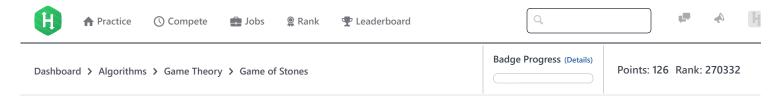
16/11/2017 HackerRank



Game of Stones **■**





Two players (numbered 1 and 2) are playing a game with n stones. Player 1 always plays first, and the two players move in alternating turns. The game's rules are as follows:

- In a single move, a player can remove either 2, 3, or 5 stones from the game board.
- If a player is unable to make a move, that player loses the game.

Given the number of stones, find and print the name of the winner (i.e., First or Second) on a new line. Each player plays optimally, meaning they will not make a move that causes them to lose the game if some better, winning move exists.

Input Format

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

Each of the T subsequent lines contains a single integer, n, denoting the number of stones in a test case.

Constraints

- $1 \le T \le 100$
- $1 \le n \le 100$

Output Format

On a new line for each test case, print First if the first player is the winner; otherwise, print Second.

Sample Input

8 1

2

3 4

5

6

10

Sample Output

Second

First

First

First

First

First

Second First

Explanation

In the sample, we have T=8 testcases.

16/11/2017 HackerRank

We'll refer to our two players as P_1 and P_2 .

If n=1, P_1 can't make any moves and loses the game (i.e., the P_2 wins and we print ${\tt Second}$ on a new line).

If n=2, P_1 removes 2 stones in their first move and wins the game, so we print **First** on a new line.

If n = 3, P_1 removes 2 stones in their first move, leaving 1 stone on the board. Because P_2 is left with no available moves, P_1 wins and we print **First** on a new line.

If n = 4, P_1 removes 3 stones in their first move, leaving 1 stone on the board. Because P_2 has no available moves, P_1 wins and we print **First** on a new line.

If n = 5, P_1 removes all 5 stones from the game board. Because P_2 is left with no available moves, P_1 wins and we print **First** on a new line.

If n = 6, P_1 removes 5 stones in their first move, leaving 1 stone on the board. Because P_2 has no available moves, P_1 wins and we print **First** on a new line.

If n=7, P_1 can make any of the following three moves:

- 1. Remove 2 stones, leaving 5 stones on the board. P_2 then removes 5 stones. Because P_1 has no available moves, P_2 wins.
- 2. Remove 3 stones, leaving 4 stones on the board. P_2 then removes 3 stones, leaving 1 stone left on the board. Because P_1 has no available moves, P_2 wins.
- 3. Remove 5 stones, leaving 2 stones on the board. P_2 then removes the 2 remaining stones. Because P_1 has no available moves, P_2 wins.

Because all possible moves result in P_2 winning, we print **Second** on a new line.

If n = 10, P_1 can remove either 2 or 3 stones to win the game, so we print **First** on a new line. Recall that each player moves optimally, so P_1 will not remove 5 stones because doing so would cause P_1 to lose the game.

f in Submissions:<u>9971</u> Max Score:15 Difficulty: Easy Rate This Challenge: ☆ ☆ ☆ ☆ ☆

```
Current Buffer (saved locally, editable) &
                                                                                            Java 7
                                                                                                                              Ö
 1 ▼ import java.io.*;
    import java.util.*;
    import java.text.*;
 4
    import java.math.*;
 5
    import java.util.regex.*;
 6
 7 ▼ public class Solution {
 8
 9 ₩
        public static void main(String[] args) {
10 ▼
            /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11
12
    }
                                                                                                                     Line: 1 Col: 1
                      Test against custom input
                                                                                                         Run Code
                                                                                                                      Submit Code
1 Upload Code as File
```

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature