**Task 1. Heaps**

Nini and Mimi are playing a game with heaps of stones and pebbles. Each heap has big stones and small pebbles. Nini and Mimi take turns performing moves and once a player has no more moves to do, they lose. Each move consists of choosing a non-empty heap and removing some stones and/or pebbles from it. Formally, one can remove stones and pebbles, where , and . However, every removed stone must be replaced with at least pebbles; it can be replaced with any natural number of pebbles not less than . Thus, in any move where , first pebbles are removed and then the player must add back pebbles, which are taken from an infinite supply of pebbles. Nini goes first. Before making her move she wonders whether she can win the game if she plays optimally. Write a program heaps.cpp, which answers her question.

**Input**

From the first line of the standard input, your program should read and . Then independent tests with that will follow. For each test, the first line contains . The next lines each have a desription of a heap: and .

**Output**

On lines, your program should output the answers to each of the tests in the order they were given. It should print Win, if Nini can win, and Loss, otherwise.

**Constraints**

**Subtasks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Subtask** | **Points** |  |  | **Additional constraints** |
| 1 | 8 |  |  |  |
| 2 | 11 |  |  | If , then . |
| 3 | 12 |  |  |  |
| 4 | 18 |  |  |  |
| 5 | 18 |  |  |  |
| 6 | 10 |  |  |  |
| 7 | 11 |  |  |  |
| 8 | 12 |  |  |  |

Your solution will receive the points for a subtask only if it passes all tests in it.

**Sample test**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 2  2  1 5  3 2  3  0 3  2 1  3 2 | Win  Loss |