

Задача А. Огромный ним

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

Петя и Вася играют в ним, но не простой, а просто огромный. У них есть очень много кучек камней. Кучки разделены на n групп. Группа i состоит из кучек размеров от l_i до r_i включительно. Помогите ребятам понять, кто выиграет при оптимальной игре

Формат входных данных

Первая строка входного файла содержит число n ($1 \leq n \leq 10^5$), следующие n строк содержат пары чисел l_i, r_i ($1 \leq l_i \leq r_i \leq 10^{18}$).

Формат выходных данных

Если первый игрок проигрывает, выведите `Lose`, если выигрывает — выведите в первой строке `Win`, а во второй строке — любой выигрышный ход для первого игрока. Ход задается размером кучки до хода и после него.

Примеры

<code>big-nim.in</code>	<code>big-nim.out</code>
1 1 10	Win 8 3
2 2 5 2 5	Lose

Задача В. Королевская игра

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

Император Шардонии и король Флатляндии во время своей встречи решили сыграть в новую настольную игру. Разумеется, играть они будут шариками на клетчатом плоском поле.

Поле для игры представляет из себя клетчатый прямоугольник, строки которого пронумерованы от 0 до 100, а столбцы — от 0 до 100. По некоторым клеткам поля будут раскиданы N шариков. Игроки ходят по очереди. В свой ход игрок может взять шарик с позиции (l_i, c_i) и выбрать положительное целое число u . После этого игрок может передвинуть этот шарик в одну из следующих ячеек:

- $(l_i - u, c_i)$
- $(l_i, c_i - u)$
- $(l_i - u, c_i - u)$

Разумеется, ход можно сделать только если соответствующая клетка существует на поле. Выигрывает игрок, который смог передвинуть какой-либо шарик в позицию $(0, 0)$.

Вы — главный советник императора Шардонии. Он вызвался ходить первым, и спрашивает вас, может ли он выиграть, если оба игрока будут играть оптимально. Ответьте ему на вопрос, потому что иначе вас казнят.

Формат входных данных

На вход программе подается число N ($1 \leq N \leq 1000$) — количество шариков на поле.

В следующих N строках расположены числа l_i, c_i ($0 \leq l_i, c_i \leq 100$) — координаты шариков.

Формат выходных данных

Вы должны вывести символ «Y», если Император может выиграть, и «N» иначе.

Примеры

стандартный ввод	стандартный вывод
2 1 3 2 3	Y
1 1 2	N

Задача С. Вас снова замяукали!

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

Два котёнка попали в запутанный лабиринт со множеством комнат и переходов между ними. Котят долго по нему плутали, обошли все комнаты по много раз, нашли выход (да даже и не один, а несколько), в общем, изучили там всё, что смогли. Теперь этот лабиринт котят используют в своих играх.

Чаще всего котят играют в следующую игру: начиная в какой-то комнате лабиринта, котят поочередно выбирают, в какую из комнат им перейти. Котят изначально находятся в одной комнате и ходят вместе. Как только котёнок, который должен выбрать следующую комнату, не может этого сделать, он признаётся проигравшим. Обычно в таких играх выигрывающий игрок стремится выиграть как можно быстрее, а проигрывающий стремится как можно дольше оттянуть свое поражение. Но у котят свои представления о победе и поражении. Если котёнок знает, что, начиная из текущей комнаты, он выиграет (вне зависимости от действий другого котёнка), то он стремится играть как можно дольше, чтобы продлить себе удовольствие от выигрыша (естественно, при этом выигрывающий котёнок должен гарантировать себе, что будет постоянно уверен в выигрыше). Котёнок, который знает, что проиграет (при условии, конечно, что другой котёнок будет действовать оптимально), старается проиграть как можно быстрее, чтобы начать новую игру, в которой и взять реванш.

Если котят будут ходить бесконечно долго, но никто из них не сможет выиграть, то котят считают игру завершившейся вничью и замяукивают Вас.

Вас попросили для каждой комнаты в лабиринте узнать, выиграет или проиграет котёнок, начинающий ходить из данной комнаты. Если котёнок, начинающий из этой комнаты, выигрывает, требуется узнать максимальное количество ходов, которое он сможет играть, если же проигрывает — минимальное количество, которое ему придётся играть.

Формат входных данных

В первой строке ввода находятся два числа n и m — число комнат и переходов между комнатами в лабиринте ($1 \leq n \leq 100\,000$, $0 \leq m \leq 100\,000$). Далее следует m строк с описаниями переходов. Описание перехода состоит из двух чисел a и b , означающих, что котёнок, начинающий игру в комнате с номером a , может выбрать комнату b в качестве следующей.

Формат выходных данных

Выведите n строк — для каждой комнаты результат игры для котёнка, который начнет игру из этой комнаты. Если игра закончится вничью, выведите «DRAW». Если начинающий котёнок выиграет, выведите «WIN K », где K — количество ходов, которые сможет играть выигрывающий котёнок. Если котёнок сможет играть сколь угодно долго, сохраняя возможность в любой момент выиграть, выведите «WIN INF». Если котёнок, начинающий из этой комнаты, проиграет, выведите «LOSE K », где K — количество ходов, которые придется играть проигрывающему котёнку. Если же котёнку придется играть сколь угодно долго, при том, что его соперник сможет в любой момент выиграть, выведите «LOSE INF».

Примеры

стандартный ввод	стандартный вывод
4 4	LOSE 2
1 2	WIN 1
1 3	WIN 1
2 4	LOSE 0
3 4	
6 6	DRAW
1 2	DRAW
2 3	DRAW
3 4	DRAW
4 1	WIN 1
4 5	LOSE 0
5 6	

Задача D. Битва за кольцо

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
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

стандартный ввод	стандартный вывод
2 3 1 2 1 1 1	G 1 1
2 3 2 1 2 1 1	S
1 10 1 2 3 4 5 6 7 8 9 10	G 1 10

Задача Е. Дровосек

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
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 ($2 \leq n \leq 100\,000$, $1 \leq r \leq n$). 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

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

Задача F. Малыш и Карлсон

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

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

стандартный ввод	стандартный вывод
1 1 1	NO
1 2 1	YES 1 1 1
1 1 10	YES 1 1 7

Задача G. Ретроанализ

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.3 секунд
Ограничение по памяти:	256 мегабайт

Дан ориентированный весёлый граф из n вершин и m ребер. Двое 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.

Пример

стандартный ввод	стандартный вывод
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 Н. Игра на графе

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

Геннадий и Георгий играют в интересную игру на ориентированном графе. В графе n вершин и m рёбер, возможны петли. Перед началом игры Геннадий и Георгий поставили в какую-то вершину фишку. Игроки ходят по очереди, перемещая фишку вдоль какого-то ребра, исходящего из вершины, в которой она находится. Если у игрока нет ни одного хода, он проигрывает.

Ваша задача — для каждой вершины графа и для обоих игроков определить результат игры, если фишка стоит в этой вершине и если этот игрок ходит первым. Звучит просто? Всё не совсем так.

С одной стороны, Геннадий получает удовольствие от процесса игры, и хочет, чтобы игра длилась как можно дольше. Он даже предпочтёт бесконечную игру победе. Но если он не может играть бесконечно, он, очевидно, предпочтёт поражению победу.

С другой стороны, у Георгия куча дел, и он не хочет играть бесконечно. Георгий хочет победить, но если он не может, он предпочтёт поражение бесконечной игре.

Оба игрока играют оптимально. Оба игрока знают предпочтения противника.

Формат входных данных

В первой строке записано два целых числа — число вершин n ($1 \leq n \leq 100\,000$) и число рёбер m ($1 \leq m \leq 200\,000$). В следующих m строках записаны по два числа a и b , обозначающих ребро из вершины a в вершину b . Вершины пронумерованы от 1 до n . Каждая пара (a, b) встречается не более одного раза.

Формат выходных данных

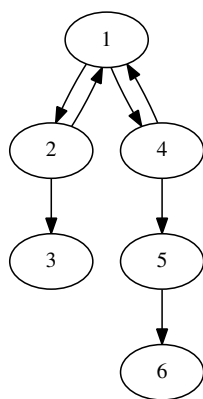
В первой строке выведите n символов — i -й символ означает результат игры, если Геннадий начинает в вершине i . Во второй строке выведите n символов — i -й символ означает результат игры, если Георгий начинает в вершине i . Результат игры обозначается буквой «W», если начинающий игру игрок побеждает, «L» — если проигрывает, и «D», если игра продолжается бесконечно.

Пример

стандартный ввод	стандартный вывод
6 7	WDLDWL
1 2	DWLLWL
2 1	
2 3	
1 4	
4 1	
4 5	
5 6	

Замечание

Вершины 3 и 6 изначально проигрывают. В вершине 5 единственный ход ведёт в вершину 6, и игрок выигрывает. Если Георгий начинает в вершине 1, или если Геннадий начинает в вершинах 2 или 4, Геннадий может всегда пойти в вершину 1 и сделать игру бесконечной. Если Георгий начинает в вершине 4, он может пойти или в вершину 1 (что приведёт к ничьей) или в вершину 5, что приведёт к поражению. Георгий предпочитает проиграть. Аналогично, из вершины 2, он предпочитает пойти в вершину 3 и победить. Из вершины 1, Геннадий может пойти в вершину 2 и проиграть, или в вершину 4 и победить. Он предпочитает победить.



Задача I. Ним в поддавки

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

Всем вам хорошо известна игра ним: на столе лежит кучка из a камней, своим ходом можно взять из неё любое число камней от 1 до a . Однако, эта игра необычная: если обычно игрок, который не может сделать ход, проигрывает, здесь всё наоборот — игрок, который не может сделать ход.

Как и в привычном вам варианте игры, на столе лежит не одна кучка камней, а n . Каждый игрок своим ходом может взять из любой кучки любое число камней. Вам нужно определить, кто победит в этой игре.

Формат входных данных

В первой строке задано целое число n ($1 \leq n \leq 100$). Во второй строке заданы размеры кучек a_i ($1 \leq a_i \leq 1000$).

Формат выходных данных

В единственной строке выведите «FIRST», если победит первый игрок, или «SECOND», если победит второй.

Примеры

стандартный ввод	стандартный вывод
2 1 2	FIRST
1 1	SECOND
2 9 4	FIRST

Задача J. Не-не-не-нечестная игра

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

Дэвид Блейн и Вася сидят на уроке математики и скучают. Через некоторое время Дэвид Блейн предлагает поиграть в следующую игру:

- он записывает на листочке некоторое большое целое число и ходит первым;
- игроки ходят по очереди;
- на каждом ходу игрок обязан разделить текущее число на его простой делитель p . На этом же ходу игрок может умножить (а может и нет) результат деления на простое число q ($1 < q < p$);
- проигрывает тот, кто не может сделать ход.

Мягко говоря, Вася не очень доверяет Дэвиду Блейну и боится, что тот выписывает только проигрышные для Васи начальные числа. Помогите ему определить, так это или нет.

Формат входных данных

Первая строка входного файла содержит единственное число n ($1 \leq n \leq 10^{12}$).

Формат выходных данных

В выходной файл выведите `Vasya`, если у Васи есть выигрышная стратегия, независимо от ходов Дэвида Блейна. Иначе выведите `David`.

Задача К. Странная игра

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

Двое играют в простую игру на доске $n \times n$. У первого игрока есть одна белая фишка, а у второго — одна чёрная. Игроки ходят по очереди, первым ходит первый игрок (белые).

Первый игрок имеет право двигать свою фишку на одну клетку в одном из четырёх основных направлений (влево, вправо, вверх, вниз). Второй игрок при своем ходе также выбирает одно из этих четырёх направлений, но может передвинуть свою фишку как на одну клетку в этом направлении, так и на две. Выигрывает тот, кто первым съедает фишку соперника.

Определите победителя и число ходов, требуемое для победы, при оптимальной игре сторон.

Формат входных данных

Во входном файле даны пять чисел — n ($2 \leq n \leq 20$), а также координаты белой и чёрной фишек. Фишки стоят в разных клетках доски.

Формат выходных данных

Выведите `WHITE` x , если выигрывают белые, `BLACK` x , если выигрывают чёрные, `DRAW`, если игра закончится вничью. Здесь x — число ходов обеих сторон (полуходов) до момента окончания игры.

Примеры

стандартный ввод	стандартный вывод
20 2 20 20 19	BLACK 54
19 2 9 19 1	BLACK 56

Задача L. Смак

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

Правильно ли я понимаю, что наше рабочее место на данной кулинарной передаче ограничено одним столом, одним тестом и тремя мисками? Вы же понимаете, что мы просто физически здесь не поместимся?

Михаил Аноприенко

Иван Ургант пригласил Ивана Сафонова на свою кулинарную передачу «Смак». В сегодняшнем выпуске они будут готовить огромный пирог из теста, которое уже заготовлено помощниками Урганта и налито в миску. В миске всегда находится целое неотрицательное число граммов теста. Оба Ивана, начиная с Урганта, по очереди применяют один из двух кулинарных приёмов:

1. Набрать мерным стаканом один грамм теста из миски и добавить в форму для пирога.
2. Взбить тесто миксером. При этом часть теста расплёскивается. Если в миске перед взбиванием было G граммов теста, то останется $\varphi(G)$ граммов теста. φ — это функция Эйлера, которая по данному положительному целому числу G возвращает количество целых чисел от 1 до G включительно, которые взаимно просты с G , то есть не имеют с G общих делителей, больших единицы. Приведём таблицу, в которой вычислены некоторые значения этой функции:

G	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$\varphi(G)$	1	1	2	2	4	2	6	4	6	4	10	4	12	6	8	8	16	6	18	8

Когда тесто в миске полностью закончится, пирог будет считаться приготовленным; более того, тот из Иванов, чей приём потратил последний грамм теста в миске, и будет считаться поваром, приготовившим пирог. Считая, что оба Ивана желают считаться создателями пирога и будут для этого каждый раз оптимально выбирать один из двух приёмов, определите, чей кулинарный приём завершит готовку.

Формат входных данных

В первой строке находится одно целое положительное число G ($1 \leq G \leq 10^9$) — изначальное количество теста в граммах.

Формат выходных данных

Если Иван Ургант может гарантированно завершить готовку пирога, в первой строке выведите «Ivan Urgant». Во второй строке выведите «Draw», если для этого Урганту нужно первым действием набрать грамм теста из миски и налить в форму, или же выведите «Mix», если первым действием ему надо взбить тесто. Если Ургант при выполнении любого из двух приёмов может гарантировать, что именно он завершит готовку пирога, то вы можете вывести любое одно из двух слов «Draw» и «Mix».

Если Иван Сафонов может гарантированно завершить готовку пирога, в первой строке выведите «Ivan Safonov».

Примеры

стандартный ввод	стандартный вывод
1	Ivan Urgant Draw
2	Ivan Safonov
4	Ivan Urgant Mix
14	Ivan Urgant Draw

Замечание

Если в миске один грамм теста, Ургант просто должен налить его в форму и сразу завершить готовку пирога.

Если в миске два грамма теста, то любой из двух кулинарных приёмов Урганта приведёт к тому, что в миске останется один грамм теста, после чего Сафонов легко доготовит пирог.

Если в миске четыре грамма теста, то Ургант проиграет, если нальёт один грамм теста в форму, и выиграет, если вместо этого он взобьёт тесто и оставит в миске $\varphi(4) = 2$ грамма.

Если в миске 14 граммов теста, то всё наоборот: если Иван Ургант его взобьёт, то оставит в миске $\varphi(14) = 6$ граммов теста, в результате чего он проиграет; если же он нальёт один грамм теста в форму, то в миске останется 13 граммов теста, и можно доказать, что в такой ситуации Иван Сафонов не может гарантировать, что именно он приготовит пирог.