов после применения первых j изменений.

Система оценки

Подзадача	Баллы	Ограничения
1	10	$N, Q \leq 100$
2	47	$N, Q \leq 3000$
3	43	$N, Q \leq 2 \cdot 10^5$

Примеры

стандартный ввод	стандартный вывод
5 3 1 2 3 4 5 3 4 1 2 5 2	4 3 2
8 5 8 1 4 2 5 4 2 3 7 4 8 4 4 1 5 8 8 4	4 4 4 7 7

Задача Н. LeftistAVL-дерево

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

LeftistAVL-дерево - это корневое бинарное дерево, в котором для любой вершины верно, что размер поддерева ее левого ребенка либо равен, либо на один больше, чем размер поддерева правого ребенка. Если какой-то ребенок отсутствует, то размер его поддерева считается равным нулю.

Жюри загадало LeftistAVL-дерево, и вам предстоит узнать его размер.

Все вершины в дереве пронумерованы, корень имеет номер 1, а остальные вершины имеют номер от 2 до 10^{100} включительно, при этом могут быть использованы не все номера.

Вы можете не более 70'000 раз сделать следующий запрос: для вершины с номером v узнать номера ее левого и правого ребенка.

Протокол взаимодействия

Что бы задать вопрос, выведите в отдельной строке «? v », где v -номер вершины, о которой вы хотите задать вопрос ($1 \leq v \leq 10^{100}$).

Если вершины с номером v не существует, то вы получите вердикт «Wrong answer».

Если же такая вершина существует, то в ответ на запрос вы получите в отдельной строке через пробел два числа l и r ($1 \leq l, r \leq 10^{100}$) - номера левого и правого ребенка вершины v .

Когда вы будете готовы вывести ответ на задачу (размер дерева), выведите строку «! n », где n - размер дерева ($1 \leq n \leq 10^{100}$).

После вывода запросов и ответа не забудьте сделать операция `flush`.

Система оценки

В этой задаче 6 подгрупп:

0. (0 баллов) Тест из условия
1. (5 баллов) $n \leq 20$
2. (15 баллов) $n \leq 70'000$
3. (40 баллов) $n \leq 10^{18}$
4. (20 баллов) $n \leq 10^{60}$
5. (20 баллов) $n \leq 10^{100}$

Пример

стандартный ввод	стандартный вывод
4 3	? 1
2 0	? 4
0 0	? 3
	! 4

Замечание

В примере дополнительные пустые строки во входных и выходных данных оставлены для удобства, вам не нужно выводить эти строки.

В этой задаче интерактор адаптивный, то есть структура дерева и номера вершин не зафиксированы до запуска программы. Однако гарантируется, что в любой момент существует хотя бы одно LeftistAVL-дерево, которое соответствует всем ответам на уже заданные вопросы.

Задача I. YDX

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

Дана строка длины n , составленная из букв Y, D и X. Буква Y встречается в строке строго чаще, чем D, и строго чаще, чем X.

Для каждой подстроки, состоящей из подряд идущих символов (от позиции ℓ до позиции r включительно, $1 \leq \ell \leq r \leq n$), известно число вхождений самого частого символа в этой подстроке. Самый частый символ может быть разным для разных подстрок и не обязан быть единственным (могут быть ничьи).

Оказывается, что по этой информации всегда можно однозначно восстановить все позиции, на которых стоит буква Y. Выведите их в порядке возрастания.

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 2000$) — длина строки.

Далее следуют n строк. В i -й строке записаны $n - i + 1$ целых чисел $M_{i,i}, M_{i,i+1}, \dots, M_{i,n}$ ($1 \leq M_{\ell,r} \leq n$), где $M_{\ell,r}$ — число вхождений самого частого символа в подстроку с (ℓ) -й по r -ю позицию включительно. Позиции нумеруются от 1 до n .

Гарантируется, что существует хотя бы одна строка из букв Y, D, X (с Y самой частой), согласованная с данными подсказками.

Формат выходных данных

Выведите одну строку: номера позиций всех вхождений буквы Y в порядке возрастания, через пробел. Каждая позиция — целое число от 1 до n .

Система оценки

Подзадача	Баллы	Ограничения
1	11	$n \leq 10$
2	12	В строке встречаются только буквы Y и D
3	10	В строке нет двух подряд одинаковых букв
4	11	$n \leq 40$
5	19	$n \leq 500$
6	37	Без дополнительных ограничений

Пример

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