Outlab

Due in two weeks Feb 6th

The Game of NIM

This lab is really not that hard, but please don't wait until the last minute.

This is a well known game with little tricks.
The game is played with two players that alternately take marbles from a pile. In each move, a player chooses how many marbles to take. The player must take at least one but at most half of the marbles. Then the other players takes a turn. The player who takes the last marble loses.

Write a program in which the computer plays against a human opponent.

Use Java, if you want to write it up in Python first you can, but you only turn in Java source for the graders. The program should be in a Class called Nim, and should have a constructor, and all the methods needed to play the game. You should not use static methods, you should create an instance of the Nim class, with instance fields, and call the methods using the dot operator.

- Generate a random integer between 10 and 100 to represent the original stack of marbles.
- Generate another random number between 0 and 1 to determine who goes first, computer or player.
- Generate another random number between 0 and 1 to determine if the computer plays smart or stupid.
  - In stupid mode the computer simply takes a random number of marbles between 1 and n/2 (n being the total marbles).
  - In smart mode the computer takes off enough marbles to make the size of the pile a power of two minus 1 -- that is 3, 7, 15, 31 or 63. (example 2^2 - 1 = 3 and 2^3 - 1 = 7 and 2^4 - 1 = 15). This will always work unless the pile is 1 less than power of 2, so in that case the computer will take a random legal move.

See if you can beat the computer if it goes first, if you programmed it correctly you can't.

Grading

This will be graded on the program working, by allowing a human player to play against the computer following the rules and a winner is decided.

Proper programming skills, good clear variable definitions, good class structure, and the proper use of methods will all be graded. As well as properly commented code.