

Задача А. Скидки на билеты

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Недавно Амир нашёл в тиктоке сервис покупки дешёвых авиабилетов «Авиа 102», который даёт возможность уменьшить стоимость перелёта в целых два раза! Но он немного не выспался и запутался в ценах.

Амиру нужно найти маршрут перелета с минимальной стоимостью из Судиславля в Магадан (почему именно туда Амир не уточняет). У него есть один 102-купон на скидку, с помощью которого он может уменьшить стоимость любого одного (и только одного) перелета на его маршруте вдвое.

Когда он использует 102-купон на скидку для перелета, стоимость которого равна x , его новая стоимость становится $\lfloor x/2 \rfloor$ (округляется вниз до целого числа).

Помогите Амиру найти маршрут минимальной стоимости из Судиславля в Магадан.

Формат входных данных

Первая строка содержит два целых числа n и m ($2 \leq n \leq 10^5$, $1 \leq m \leq 2 \cdot 10^5$) — количество городов и количество рейсов. Города пронумерованы числами от 1 до n . Город 1 — Судиславль, город n — Магадан. Каждая из следующих m строк содержит три целых числа a , b и c ($1 \leq a, b \leq n$, $1 \leq c \leq 10^9$) — начальный город рейса, конечный город рейса и его стоимость. Все рейсы являются однонаправленными. Гарантируется, что маршрут из Судиславля в Магадан всегда существует.

Формат выходных данных

Выведите одно целое число — минимальную стоимость маршрута из Судиславля в Магадан с учётом использования 102-купона.

Пример

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

Замечание

В примере оптимальный маршрут: $1 \rightarrow 2 \rightarrow 3$. Стоимость перелёта $1 \rightarrow 2$ равна 3, стоимость $2 \rightarrow 3$ равна 1. Применение 102-купона к первому рейсу снижает его стоимость до $\lfloor 3/2 \rfloor = 1$. Итоговая стоимость маршрута составляет $1 + 1 = 2$.

Задача В. Уменьшите диаметр

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

Дано неориентированное дерево, назовем расстоянием между двумя вершинами количество ребер на единственном простом пути между ними. Назовем диаметром дерева максимальное расстояние между парой вершин дерева. Назовем дерево хорошим, если его диаметр не превышает k .

Вы хотите удалить 0 или более вершин дерева, чтобы оно стало хорошим. Когда вы удаляете вершину, удаляются и все инцидентные ей ребра. Получившийся граф должен быть связным.

Найдите минимальное число вершин, которое можно удалить, чтобы получилось хорошее дерево.

Формат входных данных

Первая строка содержит два целых числа n и k ($2 \leq n \leq 2000$, $1 \leq k \leq n - 1$) — количество вершин дерева и максимальное значение диаметра.

Следующие $n - 1$ строк содержат целые числа v_i и u_i ($1 \leq v_i, u_i \leq n$) — ребра дерева. Гарантируется, что граф, образованный этими ребрами, является деревом.

Формат выходных данных

Выведите единственное целое число — минимальное число вершин, после удаления которых дерево станет хорошим.

Примеры

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

Замечание

В первом примере можно удалить вершины 5 и 6.

Во втором примере дерево изначально хорошее, и вершины удалять не надо.

Задача С. Монстры и лабиринт

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Вы и несколько монстров находитесь в прямоугольном лабиринте. Каждую минуту Вы можете сделать шаг в одном из направлений, при этом каждый из монстров также может одновременно с Вами сделать шаг в любом направлении. Ваша цель — добраться до одной из граничных клеток, ни в коем случае не деля клетку с монстром.

Выясните, достижима ли ваша цель, и если да, то выведите путь, по которому вы сможете пройти. Ваш план должен сработать в любой ситуации, даже если монстры знают его заранее.

Формат входных данных

Первая строка ввода содержит два целых числа n и m ($1 \leq n, m \leq 1000$) — высоту и ширину лабиринта.

После этого следуют n строк из m символов, описывающих карту. Каждый символ — это «.» (свободная клетка), «#» (стена), «А» (начало) или «М» (монстр). Во входных данных есть ровно одна буква А.

Формат выходных данных

Сначала выведите «YES», если ваша цель достижима, и «NO» в противном случае.

Если ваша цель достижима, также выведите пример допустимого пути (длина пути и его описание с использованием символов D, U, L и R). Вы можете вывести любой путь, если его длина не превышает $n \cdot m$ шагов.

Пример

стандартный ввод	стандартный вывод
5 8 ##### #M..A..# #.#.M#.# #M#..#.. #.#####	YES 5 RRDDR

Задача D. Сильносвязная ориентация

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Вам дан простой неориентированный граф из n вершин и m рёбер. Выберите на каждом ребре одну из двух возможных ориентаций так, чтобы результирующий ориентированный граф был сильно связным.

Формат входных данных

Первая входная строка содержит два целых числа n и m ($1 \leq n \leq 10^5$, $1 \leq m \leq 2 \cdot 10^5$) — количество вершин и рёбер. Вершины пронумерованы $1, 2, \dots, n$.

После этого идут m строк, описывающих ребра. Каждая строка содержит два целых числа a и b ($1 \leq a, b \leq n$) — между вершинами a и b проходит ребро.

Формат выходных данных

Выведите m строк, описывающих направления ребер.

Каждая строка должна содержать два целых числа a и b — от вершины a до вершины b проходит ребро. Вы можете вывести любое допустимое решение.

Если решений нет, выведите -1 .

Пример

стандартный ввод	стандартный вывод
3 3	1 2
1 2	2 3
1 3	3 1
2 3	

Задача Е. Исследование

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Вы собираетесь лететь из Москвы в Казань самолетом (возможно, посещая другие города на этом пути). Вы хотели бы получить ответы на следующие вопросы:

- какова минимальная стоимость такого маршрута?
- сколько существует маршрутов с минимальной стоимостью? (по модулю $10^9 + 7$)
- каково минимальное количество рейсов может быть на маршруте с минимальной стоимостью?
- каково максимальное количество рейсов может быть на маршруте с минимальной стоимостью?

Формат входных данных

Первая строка ввода содержит два целых числа n и m ($1 \leq n \leq 10^5$, $1 \leq m \leq 2 \cdot 10^5$) — количество городов и количество рейсов. Города пронумерованы $1, 2, \dots, n$. Город 1 называется Москва, а город n — Казань.

После этого следуют m строк с описанием рейсов. В каждой строке есть три целых числа a , b и c ($1 \leq a, b \leq n$, $1 \leq c \leq 10^9$) — есть рейс из города a в город b по цене c . Все рейсы выполняются в один конец.

Вы можете считать, что существует маршрут из Москвы в Казань.

Формат выходных данных

Выведите четыре целых числа, ответ на задачу.

Пример

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

Задача F. Очередные запросы достижимости

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Вам дан простой неориентированный граф с n вершинами и m ребрами.

Вы начинаете в определенной вершине и на каждом ходу должны переходить через ребро к соседней вершине.

Ваша задача — ответить на q запросов вида: «Возможно ли начать с вершины a и закончить в вершине b ровно через x ходов?»

Формат входных данных

Первая строка содержит три целых числа n , m и q ($2 \leq n \leq 2500$, $1 \leq m \leq 5000$, $1 \leq q \leq 10^5$) — количество вершин, ребер и запросов. Вершины пронумерованы $1, 2, \dots, n$.

После этого идут m строк, описывающих ребра. Каждая строка содержит два целых числа a и b ($1 \leq a, b \leq n$) — между вершинами a и b проходит ребро.

Наконец, есть q строк, каждая из которых описывает запрос. Каждая строка содержит три целых числа a , b и x ($1 \leq x \leq 10^9$).

Формат выходных данных

Для каждого запроса выведите ответ («YES» или «NO») в отдельной строке

Пример

стандартный ввод	стандартный вывод
4 5 6	YES
1 2	NO
2 3	YES
1 3	NO
2 4	YES
3 4	YES
1 2 2	
1 4 1	
1 4 5	
2 2 1	
2 2 2	
3 4 8	

Замечание

В запросе 1 возможным маршрутом является $1 \rightarrow 3 \rightarrow 2$.

В запросе 3 возможным маршрутом является $1 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4$.

В запросе 6 возможным маршрутом является $3 \rightarrow 4 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4$.

Задача G. Три строчки

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

Надзуна любит красить графы.

Имеется простой неориентированный граф, состоящий из n вершин, пронумерованных от 1 до n , и m рёбер. i -е ребро соединяется вершины v_i, u_i и имеет длину 1. Изначально все вершины окрашены в цвет 0.

Надзуна q раз выполнила операцию покраски в этом графе. i -я операция задаётся тремя числами v_i, d_i, c_i . В ходе операции она окрашивает все вершины на расстоянии не более d_i от вершины v_i в цвет c_i .

Найдите итоговый цвет каждой вершины.

Формат входных данных

В первой строке входных данных содержатся два числа n, m ($1 \leq n, m \leq 10^5$) — количество вершин и рёбер в графе.

В следующих m строках содержатся пары целых чисел v_i, u_i ($1 \leq v_i, u_i \leq n$) — рёбра графа. Гарантируется, что в графе нет петель и кратных рёбер.

В следующей строке входных данных содержится одно целое число q ($1 \leq q \leq 10^5$) — количество операций покраски.

В следующих q строках содержатся тройки целых чисел v_i, d_i, c_i ($1 \leq v_i \leq n, 1 \leq d_i \leq 10, 1 \leq c_i \leq 10^5$) — операции покраски в формате из условия.

Формат выходных данных

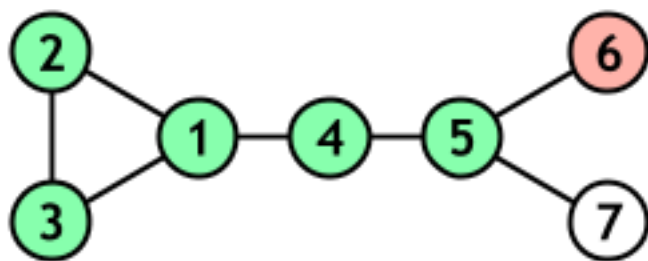
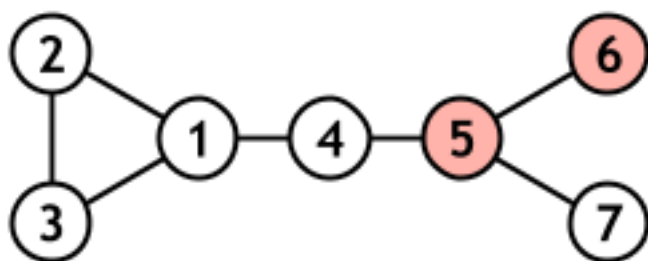
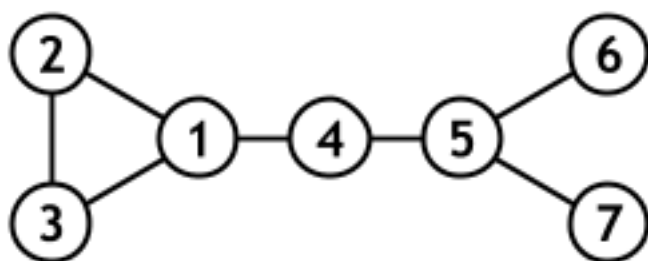
Выведите n строчек, в i из которых должен быть цвет i вершины после всех операций покраски.

Примеры

стандартный ввод	стандартный вывод
7 7 1 2 1 3 1 4 4 5 5 6 5 7 2 3 2 6 1 1 1 2 2	2 2 2 2 2 1 0
14 10 1 4 5 7 7 11 4 10 14 7 14 3 6 14 8 11 5 13 8 3 8 8 6 2 9 7 85 6 9 3 6 7 5 10 3 1 12 9 4 9 6 6 8 2 3	1 0 3 1 5 5 3 3 6 1 3 4 5 3

Замечание

Картинка к первому примеру из условия.



Задача Н. Классическая задача максимизации

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Вам даны $2n$ различных точек на плоскости. Точка i имеет координаты (x_i, y_i) .

Точки i и j называются *дружной парой* если выполняется $x_i = x_j$ или $y_i = y_j$.

Разбейте точки на n пар. Каждая точка должна принадлежать ровно одной паре. Количество дружных пар среди ваших n пар должно быть наибольшим возможным.

Формат входных данных

Первая строка содержит целое число t ($1 \leq t \leq 10^4$) — количество наборов входных данных в тесте.

Первая строка каждого набора содержит одно целое число n ($1 \leq n \leq 10^5$).

i -я из следующих $2n$ строк содержит два целых числа x_i и y_i ($-10^9 \leq x_i, y_i \leq 10^9$) — координаты i -й точки. Гарантируется, что все точки различны.

Гарантируется, что сумма n по всем наборам входных данных не превышает 10^5 .

Формат выходных данных

Для каждого набора тестовых данных выведите неотрицательное целое число k — максимальное количество дружных пар.

В i -й из следующих n строк выведите два целых числа a_i и b_i , означающих пару, образованную точками с номерами a_i и b_i ($1 \leq a_i, b_i \leq 2n$; $a_i \neq b_i$).

Каждое целое число от 1 до $2n$ должно быть среди a_i и b_i ровно один раз. Количество индексов i , таких что a_i и b_i образуют дружную пару, должно равняться k .

Пример

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