1. **Classes**

|  |
| --- |
| DataStructureException |
|  |
|  |

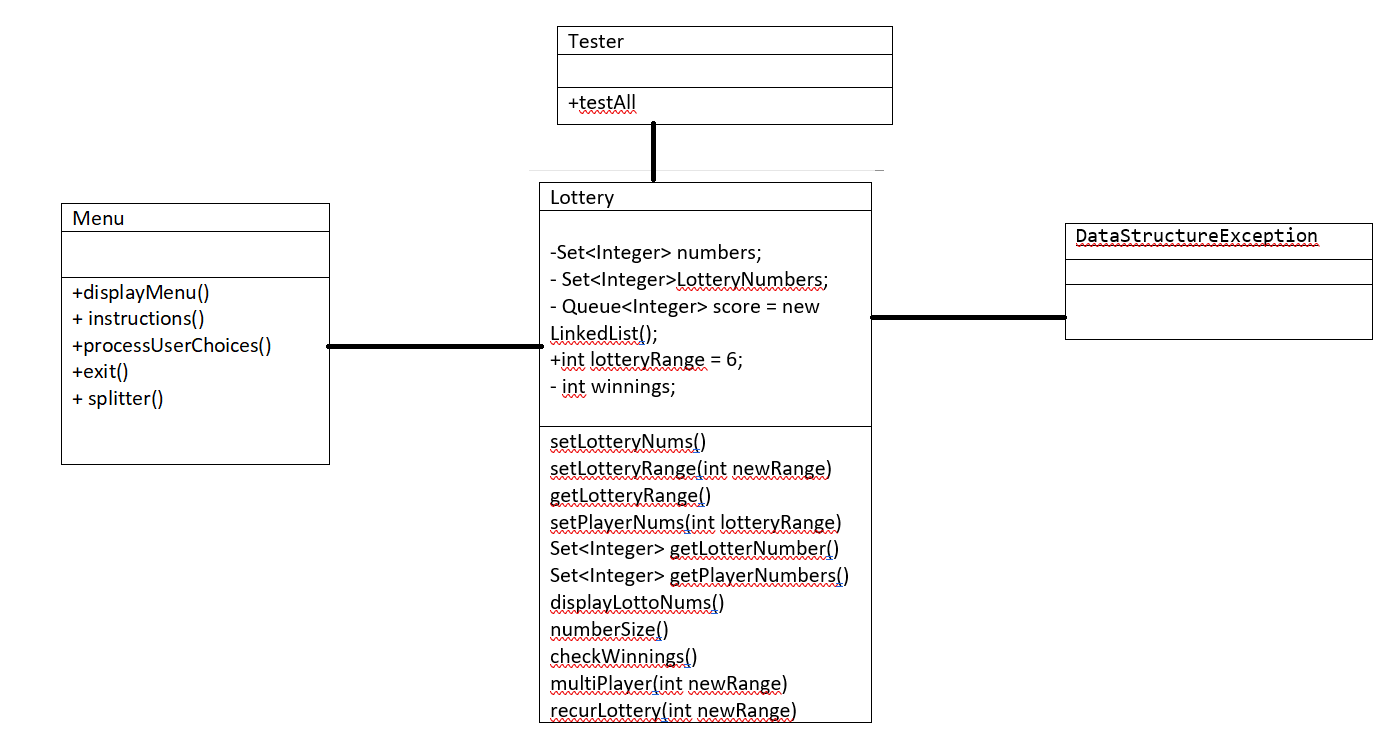
|  |
| --- |
| Lottery |
| -Set<Integer> numbers;  - Set<Integer>LotteryNumbers;  - Queue<Integer> score = new LinkedList();  +int lotteryRange = 6;  - int winnings; |
| setLotteryNums()  setLotteryRange(int newRange)  getLotteryRange()  setPlayerNums(int lotteryRange)  Set<Integer> getLotterNumber()  Set<Integer> getPlayerNumbers()  displayLottoNums()  numberSize()  checkWinnings()  multiPlayer(int newRange)  recurLottery(int newRange) |

|  |
| --- |
| Tester |
|  |
| +testAll |

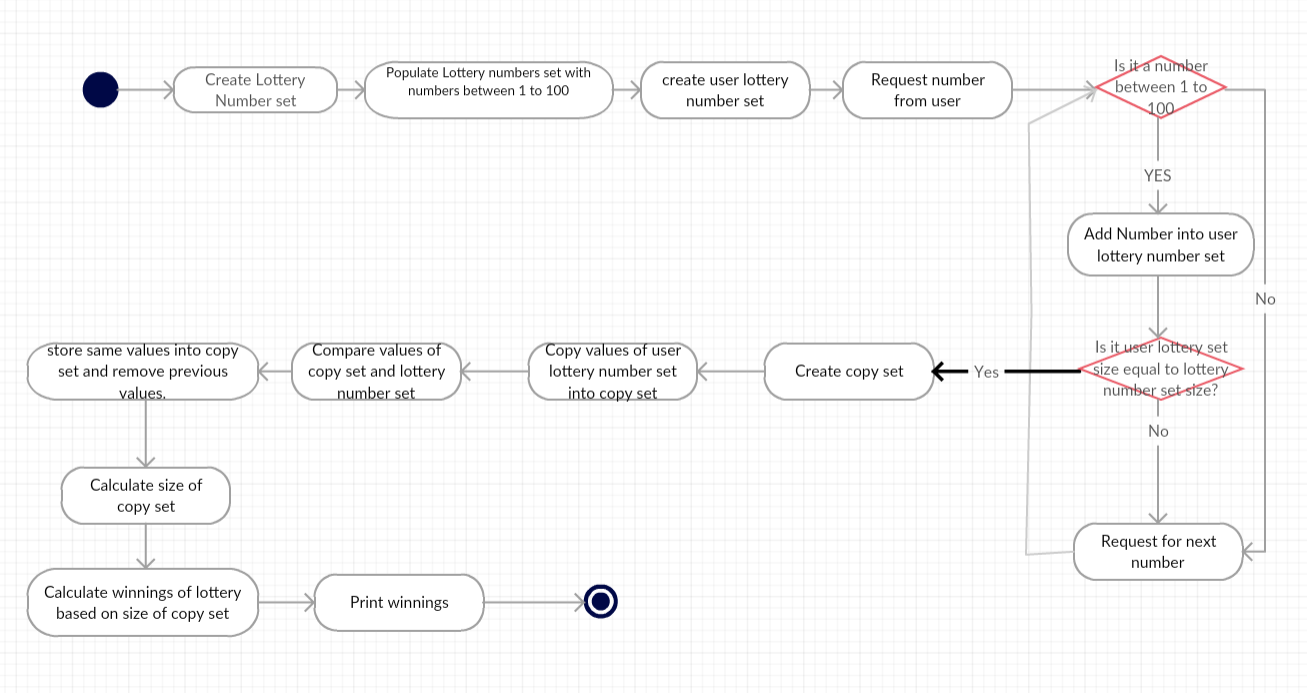
|  |
| --- |
| LotterySet |
| Set<Integer> mySet = new TreeSet<Integer>();  Set< Integer> intersection = new TreeSet<Integer>(); |
| +printSet()  +isSetEmpty()  +getCardinality()  +isInSet(int number)  + intersection() |

|  |
| --- |
| Menu |
|  |
| +displayMenu()  + instructions()  +processUserChoices()  +exit()  + splitter() |

1. **Class Diagram**



1. **Pseudo code (or activity diagram) for the main method**

****

**A summary of the requirements – saying which you were able to tackle and which were successful**

|  |  |  |
| --- | --- | --- |
| **Requirements** | **Attempted** | **Completed(Y=yes,N= no)** |
| **Documentation** |  |  |
| Report & Designs (pseudo code and/or flowcharts), selfevaluation. |  | **Y** |
| Test plan and completed test sheets |  | **Y** |
| **Code** |  |  |
| Take user input and add to a set |  | **Y** |
| Generate random lottery numbers and add to a set |  | **Y** |
| Calculate winnings (including set intersection) |  | **Y** |
| Enable user to define range of lottery numbers. |  | **Y** |
| Extension to several users |  | **Y** |
| User interface, including Menu and input validation |  | **Y** |
| **Optional Requirements** |  |  |
| Implement a MySet wrapper class to handle the set operations. | **A lotterySet class was created , along with the recommended methods. However, the class was not implemented into the program.** |  |
| Allowing a user-specified range for the lottery numbers & simulating a lottery over several weeks |  | **Y** |

Test number/date/version: 12/02 /18 ……………………………………….……..

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected result** | **Worked?** |
| Adding numbers into the stack | Data=10, 15, 16, 18, 88, 100 | Code completes, set is outputted with correct data (  Lottery Numbers:[10, 15, 16, 18, 88, 100])with correct number of items in set | Y |
| Adding erroneous data type into set | Data=ab\* | Code completes with error message(java.util.InputMismatchException  Only enter numbers between 1-100). | Y(tested in program rather than tester class). |
| Set stores numbers in ascending order | Data=10, 15, 16, 18, 100, 88 | Code completes, set is outputted with correct data in order (  Lottery Numbers:[10, 15, 16, 18, 88, 100])with correct number of items in set | Y |
| Set has no duplicates | Data=8,8 | Code completes, set saves only one data value instead of both.  (Lottery Numbers:[8]  Numbers in lottery set:1) | Y |
| Copy set | SetA=8,10,16,18 | Code completes, copies values of setA into *copy\_set* and ouputs values and size of copied set. (Copied set:[8, 10, 16, 18]  Numbers in copied set:4) | Y |
| Intersection | SetA= 8, 10, 16, 18  SetB= 8, 10, 55, 88 | Code completes, tests that the intersection of both sets are identified and outputs the values that intersect(Intersect Numbers:8,10) | Y |
| Set values are not lost during the intersection method. | SetA= 8, 10, 16, 18  SetB= 8, 10, 55, 88 | Code completes, test that the values in both sets are unchanged after intersection method is performed on both sets. | Y |
| Testing that all values are removed | SetA= 15,18,23 | Code completes, all values are removed from the set and correct value is outputted  (Set A:[15, 18, 23]  Set A:[]) . | Y |
| Validate user input for menu, numbers to be place in set, setting size of set, number of players and number of draws. |  | Error message is outputted when user enters an invalid number, character or string and user is brought back to the main menu. | Y |
| Auto populating lotteryNumber set with fixed number of data |  | Code completes, random numbers are added to the set. | Y |
| Populating lotteryNumber set with numbers within a fixed range |  | Numbers are added to the set within the range of 1-100. | Y |
| Calculating the winnings of each player. | Numbers of players;3. | Code runs and the correct amount won by each players calculated and displayed. | Y |
| Run lottery draw for fixed amount of times | Number of draws;5. | Code runs and displays lottery numbers for each draw and any matching numbers followed by calculating total winnings and gross profit. | Y |