

Задача А. Короля — в угол

Имя входного файла: `king1.in`
Имя выходного файла: `king1.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

В левом нижнем углу доски $m \times n$ стоит король. Двое игроков по очереди ходят королем, перемещая его на одну клетку по вертикали вверх, по горизонтали вправо или по диагонали вправо-вверх. Проигрывает тот, кто не может сделать ход. Определите, какой из игроков имеет выигрышную стратегию.

Формат входных данных

На вход программе подается два натуральных числа m и n , не превосходящих 10^9 .

Формат выходных данных

Программа должна вывести номер игрока (1 или 2), который имеет выигрышную стратегию.

Примеры

<code>king1.in</code>	<code>king1.out</code>
3 4	1

Задача В. 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 С. Камни

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

ЛКШатский Дед Мороз решил приехать к нам на день раньше, чтобы увидеть, как вы решаете задачи Новогодней олимпиады 2018. Но, к сожалению, его ждут и в других Летних Зимних Школах. В частности, он обязательно должен побывать в Летней Физической Школе (ЛФШ.Зима), иначе их директор обидится на Деда Мороза и больше его не пригласит. Чтобы все же ему разрешили уехать к нам пораньше, Директор ЛКШ должен обыграть директора ЛФШ.Зима в игру, придуманную, как считают ЛФШата, специально для этого.

На столе перед директорами в кучке лежат N камней. Игроки по очереди делают ходы. На каждом ходе игрок может взять от 1 до K камней из кучки. Первым ходит Директор ЛКШ. Проигрывает тот, кто на своем ходе не сможет взять камень. Их Директор оказался достаточно умным и делает всегда оптимальные ходы. Но мы-то и поумнее видали. Для нескольких игр посчитайте, сколько из них выиграет Директор ЛКШ при правильной игре обоих.

Формат входных данных

В первой строке входного файла записано число $1 \leq t \leq 10^4$ — количество игр. Далее в t строках записаны через пробел по два числа — N ($1 \leq N \leq 10^9$) и K ($1 \leq K \leq 10^9$) — описание очередной игры.

Формат выходных данных

В выходной файл выведите количество игр, которые сможет выиграть наш любимый Директор.

Примеры

<code>stones.in</code>	<code>stones.out</code>
5 3 2 3 3 7 4 9 2 212 77	3

Задача D. Конфетки

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

После разгромной победы Директора ЛКШ руководитель физической школы сдался и отпустил Деда Мороза к нам. Но ради интереса предложил сразиться ЛФШатам и ЛКШатам в еще одной непростой игре.

В каждой игре участвует один из вас и один ЛФШонок, а ходите вы по очереди. В кучку перед вами кладется N вкусных конфеток. На каждом ходе игрок может съесть от 1 до K конфеток (больше нельзя — много сладкого вредно даже в Новый Год), но при этом не больше, чем взял его противник на предыдущем ходе (не будем жадничать, мы же добрые). Второго ограничения нет лишь для первого хода каждой игры. Проигрывает тот, кому не осталось конфеток.

У нас возникли подозрения, что директор ЛФШ специально подобрал такие N и K , чтобы ЛКШата никогда не смогли выиграть. Мы надеемся, что это не так, и очень просим вас проверить это.

Формат входных данных

Во входном файле записаны через пробел два целых числа — N ($1 \leq N \leq 500$) и K ($1 \leq K \leq 100$).

Формат выходных данных

В выходной файл выведите минимальное число конфет, которое должен съесть ЛКШонок первым ходом, чтобы выиграть при оптимальной игре ЛФШонка, либо 0, если даже самый умный из нас не сможет одолеть идеального играющего противника.

Примеры

<code>sweets.in</code>	<code>sweets.out</code>
7 3	1

Задача Е. 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 что-нибудь 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 F. Битва C-шечек

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

Два кандидата, Катя и Валя из параллели C, решили не заморачиваться агитациями и дебатами, а перераспределить голоса в соответствии с игрой, которую они изобрели перед самыми выборами. Правила у неё следующие:

- В начале у Кати есть лента бумаги, разбитая на n клеток. Длина ленты является степенью двух. В каждой из клеток написано целое число.
- Ход состоит в делении ленты на две равные по длине части и выборе той из них, которая выбывает из игры.
- Игрок, которому достаётся лента длины один, обязан передать другому количество голосов, равное числу, написанному на оставшейся клетке.

Катя ходит первой и хочет узнать, какое количество голосов ей придётся отдать, если оба кандидата будут играть оптимальным образом.

Формат входных данных

В первой строке содержится число n ($1 \leq n \leq 65\,536$). Следующая строка содержит n целых чисел a_i — числа, записанные в клетки ($-10^5 \leq a_i \leq 10^5$).

Гарантируется, что N является степенью двойки.

Формат выходных данных

Выведите целое единственное число — количество голосов, которое Катя должна будет отдать Вале.

Примеры

<code>cutting-game.in</code>	<code>cutting-game.out</code>
4 1 2 3 4	2

Замечание

В случае, если одно оставшееся число — отрицательное, кандидат, у которого оно осталось, получает голоса от второго кандидата.

Задача G. 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 Н. Жестокая задача

Имя входного файла: `cruel.in`
Имя выходного файла: `cruel.out`
Ограничение по времени: 10 секунды
Ограничение по памяти: 64 мегабайта

Штирлиц и Мюллер стреляют по очереди. В очереди n человек, стоящих друг за другом. Каждым выстрелом убивается один из стоящих. Кроме того, если у кого-то из стоящих в очереди убиты все его соседи, то этот человек в ужасе убегает. Проигрывает тот, кто не может ходить. Первым стреляет Штирлиц. Требуется определить, кто выиграет при оптимальной игре обеих сторон, и если победителем будет Штирлиц, то найти все возможные первые ходы, ведущие к его победе.

Формат входных данных

Входной файл содержит единственное число n ($2 \leq n \leq 3000$) — количество человек в очереди.

Формат выходных данных

Если выигрывает Мюллер, выходной файл должен состоять из единственного слова `Mueller`. Иначе в первой строке необходимо вывести слово `Schtirlitz`, а в последующих строках — номера людей в очереди, которых мог бы первым ходом убить Штирлиц для достижения своей победы. Номера необходимо выводить в порядке возрастания.

Примеры

<code>cruel.in</code>	<code>cruel.out</code>
3	Schtirlitz 2
4	Mueller
5	Schtirlitz 1 3 5

Задача I. 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 J. 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 ($2 \leq N \leq 100\,000$, $1 \leq R \leq N$). 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	