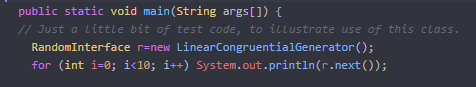
CM2307

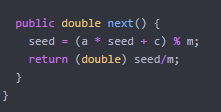
Object Oriented modelling and programming

C1816377 Harry Batchelor

1. * 1. To fix the problems discussed in question 1 I altered some lines in LinearCongruentialGgenerator.java

I changed the code from IncompatibleRandomInterface to RandomInterface



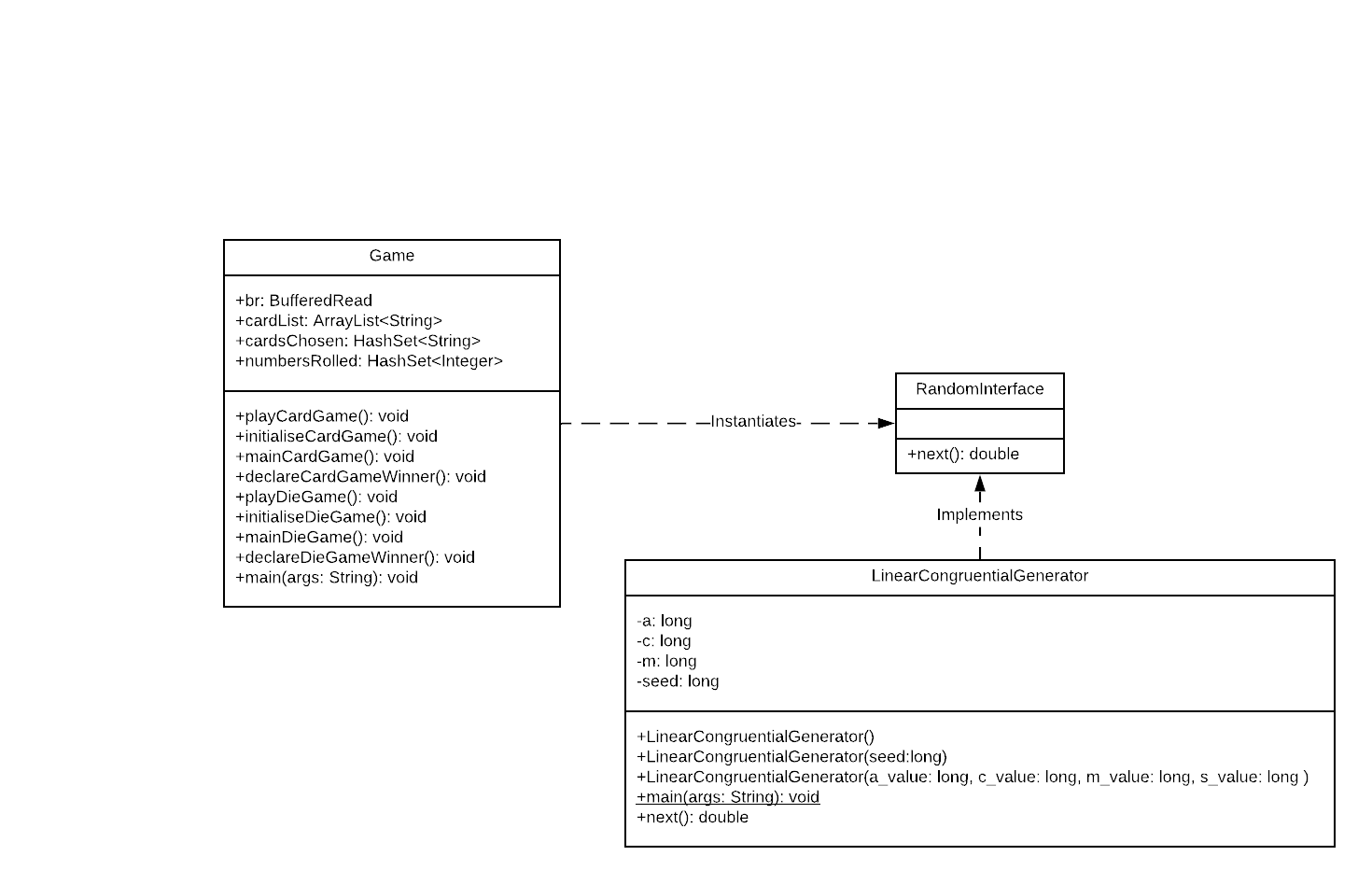
I changed IncompatibleRandomInterface to Randominterface and r.getNextNumber() to r.next()

Changed the name of the operation from getNextNumber() to next()

I also fixed the Game.java file by removing the “;//” which was commenting out the LinearCongruentialGenerator() so it wasn’t being implemented correctly.



* + 1. UML diagram for Question 1 after modifying the code for part i



* + 1. The program contains an interface and 2 classes. The interface, “RandomInterface” is implemented by the class “LinearCongruentialGenerator” which picks a random number normalised from 0 to 1. The value that is given out from the “LinearCongruentialGenerator” is instantiated by the class “Game”; this is then used on the methods that are used to play the different games. The program current contains low cohesion because there are a lot of methods which are all doing lots of different things which all group into just 2 classes. Every class should be responsible for doing one task. So especially for the “Game” class which has 9 methods that should have been split into different classes in order to maximise cohesion. To further optimise the program this class should have been split into 3 more different classes, “Main”, “DiceGame”, “Cardgame” focusing the different methods in relation to the role of the class.

Then again, the classes could be considered loosely coupled because the “Game” class interacts with the “LinearCongruentialGenerator” class, but it also depends on the “RandomInterface” interface. This does make it relatively good for the programme since it maintenance of the code is easier, but there is also from for improvement. They are also not strongly connected as they don’t rely on the internal representation of the other.