

Задача А. Ретроанализ для маленьких

Имя входного файла: `retro.in`
Имя выходного файла: `retro.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан ориентированный весёлый граф из n вершин и m ребер. Оля и Коля в игру. Изначально фишка стоит в вершине i . За ход можно передвинуть фишку по любому из исходящих ребер. Тот, кто не может сделать ход, проигрывает. Ваша задача — для каждой вершины i определить, кто выиграет при оптимальной игре обоих.

Формат входных данных

Входные данные состоят из одного или нескольких тестов. Каждый тест содержит описание весёлого ориентированного графа. Граф описывается та: на первой два целых числа n ($1 \leq n \leq 300\,000$) и m ($1 \leq m \leq 300\,000$). Следующие m строк содержат ребра графа, каждое описывается парой целых чисел от 1 до n . Пара $a\ b$ обозначает, что ребро ведет из вершины a в вершину b . В графе могут быть петли, могут быть кратные ребра. Сумма n по всем тестам не превосходит 300 000, сумма m по всем тестам также не превосходит 300 000.

Формат выходных данных

Для каждого теста выведите для каждой вершины `FIRST`, `SECOND` или `DRAW` в зависимости от того, кто выиграет при оптимальной игре из этой вершины. Ответы к тестам разделяйте пустой строкой.

Примеры

<code>retro.in</code>	<code>retro.out</code>
5 5	DRAW
1 2	DRAW
2 3	DRAW
3 1	FIRST
1 4	SECOND
4 5	FIRST
2 1	SECOND
1 2	FIRST
4 4	FIRST
1 2	SECOND
2 3	SECOND
3 1	
1 4	

Задача В. Шоколадка

Имя входного файла: `choco.in`
Имя выходного файла: `choco.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Двое играют в такую игру: перед ними лежит шоколадка размера $N \times M$. Игроки ходят по очереди. За ход разломить любой имеющийся кусок шоколадки на 2 «непустых» куска, при этом запрещено ломать куски размером не больше, чем $1 \times S$ (т.е. нельзя ломать куски, у которых один размер равен 1, а другой не превосходит S), куски можно поворачивать. Ломать, конечно, можно только вдоль линий, нанесенных на шоколадке, т.е. после разлома должны получаться два прямоугольника с целочисленными ненулевыми сторонами.

Проигрывает тот, кто не может сделать хода.

Формат входных данных

Во входном файле находятся три целых числа N , M и S ($0 < N, M, S \leq 100$).

Формат выходных данных

Выведите в выходной файл одно число 1 или 2 — номер игрока, который выигрывает при правильной игре.

Примеры

<code>choco.in</code>	<code>choco.out</code>
1 1 1	2
2 2 1	1

Задача С. Шашки

Имя входного файла: `checkers.in`
Имя выходного файла: `checkers.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Как известно, в ЛКШ каждую смену проходит турнир по шашкам. В ЛКШ.Зима организаторы тоже хотели провести такой турнир, но проводить его по стандартным правилам было бы банально. Культурги полезли в Википедию и выяснили, что шашки имеют много разновидностей. Одна из них, «Новогодняя», абсолютно не похожа на привычные нам шашки. Игра происходит на прямоугольной доске $n \times m$. В каждой из n строк расположено ровно 2 шашки (одна белая, другая черная, при этом белая всегда стоит левее черной). За один ход игрок может взять любую шашку своего цвета и передвинуть ее на любую клетку этой же строки, находящуюся ближе к шашке соперника, но не перепрыгивая через нее. Проигрывает тот, кто не может сделать ход. Найдите, кто выиграет при правильной игре.

Формат входных данных

Во входном файле записаны через пробел два числа — n и m ($1 \leq n \leq 10^5$, $2 \leq m \leq 10^9$). Далее в n строках записаны по два целых числа — l и r ($1 \leq l < r \leq m$) — расположение белой и черной шашки в i -ой строке соответственно.

Формат выходных данных

В выходной файл выведите «**First**», если у первого игрока есть выигрышная стратегия и «**Second**» в противном случае.

Примеры

<code>checkers.in</code>	<code>checkers.out</code>
1 10 1 5	First
2 10 1 5 1 5	Second

Задача D. Дровосек

Имя входного файла: `woodcut.in`
Имя выходного файла: `woodcut.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Двое играют в следующую игру: имеется дерево с отмеченной вершиной (корнем). Игроки ходят по очереди. За ход игрок разрубает ветку (стирает ребро), причем из двух получившихся компонент связности остается только та, которая содержит корень — остальная отваливается и больше в игре не участвует. Проигрывает тот, кто не может сделать ход.

Определите, может ли выиграть первый игрок, и если да, то укажите любой из его выигрышных ходов.

Формат входных данных

В первой строке входного файла находится 2 числа N и R — количество вершин дерева и номер корня ($2 \leq N \leq 1000\,000$, $1 \leq R \leq N$). Далее следует $N - 1$ строк, в каждой из которых находятся два числа — номера вершин, которые соединяет очередное ребро.

Формат выходных данных

Выведите в выходной файл одно число: 1 или 2 — номер игрока, который выигрывает при правильной игре. Если выигрывает первый игрок, то выведите также любой его выигрышный ход, т.е. порядковый номер ребра во входном файле, которое ему достаточно разрубить первым ходом (число от 1 до $N - 1$).

Примеры

<code>woodcut.in</code>	<code>woodcut.out</code>
5 5	1
2 3	1
1 3	
2 5	
4 5	

Задача Е. Игра с окружностью

Имя входного файла: `circlegame.in`
Имя выходного файла: `circlegame.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

На листе бумаги нарисована окружность, на которой отмечено n различных точек. Гидран и Диореп играют в игру, делая ходы по очереди. Ход состоит в том, чтобы выбрать две различные отмеченные точки и провести между ними хорду; это можно сделать, только если новая хорда не имеет общих точек (в том числе и концов) со всеми ранее проведёнными хордами. Проигрывает тот, кто не может сделать ход. Первым ходит Гидран. Кто выигрывает при правильной игре?

Формат входных данных

На ввод подаётся несколько строк. Первая из них содержит целое число t — количество тестов ($1 \leq t \leq 50$). Каждая из следующих t строк описывает один тест. Описание теста состоит из одного целого числа n — количества отмеченных точек на окружности ($3 \leq n \leq 10^9$).

Формат выходных данных

Для каждого заданного n выведите на отдельной строке “53”, если выигрывает Гидран, и “34”, если выигрывает Диореп. Ответы должны быть выведены в том же порядке, в котором числа n заданы во вводе.

Примеры

	<code>circlegame.in</code>	<code>circlegame.out</code>
2		53
4		34
5		

Задача F. Битва за кольцо

Имя входного файла: `rings.in`
Имя выходного файла: `rings.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Саруман Белый и Гэндальф Серый решили сыграть в игру. Победителю достается Кольцо Всевластия. Перед игроками лежат кольца, соединенные в K цепочек. Для каждого кольца известно содержание золота в нем в процентах — целое число от 1 до 100. Ходят по очереди. За ход разрешается выбрать одну из цепочек и какое-то кольцо из этой цепочки и дематериализовать все кольца из данной цепочки с процентным содержанием золота не больше, чем у выбранного. При этом, понятно, цепочка может распасться на несколько. Игра продолжается на оставшихся цепочках. Тот, кто дематериализовал последнее кольцо, выиграл. Первым ходит Гэндальф. Определите, может ли Гэндальф выиграть и, если может, какой первый ход он должен для этого сделать.

Формат входных данных

В первой строке дано целое число K ($1 \leq K \leq 50$). В следующих K строках приведены описания цепочек в следующем формате: сперва дана длина цепочки — целое число от 1 до 100, затем — процентные содержания золота в кольцах цепочки. Числа в строке разделены пробелом.

Формат выходных данных

Выведите "S", если Кольцо Всевластия достанется Саруману. В противном случае выведите в первой строке "G", а во второй пару чисел, описывающих выигрышный первый ход Гэндальфа — номер цепочки и номер кольца в ней. Цепочки и кольца внутри цепочек нумеруются с 1. Если существует несколько выигрышных первых ходов, выведите ход с наименьшим номером цепочки, если и таких несколько — с наименьшим номером кольца.

Примеры

<code>rings.in</code>	<code>rings.out</code>
2 3 1 2 1 1 1	G 1 1
2 3 2 1 2 1 1	S