

Задача А. Ферзи

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

На доске $N \times N$ расставьте, пожалуйста, N обычных шахматных ферзей так, чтобы они друг друга не били.

Формат входных данных

В единственной строке входных данных содержится одно целое число N — размер доски ($4 \leq N \leq 200$).

Формат выходных данных

Для каждой горизонтали исходной доски выведите номер вертикали, на которой стоит ферзь в этой горизонтали, вертикали нумеруются слева направо, начиная с единицы.

Пример

стандартный ввод	стандартный вывод
4	2 4 1 3

Задача В. Пузыри

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

Девочка пускала мыльные пузыри в комнате размером $100 \times 100 \times 100$. Её очень забавлял тот факт, что пузыри являются идеальными сферами, но в конце концов это занятие наскучило девочке, и теперь она задалась вопросом, какая доля от общего объёма комнаты занята объединением пузырей? Во вселенной девочки пузыри могут пересекаться.

Формат входных данных

В первой строке записано единственное число n ($1 \leq n \leq 100$) — количество пузырей. В каждой из последующих n строк записаны четыре целых числа x_i, y_i, z_i и r_i ($1 \leq x_i, y_i, z_i, r_i \leq 99$) — координаты центра и радиус соответствующего пузыря. Гарантируется, что все пузыри целиком помещаются в комнате.

Формат выходных данных

Выведите вещественное число от 0 до 1 равное доле комнаты, занятой пузырями. Ваш ответ будет признан верным, если он будет отличаться от ответа жюри не более, чем на 0.01.

Примеры

стандартный ввод	стандартный вывод
1 50 50 50 40	0.268022
4 40 40 40 30 40 40 60 30 40 40 60 30 70 70 70 20	0.199398

Задача С. Хорошие раскраски

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

Назовем раскраску клеток таблицы $n \times m$ хорошей, если никакие четыре клетки, центры которых образуют вершины прямоугольника со сторонами, параллельными осям координат, не покрашены в один цвет. Иначе говоря, для раскраски не должно быть четверки целых чисел x_1, x_2, y_1, y_2 , что $1 \leq x_1 < x_2 \leq n, 1 \leq y_1 < y_2 \leq m$, и клетки $(x_1, y_1), (x_2, y_1), (x_1, y_2)$ и (x_2, y_2) покрашены в одинаковый цвет.

Требуется написать программу, которая по заданным целым числам n, m и c находит любую хорошую раскраску таблицы $n \times m$ в c цветов.

Формат входных данных

В первой строке записаны три целых числа n, m, c ($2 \leq n, m \leq 10, 2 \leq c \leq 3$).

Гарантируется, что для заданных во входных данных значений существует хотя бы одна хорошая раскраска.

Формат выходных данных

Выведите n строк по m чисел в каждой.

В качестве j -го числа i -й строки выведите $a_{i,j}$ — цвет клетки (i, j) ($1 \leq a_{i,j} \leq c$).

Если есть несколько хороших раскрасок, можно вывести любую из них.

Пример

стандартный ввод	стандартный вывод
2 2 2	1 2 2 2

Замечание

Примените какой-нибудь неточный алгоритм нахождения минимума функции, например алгоритм отжига.

Задача D. ЮграНефтеТранс

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

Ханты-Мансийский автономный округ — Югра является важнейшим нефтяным регионом России. Добыча нефти составляет 267 млн т в год, её транспортировка осуществляется по трубопроводам, общая длина которых превышает длину экватора Земли.

Система транспортировки нефти представляет собой совокупность n распределительных станций и m трубопроводов. Каждый трубопровод соединяет две различные станции. Между любыми двумя станциями проложено не более одного трубопровода.

Эффективность работы станций существенно зависит от вязкости нефти. Поэтому компания «ЮграНефтеТранс», в ведении которой находится сеть трубопроводов, заказала инновационному исследовательскому предприятию разработку и изготовление новых сверхточных датчиков вязкости на основе самых современных технологий.

Изготовление датчиков — процесс трудоёмкий и дорогостоящий, поэтому было решено изготовить k датчиков ($k \leq 40$) и выбрать k различных станций, на которых датчики будут установлены. Необходимо осуществить выбор станций так, чтобы датчики контролировали все трубопроводы: для каждого трубопровода хотя бы один датчик должен быть установлен на станции, где начинается или заканчивается этот трубопровод.

Напишите программу, которая проверяет, существует ли требуемое расположение датчиков, и в случае положительного ответа находит это расположение.

Формат входных данных

В первой строке входного файла записаны три натуральных числа — n , m и k ($k \leq n \leq 2000$, $1 \leq m \leq 10^5$, $1 \leq k \leq 40$). Далее следуют m строк, каждая из которых описывает один трубопровод. Трубопровод задаётся двумя целыми числами — порядковыми номерами станций, которые он соединяет. Станции пронумерованы от 1 до n . Гарантируется, что к любой станции подведён хотя бы один трубопровод и между любыми двумя станциями проложено не более одного трубопровода. Числа в каждой строке разделены пробелами.

Формат выходных данных

В первую строку выходного файла выведите слово **Yes**, если требуемое расположение датчиков существует, в противном случае — слово **No**. В случае положительного ответа выведите во вторую строку выходного файла k различных целых чисел — номера станций, на которых необходимо установить датчики. Номера можно выводить в любом порядке. Если существует несколько подходящих расположений датчиков, выведите любое из них. Разделяйте числа во второй строке пробелами.

Примеры

стандартный ввод	стандартный вывод
2 1 2 1 2	Yes 2 1
3 3 1 1 2 2 3 3 1	No
7 6 2 1 2 1 3 1 4 2 5 2 6 2 7	Yes 1 2
5 5 2 1 2 1 3 1 4 1 5 4 5	Yes 4 1

Задача E. Folding

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

Маленький Саша очень любит оригами. Он проводит дни и ночи складывая из листа бумаги разные странные фигуры. Когда Саша начинает собирать новую фигуру, он делается следующее:

- Берёт квадратный лист бумаги размера $n \times n$ и кладёт его на координатную плоскость, так что центр квадрата совпадает с началом координат $(0, 0)$, а стороны квадратного листа параллельны осям координат.
- Производит несколько складываний.

Каждое складывание устроено следующим образом. Саша выбирает **ориентированную** прямую l , берёт часть фигуры, расположенную по левую сторону от прямой l и кладёт её на часть фигуры, расположенную справа от прямой l . Таким образом, слева от прямой не остаётся никаких частей фигуры, а справа находится объединение того что находилось справа до сгиба, и того что находилось до сгиба слева, отражённое относительно прямой l .

Саша сделал k сгибов и теперь хочет знать, сколько процентов от площади изначального листа составляет площадь итоговой фигуры. Помогите ему найти это значение.

Формат входных данных

В первой строке входных данных записаны два целых числа n и k ($1 \leq n \leq 100$, $1 \leq k \leq 5$), длина стороны исходного квадрата и количество выполненных Сашей сгибов.

Далее следуют k строк с описаниями сгибов. Каждое описание состоит из четырёх целых чисел x_1, y_1, x_2, y_2 ($-100 \leq x_1, x_2, y_1, y_2 \leq 100$), который определяют **ориентированную** прямую сгиба l . Гарантируется, что точки (x_1, y_1) и (x_2, y_2) различны.

Левая сторона фигуры накладывается на правую сторону. Чтобы определить какая сторона является левой, а какая правой, представьте что вы стоите в точке (x_1, y_1) и смотрите в направлении точки (x_2, y_2) , тогда левой частью будет являться всё что находится по левую руку, а правой то что находится по правую. Обратите внимание, что в таком определении имеет значение порядок точек (x_1, y_1) и (x_2, y_2) во входных данных.

Может возникнуть ситуация что прямая l и вовсе не пересекает фигуру. Такой случай обрабатывается в соответствии с определением сгиба: если фигура находится по левую сторону от l , то она просто зеркально отражается относительно прямой, а если фигура находится справа от l , то ничего не происходит.

Формат выходных данных

В единственной строке выведите целое число от 0 до 100, определяющее сколько процентов составляет площадь итоговой фигуры от площади изначального листа.

Ваш ответ будет считаться верным, если точный ответ (без округления до целого числа процентов) отличается от вашего не более чем на 1.

Примеры

стандартный ввод	стандартный вывод
10 3 0 0 5 5 0 0 0 5 -5 5 -5 -5	38
10 1 -5 -5 1 4	67

Задача F. Лёгкий Банах

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 512 мегабайт

Стефан, Дэвид и Банах готовятся принять участие в командном чемпионате по программированию. Стефан предлагает попрактиковаться на следующей задаче.

Вам даны n точек на плоскости и n векторов сдвига. Требуется найти взаимно-однозначное соответствие векторов и точек, такое что после сдвига каждой из точек на соответствующий ей вектор, для любой пары точек евклидово расстояние между ними не уменьшится.

Дэвид сказал, что Стефан мог бы придумать задачу и посложнее, вы согласны с ним?

Формат входных данных

В первой строке входных данных записано целое число n ($1 \leq n \leq 500$), равное количеству точек и количеству векторов.

Следующие n строк содержат описание точек. В каждой из них записаны два целых числа px_i и py_i ($0 \leq |px_i|, |py_i| \leq 10\,000$).

Далее следуют n строк с описанием векторов сдвига. Каждый вектор определяется двумя целыми числами vx_i и vy_i ($0 \leq |vx_i|, |vy_i| \leq 10\,000$).

Формат выходных данных

Если существует подходящее попарное соответствие точек и векторов сдвига, такое что расстояние между каждой парой точек не уменьшится, то выведите "Yes" в первой строке вывода. В следующей строке выведите перестановку чисел от 1 до n , i -е из этих чисел должно означать какой вектор будет назначен i -й точке.

Если подходящих решений несколько, то выведите любое из них.

Если подходящего соответствия не существует, то выведите "No" в единственной строке вывода.

Примеры

стандартный ввод	стандартный вывод
2 0 0 1 0 2 0 -2 0	Yes 2 1
2 2 2 2 2 -1 -1 -1 -1	Yes 1 2
3 -2 -3 3 1 -3 -3 -3 5 -2 -4 4 -1	Yes 2 3 1

Задача G. Ферзи HARD

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	10 секунд
Ограничение по памяти:	512 мегабайт

Дана шахматная доска размера N и K ферзей, которые уже расставлены на доске так, что они не бьют друг друга. При этом ферзи в данной задаче необычные – кроме стандартных движений по вертикали, горизонтали и диагонали они также могут ходить как шахматный конь на произвольное число шагов. То есть, если ферзь находится на позиции (x, y) , он может пойти в позицию $(x + n, y + 2n)$, $(x + 2n, y + n)$, $(x - n, y + 2n)$, $(x - 2n, y + n)$ для любого целого (не обязательно положительного) числа n (позиция должны быть в пределах доски).

Требуется расставить на доске оставшиеся $N - K$ ферзей так, чтобы в результате никакие два ферзя не били друг друга.

Формат входных данных

Первая строка входных данных содержит два целых числа N и K – размер доски и количество уже расставленных ферзей ($1 \leq N \leq 10000$, $0 \leq K \leq N/2$). В следующих K строках содержатся пары целых чисел (x_i, y_i) , описывающие текущую расстановку ферзей на доске ($1 \leq x_i, y_i \leq N$).

Формат выходных данных

Первая строка выходных данных должна содержать одно слово YES или NO, означающее можно ли расставить ферзей на доске, так чтобы они не били друг друга. В случае, если ферзей расставить можно, дальше должны идти N чисел y_1, y_2, \dots, y_N , каждое на своей строке. Число y_i должно описывать столбец, в котором стоит ферзь на строке i . Полученное решение должно быть согласовано с исходной расстановкой K ферзей.

Примеры

стандартный ввод	стандартный вывод
10 0	YES 7 3 10 6 2 9 5 1 8 4
10 1 1 1	NO