Juan Pablo Sanin
A00296776
Lazer Mirror Game

Functional Requirements:

RF1. The program must create a game board using information from the user that indicates the rows(m), columns (n), and mirrors (k).

RF2. The program must receive the information in one line containing the users' nickname, and three positive integers (n, m, k).

RF3. The program must place the k amount of mirrors randomly on the game board.

RF4. The program must show the game board in a grid manner using brackets [ ].

RF5. The program must represent each cell of the game board with an identifier composed of a number and a letter (a number for the row, a letter for the column).

RF6. The program must allow the user to shoot lasers through the game board as many times as it likes using the single line command indicating the cell it will be shot from. The command must have the following order (Number+Letter (3A)), this means the shot will be HORIZONTAL from the third row in the first column. If the cell is a corner the user must indicate the direction of the shot using H for horizontal or V for vertical at the end of the single line command (1AH).

RF7. After each shot, the program must show the board with the start of the shot represented by an S, and the end of the shot represented by an E. ([S], [E] in their respective locations). If a cell is both start and end it will be represented by and M.

RF8. The program must allow the user to locate a mirror using a single line command composed of L for locate + the cell number and letter + the direction of the mirror (R for right, L for left). If the user is correct the program must show the mirror in the game grid, if he is wrong the program must show an X in the game grid. If the user is correct about the cell but wrong about the direction of the mirror then the game board must show an asterisk *.

RF9. Before showing the game board, the program must print in a line above the board the users' nickname, and how many mirrors are left on the board.

RF10. The program must allow the user to go back to the menu by winning or by typing the command menu.

RF11. The program must have a 3-option menu, option 1 is to play, option 2 to show the scores, and option 3 to exit.

RF12. The program must save the player score in a Binary Search Tree.

RF13. The program must traverse the BST using in-order.

Juan Pablo Sanin
A00296776
Lazer Mirror Game

RF14. The program must use Linked List for the game board.

RF15. The program cannot have any arrays except for String arrays from user input (THIS IS THE ONE AND ONLY EXCEPTION).

RF16. The program cannot have any loops, it can only use recursion for its operations.

Non-functional Requirments:
-The program will be coded in Java.
-The program will have a menu class for organizing the operations
-The program will have an easter egg that shows all the mirrors on the board.

Class Diagram: