Programming Assignment 3 – Basic Probability, Computing and Statistics 2015

Fall 2015, Master of Logic, University of Amsterdam

Submission deadline: Monday, September 21st, 9 a.m.

Note: if the assignment is unclear to you or if you get stuck, do not hesitate to contact Philip.

1 Playing games

This week we will implement a little game to entertain ourselves with. The game is called Rock, Paper, Scissors. The rules are straightforward. Each of two players chooses one of rock, paper and scissors. Rock beats scissors, scissors beat paper and paper beats rock. The player who chooses the winning item gets one point. If both players choose the same item, no points are awarded. The game is played for n rounds and the player with the most points is declared the winner. If both players have the same amount of points, a tie breaking round is played. If both players pick the same item in the tie breaker, another round is played until finally one player wins the game.

You will implement *Rock, Paper, Scissors*. There should be three computer player: one who picks his item randomly in each round, one who plays a pre-determined strategy of your choice and one who cheats. The cheater waits till has opponent has chosen and then picks the item that beats the item of the opponent. The second computer player, who uses a strategy of your choice, must not consider the current pick of his opponent but may take into account previous picks.

The user should be able to determine the number of round that shall be played. He can either play against one of the computer players himself or he can have two computer players play against each other.

2 Grading

In this assignment everything should be either a function or a class. There should be no code outside a function or a class (except for import statements). Your program has to be executable from the command line. The user first gets to choose whether he wants to play himself or whether he wants to simulate a game between computer players. He then gets too choose the computer players (either one or two of them, depending on the previous choice). Finally, he selects the number of rounds and the game starts.

2 points All players, computer or human, are classes that inherit from the same parent class.

1 point The user is given options as described in the previous paragraph.

3 points Three computer players are available as described in the previous section (1 point per computer player). They work correctly (i.e. they play the strategies described in Section 1). Only give 0.5 points if the description of the strategy that was chosen by the programmer for the second computer player is not clear. This description should be given when the user is asked to choose the computer player(s).

1 point Human against computer gameplay works correctly.

1 point Computer against computer gameplay works correctly.

1 point If after n rounds there is a winner, the program determines that winner correctly.

1 point If after n rounds there is a tie, additional rounds are played till the tie is broken. The program then correctly determines the winner (give 0.5 points if tie break rounds are played but the winner is not declared immediately afterwards).

-1 point If there is any code outside of classes or functions (except imports and the main method's if-statement).

-2 points If the program crashes on any of the user inputs.

-1 point If the messages described below are not printed.

To spare the user the pain of having to type his choice each time, the options should be numeric and the user should be reminded of what the options are in each round. Thus, at the beginning of each round, the program should print something like

Please choose your item (1=rock, 2=paper, 3=scissors)

Also after both the user and the computer player (or both computer players) have made their choice, there should be a print-out about what has happened in this round, followed by a tally of the points (there is no need to centre this).

Player1 chose item1.
Player2 chose item2.
Item1/2 beats item1/2.
Player1 : X points   Player2: Y points

Here item1/2 are the items chosen by the players. You can use Player1/2 if you like, as long as it is clear who is who (e.g. a human player could always be Player1).