Project: Rock-Paper-Scissors

By: Jose Luis Centeno.

Contenido

[Requirements 3](#_Toc475921175)

[Solution 4](#_Toc475921176)

[UML Diagram 4](#_Toc475921177)

[Implementation 4](#_Toc475921178)

[Project 5](#_Toc475921179)

## Requirements

The Task

\*\*\*\*\*\*\*\*\*\*\*\*  
Write a program that plays 100 iterations of Rock, Paper, Scissors ([https://en.wikipedia.org/wiki/Rock-paper-scissors](http://de.wikipedia.org/wiki/Schere,_Stein,_Papier)).   
One player should always play randomly, the other should always choose rock. It should show at the end how many games each player has one and how many were a draw.

The code should have high test coverage.  
Keep it simple - remember: we're not looking for the perfect "correct" solution, but a basis for discussion and development.

Technical requirements  
Language: Java, tests in a JVM language of your choice  
Approach: ideally "test-driven"

Please use local git (initially a “git init” in the project directory), so that we see a little of your working methods.

## Solution

### UML Diagram



### Implementation

Test on Figure class:

* constructor () generate a random figure
* constructor(index) generate the figure indicate by the parameter.
* winTo() return true if wins to figure indicate by the parameter.
* draw() return true if is the same figure that the indicate by the parameter.

Test on Move class:

* go() execute a move and sets Figure 2 always Rock
* go() execute a move and sets Figure 1 a random figure
* go() execute a move and sets winner:
  + 0 – Draw
  + 1 – Player 1 wins
  + 2 – Player 2 wins

Test on Match class:

* Play() execute a ‘x’ iterations loop, the lstMove length must be the same of iterations property.
* Play() execute a ‘x’ iterations loop, winsPlayer1 must by highest than 0
* Play() execute a ‘x’ iterations loop, winsPlayer2 must by highest than 0
* Play() execute a ‘x’ iterations loop, draw must by highest than 0

### Project

The Project is generated with maven 3.3.9. It has 3 modules:

* libModule contains the mains classes and jUnit Cases. Library dependencies resolved with maven: jUnit 4.12
* consoleTest: contains a console application to execute the match. Library dependencies resolved with maven:
  + libModule
  + jUnit 4.12
* webTest: contains a web application to execute the match. Include a Tomcat 7 pluging to deploy localy the application. Library dependencies resolved with maven:
  + libModule
  + javax.faces 2.1.17
  + primefaces 5.3