Rock-Paper-Scissors strategy competition program

The `Rock-Paper-Scissors strategy competition program` is a program which simulated two players playing the traditional hand game rock paper scissors. The two players use two different strategies:

- Player A always chooses paper as a weapon.

- Player B chooses one of the three weapons randomly.

The program simulated 100 rounds of this game and prints some statistics about the the number of victories and ties in the game.

Design

Maximum effort was made to make a simple design.

The user interface is minimal, the program is a command line application with only one possible optional argument to display verbose output in order to be able to inspect each round of the game.

There are 5 classes and one interface. The single interface is called `Player` and defines a single method which must be implemented, called `choose` as all types of players must be able to choose a weapon. There are two implementations:

- `NoChoicePlayer` which implements the strategy of player A.
- `RandomChoicePlayerTest` which implements the strategy of player B.

This is an object-oriented design aimed at reducing coupling and increasing conceptual cohesion to make the program easier to read and understand.

The rules of the game are enforced by another class as well as the outcome of a game, this class is the `Board` of the game. If we wanted to test our players in a different game, we could create an alternative class.

The three weapons that can be used by the players are defined in an enumeration.

There are three test cases which test the classes mentioned above to ensure correctness and improve the ease of future change.

The `App` class is the main class of the program where all the ingredients mentioned are put into use and the output is displayed.

How to build the app

To build the app you will need to download and install (if not already available on your system):

- [Java SE 8 JDK](#) or later
- [Maven](#)

Open a terminal (in UNIX-like systems) or command prompt (in Windows) and do the following steps:

Clone the project locally:

```
git clone git@github.com:velissarious/Rock-Paper-Scissors-strategy-competition-
program.git
```

Move to the directory:

```
cd Rock-Paper-Scissors-strategy-competition-program/
```

Use Maven with the `package` target to generate the executable jar file which will also contain all
the dependencies. Like this:

```
mvn package
```

This will produce a jar file. Specifically:

```
rock_paper_scissors-1.0.0-SNAPSHOT.jar
```

This is the executable of the program. Run with the instructions of the next section.

How to run and sample output

You can run the program from the command line like this:

```
java -jar rock_paper_scissors-1.0.0-SNAPSHOT.jar
```

Here are three samples of output produced by three different runs of this program.

First run:

```
Rock-Paper-Scissors
Player A wins 34 of 100 games
Player B wins 34 of 100 games
Tie: 32 of 100 games
```

In this run, player A and B win an equal amount of times.

Second run:

```
Rock-Paper-Scissors
Player A wins 36 of 100 games
Player B wins 31 of 100 games
Tie: 33 of 100 games
```

In this run, player A wins most of the games.

Third run:

```
Rock-Paper-Scissors
Player A wins 31 of 100 games
Player B wins 38 of 100 games
Tie: 31 of 100 games
```

In this run, player B wins most of the games.

As you can see due to the random nature of player B the winner is different. However, the results
are mostly evenly distributed.

Verbose output option
```

You can produce a verbose version of the output by running the program with the `-v` option like this:

```
java -jar rock_paper_scissors-1.0.0-SNAPSHOT.jar -v
```

A sample output of this might look like this:

```
Rock-Paper-Scissors
Player A PAPER Player B ROCK
Round 0 victor 1
Player A PAPER Player B PAPER
Round 1 victor 0
Player A PAPER Player B PAPER
Round 2 victor 0
Player A PAPER Player B ROCK
Round 3 victor 1
Player A PAPER Player B ROCK
Round 4 victor 1
Player A PAPER Player B SCISSORS
Round 5 victor 2
Player A PAPER Player B PAPER
Round 6 victor 0
Player A PAPER Player B ROCK
Round 7 victor 1
Player A PAPER Player B SCISSORS
Round 8 victor 2
Player A PAPER Player B SCISSORS
Round 9 victor 2
Player A PAPER Player B PAPER
Round 10 victor 0
Player A PAPER Player B PAPER
Round 11 victor 0
Player A PAPER Player B ROCK
Round 12 victor 1
Player A PAPER Player B PAPER
Round 13 victor 0
Player A PAPER Player B SCISSORS
Round 14 victor 2
Player A PAPER Player B PAPER
Round 15 victor 0
Player A PAPER Player B PAPER
Round 16 victor 0
Player A PAPER Player B PAPER
Round 17 victor 0
Player A PAPER Player B ROCK
Round 18 victor 1
Player A PAPER Player B PAPER
Round 19 victor 0
Player A PAPER Player B ROCK
Round 20 victor 1
Player A PAPER Player B SCISSORS
Round 21 victor 2
Player A PAPER Player B ROCK
Round 22 victor 1
Player A PAPER Player B SCISSORS
Round 23 victor 2
Player A PAPER Player B SCISSORS
Round 24 victor 2
```

```
Player A PAPER Player B SCISSORS
Round 25 victor 2
Player A PAPER Player B SCISSORS
Round 26 victor 2
Player A PAPER Player B SCISSORS
Round 27 victor 2
Player A PAPER Player B PAPER
Round 28 victor 0
Player A PAPER Player B ROCK
Round 29 victor 1
Player A PAPER Player B SCISSORS
Round 30 victor 2
Player A PAPER Player B SCISSORS
Round 31 victor 2
Player A PAPER Player B SCISSORS
Round 32 victor 2
Player A PAPER Player B ROCK
Round 33 victor 1
Player A PAPER Player B PAPER
Round 34 victor 0
Player A PAPER Player B ROCK
Round 35 victor 1
Player A PAPER Player B SCISSORS
Round 36 victor 2
Player A PAPER Player B PAPER
Round 37 victor 0
Player A PAPER Player B PAPER
Round 38 victor 0
Player A PAPER Player B SCISSORS
Round 39 victor 2
Player A PAPER Player B ROCK
Round 40 victor 1
Player A PAPER Player B ROCK
Round 41 victor 1
Player A PAPER Player B ROCK
Round 42 victor 1
Player A PAPER Player B SCISSORS
Round 43 victor 2
Player A PAPER Player B SCISSORS
Round 44 victor 2
Player A PAPER Player B ROCK
Round 45 victor 1
Player A PAPER Player B PAPER
Round 46 victor 0
Player A PAPER Player B ROCK
Round 47 victor 1
Player A PAPER Player B ROCK
Round 48 victor 1
Player A PAPER Player B ROCK
Round 49 victor 1
Player A PAPER Player B ROCK
Round 50 victor 1
Player A PAPER Player B ROCK
Round 51 victor 1
Player A PAPER Player B ROCK
Round 52 victor 1
Player A PAPER Player B ROCK
Round 53 victor 1
```

```
Player A PAPER Player B PAPER
Round 54 victor 0
Player A PAPER Player B PAPER
Round 55 victor 0
Player A PAPER Player B ROCK
Round 56 victor 1
Player A PAPER Player B ROCK
Round 57 victor 1
Player A PAPER Player B PAPER
Round 58 victor 0
Player A PAPER Player B ROCK
Round 59 victor 1
Player A PAPER Player B ROCK
Round 60 victor 1
Player A PAPER Player B SCISSORS
Round 61 victor 2
Player A PAPER Player B PAPER
Round 62 victor 0
Player A PAPER Player B ROCK
Round 63 victor 1
Player A PAPER Player B SCISSORS
Round 64 victor 2
Player A PAPER Player B SCISSORS
Round 65 victor 2
Player A PAPER Player B PAPER
Round 66 victor 0
Player A PAPER Player B ROCK
Round 67 victor 1
Player A PAPER Player B SCISSORS
Round 68 victor 2
Player A PAPER Player B SCISSORS
Round 69 victor 2
Player A PAPER Player B SCISSORS
Round 70 victor 2
Player A PAPER Player B ROCK
Round 71 victor 1
Player A PAPER Player B PAPER
Round 72 victor 0
Player A PAPER Player B ROCK
Round 73 victor 1
Player A PAPER Player B PAPER
Round 74 victor 0
Player A PAPER Player B SCISSORS
Round 75 victor 2
Player A PAPER Player B SCISSORS
Round 76 victor 2
Player A PAPER Player B ROCK
Round 77 victor 1
Player A PAPER Player B PAPER
Round 78 victor 0
Player A PAPER Player B ROCK
Round 79 victor 1
Player A PAPER Player B PAPER
Round 80 victor 0
Player A PAPER Player B SCISSORS
Round 81 victor 2
Player A PAPER Player B ROCK
Round 82 victor 1
```

```
Player A PAPER Player B PAPER
Round 83 victor 0
Player A PAPER Player B ROCK
Round 84 victor 1
Player A PAPER Player B SCISSORS
Round 85 victor 2
Player A PAPER Player B SCISSORS
Round 86 victor 2
Player A PAPER Player B SCISSORS
Round 87 victor 2
Player A PAPER Player B ROCK
Round 88 victor 1
Player A PAPER Player B PAPER
Round 89 victor 0
Player A PAPER Player B PAPER
Round 90 victor 0
Player A PAPER Player B SCISSORS
Round 91 victor 2
Player A PAPER Player B SCISSORS
Round 92 victor 2
Player A PAPER Player B PAPER
Round 93 victor 0
Player A PAPER Player B SCISSORS
Round 94 victor 2
Player A PAPER Player B SCISSORS
Round 95 victor 2
Player A PAPER Player B ROCK
Round 96 victor 1
Player A PAPER Player B PAPER
Round 97 victor 0
Player A PAPER Player B SCISSORS
Round 98 victor 2
Player A PAPER Player B SCISSORS
Round 99 victor 2
Player A wins 36 of 100 games
Player B wins 35 of 100 games
Tie: 29 of 100 games
```

In this output we can see the selections of both players for each one of the 100 rounds and the winner of each round.

- If the victor is 0 then the round ended in a tie.
- If the victor is 1 then player A won.
- If the victor is 2 then player B won.

Testing

Unit tests are also available, written in [JUnit 4](#).