

## Задача А. 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 В. 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 С. 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 D. Новогодняя игра

Имя входного файла: `substring.in`  
Имя выходного файла: `substring.out`  
Ограничение по времени: 6 секунды  
Ограничение по памяти: 64 мегабайта

Андрей Сергеевич и Андрей Сергеевич оба хотят быть Дедом Морозом в ЛКШ.Зима. Но в ЛКШ.Зима может быть только один Дед Мороз. Чтобы определить, кому же быть Дедом Морозом, Андрей Сергеевич и Андрей Сергеевич решили сыграть в следующую игру.

Маргарита Марковна пишет на доске целое положительное число  $n$ . Андрей Сергеевич и Андрей Сергеевич ходят по очереди.

Каждый ход заключается в том, что Андрей Сергеевич выбирает положительное число  $m$ , которое является собственной подстрокой числа, которое в данный момент написано на доске. После этого число на доске уменьшают на  $m$ .

Напомним, что собственной подстрокой строки  $s$  называется подстрока  $s$ , не совпадающая с  $s$ .

Например, если на доске написано число 2309, то Андрей Сергеевич может выбрать  $m = 2, 3, 9, 23, 30, 230$  или 309. Соответственно, после хода Андрея Сергеевича на доске будет написано 2000, 2079, 2279, 2286, 2300, 2306 или 2307.

Если Андрей Сергеевич не может сделать ход, то он проиграл.

Естественно, и Андрей Сергеевич, и Андрей Сергеевич всегда делают оптимальные ходы.

Дано число  $n$ . Найдите такое число  $m$ , которое Андрею Сергеевичу следует вычесть первым ходом, чтобы выиграть игру. Если возможных  $m$  несколько, выведите наименьшее из них. Если тот Андрей Сергеевич, который ходит первым, не может выиграть игру, выведите  $-1$ .

### Формат входных данных

Во входном файле содержится целое число  $n$  ( $1 \leq n \leq 100\,000$ ).

### Формат выходных данных

В выходной файл выведите наименьшее  $m$ , которое Андрею Сергеевичу следует вычесть первым ходом, или  $-1$ , если Андрей Сергеевич не может сделать выигрышный первый ход.

### Примеры

<code>substring.in</code>	<code>substring.out</code>
5	-1
10	1
17	-1
239	9
566	66
23900	-1

## Задача Е. 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 F. 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 Г. 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	