

Задача А. Лошадью ходи!

Имя входного файла: **knight.in**
 Имя выходного файла: **knight.out**
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 16 мегабайт

В левом верхнем углу прямоугольной доски $N \times M$ находится шахматный конь. Перемещаться по доске он может только двумя способами: на две клетки вправо и на одну вниз, либо на две клетки вниз и на одну вправо. Ваша задача состоит в том, чтобы посчитать количество способов, которыми конь может попасть в правую нижнюю клетку.

Формат входного файла

В единственной строчке находятся два числа N и M ($1 \leq N, M \leq 50$) — размеры доски.

Формат выходного файла

Нужно вывести одно число — количество различных способов переместиться коню из левой верхней в правую нижнюю клетку.

Пример

knight.in	knight.out
4 4	2

Задача В. Конфетки

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

В кучке лежит N конфеток. Двое по очереди делают ходы. На каждом ходе игрок может взять от 1 до K конфеток, но не больше, чем взял его противник на предыдущем ходе (при первом ходе первого этого ограничения нет). Проигрывает тот, кто на своем ходе не сможет взять конфетку. Сможет ли первый игрок выиграть, и как для этого он должен действовать? (Считается, что второй игрок всегда делает оптимальные ходы).

Формат входного файла

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

Формат выходного файла

В выходной файл выведите число конфет, которое должен взять первый игрок, чтобы выиграть, либо 0, если первый игрок выиграть не может.

Пример

candy.in	candy.out
7 3	3

Если в свой ход второй игрок берёт 2 или 3 конфеты, то следующим ходом первый игрок заканчивает игру, если же второй игрок возьмёт одну конфету, то далее каждый будет брать по одной конфете и последняя достанется первому игроку.

Задача С. Две шашки

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

Шашки имеют много разновидностей. Одна из них абсолютна не похожа на привычные нам шашки. Игра происходит на прямоугольной доске $1 \times m$, на которой расположено ровно 2 шашки (одна белая другая черная, при этом белая всегда стоит левее черной). За один ход

игрок может взять шашку своего цвета и передвинуть ее на любую клетку этой же строки, не перепрыгивая через шашку противника. Проигрывает тот, кто не может сделать ход. Найдите кто выиграет при правильной игре.

Формат входного файла

Во входном файле записано число — m . ($2 \leq m \leq 10^9$)
 За ним записаны два числа — расположение шашек в строке: числа l и r ($1 \leq l < r \leq m$) — расположение белой и черной шашки соответственно.

Формат выходного файла

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

Пример

checkers.in	checkers.out
10 1 5	First
10000 9999 10000	Second

Задача D. Ним

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

Игра Ним заключается в следующем. Дано N кучек, в которых лежат A_1, A_2, \dots, A_N камней соответственно. На своём ходу игрок выбирает одну кучку и берёт из неё любое натуральное число камней. Проигрывает тот, кто берёт последний камень. Ваша задача заключается в том, чтобы узнать кто из двух игроков выиграет в оптимальной игре.

Формат входного файла

В первой строке входного файла записано целое число N — количество кучек ($1 \leq N \leq 100$). Во второй строке входного файла записано N целых чисел A_i ($1 \leq A_i \leq 1023$) — количество камней в i -ой кучке.

Формат выходного файла

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

Примеры

nim.in	nim.out
1 10	First
2 2 2	Second