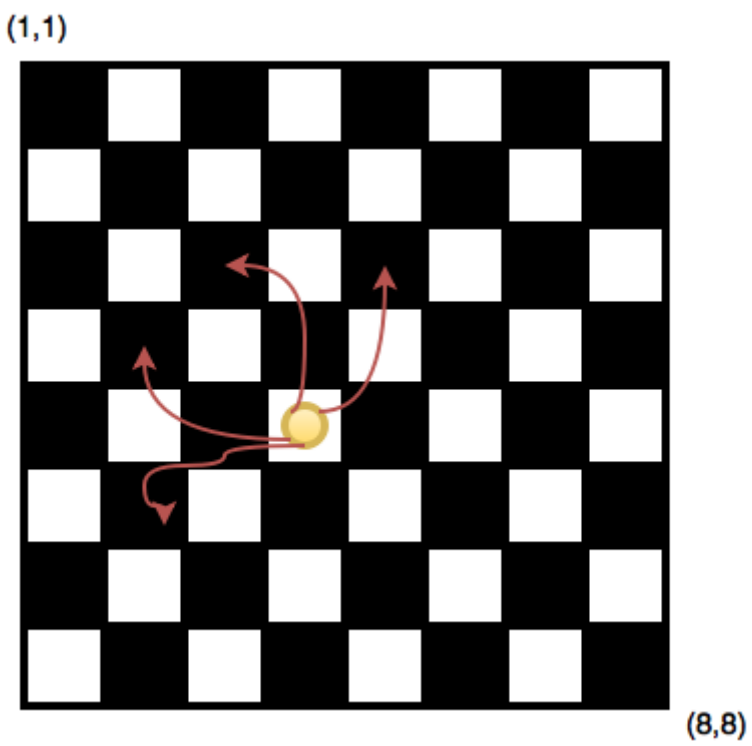
A Chessboard Game ☆

Two players are playing a game on a 15×15 chessboard. The rules of the game are as follows:

- The game starts with a single coin located at some x, y coordinates. The coordinates of the upper left cell are $(1, 1)$, and of the lower right cell are $(15, 15)$.
- In each move, a player must move the coin from cell (x, y) to one of the following locations:
  1. $(x-2, y+1)$
  2. $(x-2, y-1)$
  3. $(x+1, y-2)$
  4. $(x-1, y-2)$

  Note: The coin must remain inside the confines of the board.
- Beginning with player 1, the players alternate turns. The first player who is unable to make a move loses the game.

The figure below shows all four possible moves using an 8×8 board for illustration:



Given the initial coordinates of the players' coins, assuming optimal play, determine which player will win the game.

Function Description

Complete the chessboardGame function in the editor below. It should return a string, either `First` or `Second`.

chessboardGame has the following parameter(s):

- x: an integer that represents the starting column position
- y: an integer that represents the starting row position

Input Format

The first line contains an integer t, the number of test cases.

Each of the next t lines contains 2 space-separated integers x and y.

Constraints

- $1 \leq t \leq 225$
- $1 \leq x[i], y[i] \leq 15$

Author Shafaet

Difficulty Easy

Max Score 15

Submitted By 5592

NEED HELP?

View discussions

View editorial

View top submissions

RATE THIS CHALLENGE ☆ ☆ ☆ ☆ ☆

MORE DETAILS

Download problem statement

Download sample test cases

Suggest Edits

Output Format

On a new line for each test case, print **`First`** if the first player is the winner. Otherwise, print **`Second`**.

Sample Input

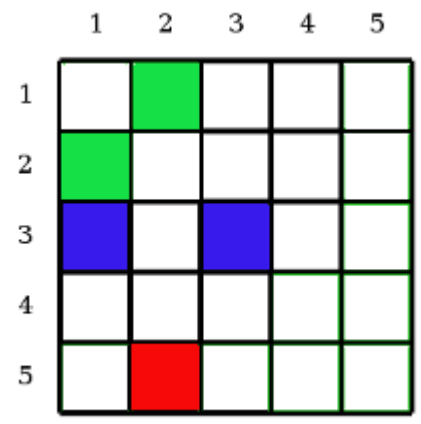
```
3
5 2
5 3
8 8
```

Sample Output

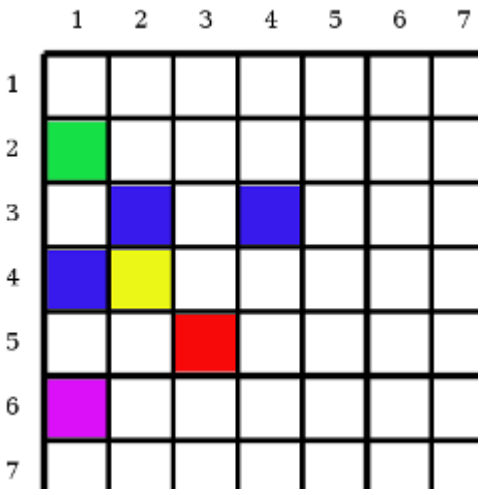
```
Second
First
First
```

Explanation

In the first case, player1 starts at the red square and can move to any of the blue squares. Regardless of which one is chosen, the player 2 can move to one of the green squares to win the game.



In the second case, player 1 starts at the red square and can move to any of the blue squares or the purple one. Moving to the purple one limits player 2 to the yellow square. From the yellow square, player 1 moves to the green square and wins.



C++14

```cpp
#include <iostream>
#include <algorithm>
#include <string>
#include <unordered_set>
#include <unordered_map>
#include <vector>
```

```
#include <map>

std::vector<std::pair<int,int>> moves(int x, int y ){
  std::vector<std::pair<int,int>> v = { {x-2,y+1},{x-2,y-1},{x+1,y-2},{x-1,y-2} };
  auto it_end = std::remove_if( v.begin(), v.end(),
                [](auto a){ return a.first < 1 || a.first > 15 ||
                        a.second < 1 || a.second > 15; });
  v.resize( std::distance(v.begin(),it_end) );
  return v;
}

bool simulate(int x, int y, bool player1, std::map<std::pair<int,int>,bool> & cache ){

  auto m = moves(x,y);
  if (m.empty()){
    return false; //no moves left, current player loses
  }

  auto mem = cache.find({x,y});
  if(mem!=cache.end()){
    return mem->second;
  }

  bool current_player_win = false;
  for(auto i: m){
    bool result = !simulate(i.first,i.second, !player1, cache );
    current_player_win |= result;
  }

  cache[{x,y}] = current_player_win;

  return current_player_win;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code