**Project report**

# Software Development – Problem Based Learning Project

Ronan Hoyne, Sigmund Vestergaard and Martin O’Sullivan.

# Analysis

1. The main game is implemented using a do-while loop.
2. The history of fingers shown by each user has been implemented using a 2-D array
3. The logic of scoring is implemented in an instantiable class, which has the following methods:
   1. public void compute() – which contains the scoring logic
   2. public int getCompFingers() – which gets the number of fingers shown by the computer
   3. public int getComputerScore() – which gets the compute’s score determined by the compute() method
   4. public int getPlayerScore() – ditto
   5. public int getSum() – get the sum of the fingers shown; used for printing purposes
   6. public void setComputerScore(int computerScore) – used to initialize the computer’s score
   7. public void setFingers(int fingers) – used to set the number of fingers typed in by the human player
   8. public void setOddOrEven(int oddOrEven) – used to set the choice of odd/even selected by the human player
   9. public void setPlayerScore(int playerScore) – used to initialize the human player’s score
4. A number of print statements are used to tell the human player what is going on in the game and to prompt for input:
   1. The statement reading in the user’s choice of odd or even has been wrapped in a while loop checking that the user chose either 1 or 2
   2. The same has been done for the statement reading in the user’s choice of fingers to show; here the while loop is checking that the number typed is between 1 and 10 (inclusive)
5. At the end of the loop and if-else-if block is used to determine which of the two players has won the game, i.e. which player has a score of 6 or more and feed this information back to the human player
6. After each game has been finished the user is asked whether they want to play another game; if they answer no the loop is exited and the game history shown on the screen

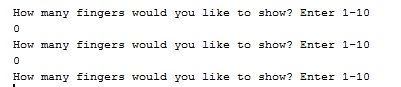
# Share of workload

|  |  |
| --- | --- |
| Item | Done by |
| 2D arrays | Sigmund |
| App class main method | Sigmund, Ronan |
| If statements for game end | Martin |
| Test cases | Martin |
| Main game loop | Ronan, Sigmund |
| Instantiable class methods | Ronan, Sigmund |
| Games testing | Ronan, Sigmund, Martin |
| Exception handling | Sigmund |

# Test cases

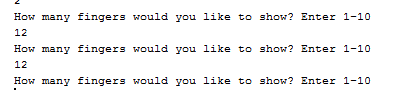
Test if a player puts in a character, where a number is expected.

A player enters a number less than 1 .



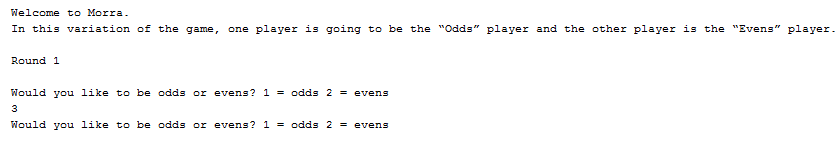
If a players enters a number less than 0 , he will be prompted to enter again.

Player enters a number greater than 10.



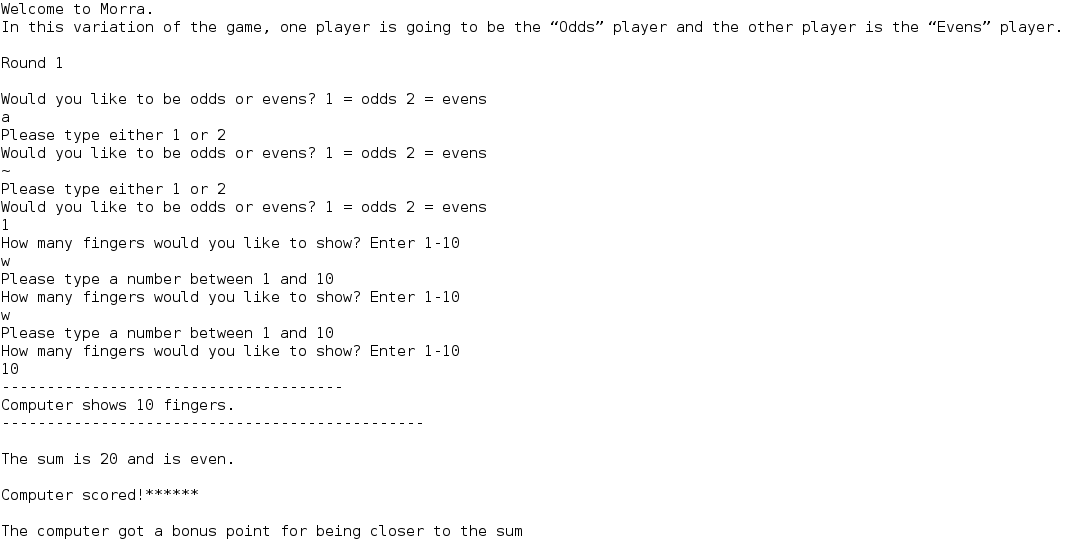
If a players enters a number greater than 10 , he will be prompted to enter again.

If a player enters a number other than 1 or 2 for the odd or even.



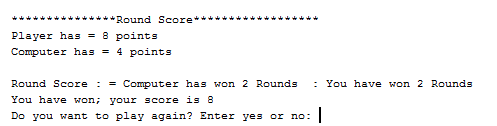
If the players enters a number greater than 2 he will be prompted again to enter either 1 or 2.

If a user enters a letter where a number is expected.



If a letter is entered the user is asked to enter a number

Test that the game will end once one user reaches 6 points or more.



Test that game history is shown for all games at the end of the final game