AC12001 - Assignment 2 - Sets

Report

**Problem**

For this assignment I had been tasked with writing a program that uses Java classes to create a lottery game. This game should be multiplayer and allow one or multiple users to enter 6 numbers and then the computer will generate random numbers which will be compared to the inputted numbers. The program will calculate the amount that each player one based on the numbers matched. The user should be able to choose how many players there are and also pick the range of numbers that will be chosen.

**Summary of Requirements**

I managed to complete all of these requirements.

1. The user should be able to enter the amount of people playing
2. The user should b able to pick a range of lottery numbers that will be chosen
3. The user must enter 6 lottery numbers between the range
4. The computer should store the lottery numbers in a set using a java class
5. The computer should choose random numbers in the range
6. The program should use an intersection method and use retainAll()
7. The winnings should be calculated depending on how many numbers matched the lottery numbers
8. The amount each player won should be displayed on screen

**Pseudocode**

addToSet()

|  |  |
| --- | --- |
| 1 | FOR EVERY player DO |
| 2 | FOR EVERY number IN lottery DO |
| 3 | GET number FROM user |
| 4 | player.add(number) |
| 5 | END FOR |
| 6 | END FOR |

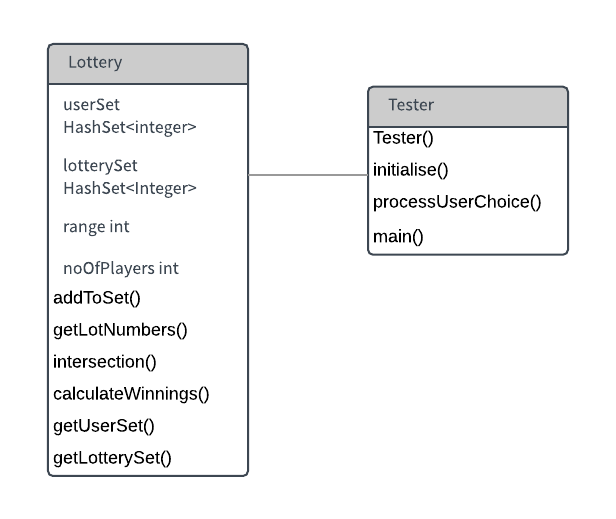
generateLottery(range)

|  |  |
| --- | --- |
| 1 | FOR EVERY number IN lottery |
| 2 | SET number TO randomInt(range) |
| 3 | lottery.add(number) |
| 4 | END FOR |

calculateWinnings()

|  |  |
| --- | --- |
| 1 | FOR EVERY player DO |
| 2 | IF noOfMatches == 3 DO |
| 3 | PRINT “£25 won” |
| 4 | ELSE IF noOfMatches == 4 DO |
| 5 | PRINT “£100 won” |
| 6 | ELSE IF noOfMatches == 5 DO |
| 7 | PRINT “£1000 won” |
| 8 | ELSE IF noOfMatches == 6 DO |
| 9 | PRINT “£1,000,00 won” |
| 10 | ELSE DO |
| 11 | PRINT “£0 won” |
| 12 | END IF |
| 13 | END FOR |

**Class Diagram**



**Test Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected result** | **Worked?** |
| 3 matched numbers | userNums = [1,2,3,4,5,6]  lotteryNums= [4,5,6,7,8,9] | Intersection method retains 3 numbers and displays that user has won £25 | Y |
| No matched numbers | userNums = [1,2,3,4,5,6]  lotteryNums= [7,8,9,10,11,12] | Intersection method retains no numbers and displays that user has not won anything | Y |
| 6 matched numbers | userNums = [1,2,3,4,5,6]  lotteryNums= [1,2,3,4,5,6]] | Intersection method retains 6 numbers and displays that user has won £1,000,000 | Y |
| Number of players entered (normal) | 3 | Code runs, with 3 sets in the array | Y |
| Incorrect number of players entered(Exceptional) | 0 | Error is displayed, goes back to menu | Y |
| Incorrect number of players entered(Exceptional) | “two” | Error is displayed, goes back to menu | Y |
| Incorrect number of players entered(Extreme) | 1 | Code continues but single player | Y |
| Receiving range from user (Normal) | 10 | Program continues with a range of 10 | Y |
| Receiving range from user(Extreme) | 7 | Program continues with a range of 7 | Y |
| Receiving range from user(Exceptional) | 3 | Error is displayed and user is asked to enter range again | Y |
| Receiving range from user (Exceptional) | “ten” | Error is displayed and user is asked to enter range again | Y |
| Inputting user’s lottery numbers(normal) | 8 (when range=10) | Program compiles and asks for next number |  |
| Inputting user’s lottery numbers (Extreme) | 1 (when range=10) | Program compiles and asks for next number |  |
| Inputting user’s lottery numbers (Extreme) | 10 (when range=10) | Program compiles and asks for next number |  |
| Inputting user’s lottery numbers (Exceptional) | 0 | Error is displayed and user is asked for same number again |  |
| Inputting user’s lottery numbers (Exceptional) | 11 | Error is displayed and user is asked for same number again |  |
| Inputting user’s lottery numbers (Exceptional) | (when range=10) | Error is displayed and user is asked for same number again |  |
| Inputting user’s lottery numbers (Exceptional) | “two” | Error is displayed and user is asked for same number again |  |
| Menu option selected | 0 | Program closes | Y |
| Menu option selected | 1 | Method in Lottery class is run with player = 1 | Y |
| Menu option selected | 2 | User is then asked to enter number of players | Y |
| Menu option selected | 5 | Error then menu opens again for user to choose again | Y |
| Menu option selected | one | Error then menu opens again for user to choose again | Y |
| Menu option selected | 3 | pop() and push() methods are tested and message displays saying there are no errors | Y |

Evaluation

Overall I found this assignment quite difficult to get my head around but when I create the shell for the single player lottery I found it quite easy to implement the program for more players. I was not used to using imported set classes and I found the HashSet class confusing but I got to grips of the program quickly. I didn’t face any big problems and I was most proud of myself for implementing a multiplayer mode using an array of sets as I found this part not too difficult.