

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 играют 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 так: 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. Полоска бумаги

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

Артем и Саша на скучной ЛКШатской лекции решили поиграть в интересную игру. Они берут клетчатую полоску бумаги $1 \times N$ и пишут в каждой клеточке некоторое натуральное число. После этого каждый из них по очереди может согнуть листок вдоль какой-нибудь линии, разделяющей клеточки, и затем положить согнутый лист лицом вверх так, чтобы сверху оказалась та часть листа, которая покрывает другую целиком. Если обе стороны согнутого листа могут оказаться сверху, игрок имеет право выбора.

В конце игры лицевая сторона листа представляет собой одну клеточку. Число, написанное в этой клеточке, является результатом игры. Артем всегда старается максимизировать результат, а Саша — минимизировать. Артем ходит первым.

Определите результат игры, если Артем и Саша всегда ходят оптимально.

Формат входных данных

Первая строка содержит натуральное число N ($1 \leq N \leq 100$) — длина полоски. В следующей строке содержится N натуральных чисел, не превышающих 10^9 — числа, записанные на полоске.

Формат выходных данных

В выходной файл выведите одно натуральное число — результат игры.

Примеры

<code>paper.in</code>	<code>paper.out</code>
3 1 2 1	1

Задача С. Малыш и Карлсон

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

На свой День рождения Малыш позвал своего лучшего друга Карлсона. Мама испекла его любимый пирог прямоугольной формы $a \times b \times c$ сантиметров. Карлсон знает, что у Малыша еще есть килограмм колбасы. Чтобы заполучить ее, он предложил поиграть следующим образом: они по очереди разрезают пирог на две ненулевые по объему прямоугольные части с целыми измерениями и съедают меньшую часть (в случае, когда части равные, можно съесть любую). Проигрывает тот, кто не может сделать хода (то есть когда размеры будут $1 \times 1 \times 1$). Естественно, победителю достается колбаса.

Малыш настаивает на том, чтобы он ходил вторым.

Помогите Карлсону выяснить, сможет ли он выиграть, и если сможет — какой должен быть его первый ход для этого.

Считается, что Малыш всегда ходит оптимально.

Формат входных данных

Во входном файле содержится 3 целых числа a, b, c ($1 \leq a, b, c \leq 5000$) — размеры пирога.

Формат выходных данных

В случае, если Карлсон не сможет выиграть в Малыша, выведите NO. В противном случае в первой строке выведите YES, во второй — размеры пирога после первого хода Карлсона в том же порядке, что и во входном файле.

Примеры

<code>karlsson.in</code>	<code>karlsson.out</code>
1 1 1	NO
1 2 1	YES 1 1 1
1 1 10	YES 1 1 7

Задача D. Демиурги

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

Демиурги Шамбамбукли и Мазукта любят на досуге поиграть в какую-нибудь игру. Но обычно Мазукту раздражает, что Шамбамбукли слишком долго думает над своим ходом. Поскольку в прошлый раз людские «шахматы» Шамбамбукли просчитывал целую миллисекунду, на этот раз Мазукта решил предложить ему чтонибудь попроще.

Самая простая игра, которую знает Мазукта — это игра «ним». Но, чтобы не обидеть Шамбамбукли излишней простотой игры, он решил внести долю интриги. Перед собственно разбором кучек игроки «торгуются» о наборе кучек. Сперва Мазукта называет натуральное число a . После этого Шамбамбукли называет еще одно натуральное число b . В результате игра происходит на наборе кучек, содержащих $\min(a, b)$, $\min(a, b) + 1, \dots, \max(a, b)$. Например, при $a = 5$ и $b = 2$ демиурги играют на кучках из 2, 3, 4 и 5 камней соответственно, и Мазукта ходит первым.

Помогите Шамбамбукли выбрать свое число так, чтобы точно обыграть вредного Мазукту.

Формат входных данных

Первая строка входного файла содержит единственное натуральное число a ($1 \leq a \leq 10^9$) — число, названное Мазуктой.

Формат выходных данных

В выходной файл выведите единственное натуральное число, которое должен назвать Шамбамбукли, чтобы выиграть. Если такого числа не существует, выведите 0.

Примеры

<code>demiurgs.in</code>	<code>demiurgs.out</code>
2	5

Задача Е. 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 F. 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 разрушает 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 ($2 \leq N \leq 100\,000, 1 \leq R \leq N$). 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	