COMP4005 – Resit Assignment

For this assignment you will design and implement a set of classes that work together to simulate a game with a pair of dice. There are several games with dice but the one that you are going to develop only requires two dice and some specific rules that are used to get some points. In this game two players will alternatively throw a pair of dice until one of the players reaches 100 points and therefore wins the game.

You have to implement a program in Java where the computer simulates the game. The computer will simulate throwing the dice for each player by using random numbers between 1 and 6. The objective of this game is to be the first to score 100 points.

Here we have the rules of the game:

1. If the number one does not appear on either of the dice, the player adds the two dice values to his/her score and the turn passes to the next player.

2. If the number one appears on one dice, nothing is added and the turn passes to the next player.

3. If the number one appears on both dice then the player loses all the accumulated points and the turn passes to the next player.

You should use classes to implement this program. You may need a Game class, Dice class and Player class but you may need other classes as well. It is up to you to design the different classes and implement them.

Your program should meet the following functional requirements:

1. The program must simulate a game with 2 players and 2 dice
2. Each player must keep track of their score
3. The program will roll two dice to a player and the rules of the game must be followed to calculate the points for that player.
4. The turn will then pass to the other player
5. The program finishes when a player has reached a score of 100 once the second player has rolled their dice. If both players obtain a score greater than 100 the one with the highest score wins.
6. Once the game is complete the winner must be displayed.

Your program also must meet the following non-functional requirements:

1. At least three classes must be implemented: Player, Dice and Main class
2. You have to show the relationship between the classes by producing an UML diagram
3. You should create a well-structured Java program that uses the concepts that we have learned up to this point and makes sensible use of comments.
4. You should carry out testing of your program. You should work out a sensible way to test that the program works by tracing the simulation of the game.

Submission

The work you submit should comprise two parts. A report (MS Word or pdf file) and the source codet . The report should contain the design, the results of running your program with an explanation of how it has been tested and the Java source code copied and pasted in.

This is an individual piece of work and you have to work on your own and submit your own original attempt to the assignment. Any code that has been copied from any source (e.g. Stack Overflow, online tutorial, textbooks, other student, etc.) must be properly referenced to avoid any suspicion of plagiarism.

Marking scheme 60 marks

1. The design using UML diagrams **[20 marks]**. Use the correct diagrammatic conventions to show relationships between classes. Show the visibility of attributes and methods and the parameters of methods. You have to describe what every method does specially the main program.

    i) UML diagram

    [10 marks]

    ii) Description of each method.

    [10 marks]

3) Implementation **[35 marks]**. Please cut and paste your code into the word-processed document that you submit. Your code should be in a fixed-pitch font such as Courier and indented sensibly. You should provide comments in your program. You will gain marks for a correct implementation that works. Implementation marks are divided as follows:

    i) Implementation of at least three classes: Dice, Player and the main class

    [10 marks]

    ii) Correct use of parameters and attributes, implementation of methods in classes

    [10 marks]

    iii) Implementation of the game with correct use of variables, if statements, and loops

    [10 marks]

    iv) Program works [5 marks]

4) Testing the program and results **[5 marks]**. Since your program is a simulation you should use print statements to see the value on the dice to check that game is played following the rules. You should also provide a description of problems encountered and how you have solved them or failed to solve them.