

NAME: Christopher Nies
ID: A11393577
LOGIN: cs12sfl

Part 3a

1. Since the MineCell is implementing the MouseListener interface, that is where I should look for what happens when the mouse is clicked. In the mouseReleased method, the method first checks whether or not the mouse has been right clicked or left clicked. A right click flags or unflags the method (which serves as a reminder that the cell may have a mine under it. A left click is more interesting, and calls the exposeCells method, which exposes all the cells by calling either exposeCellsBFS or exposeCellsDFS depending on what the user specified in the arguments
2. MineSweeperGUI creates the user interface for the user, which displays the board. The board itself is comprised of MineCells, which are all of the squares the user can click on to either reveal it or flag it. A cell is the fundamental piece of gameplay, the GUI displays it.
3. The MineCell listens for the mouse clicks.
4. exposeSlowly simply animates the exposure of the MineCells, by using a timer and the actionPerformed method inside the class. The animation speed can be changed by changing the "DELAY" constant in the class definition.
5. The actionPerformed method is listening for the actionPerformed sent out by the timer in the exposeSlowly method.

Part 3c

With the Breadth First search, the cells immediately around the clicked cell are exposed starting with the top left, and then reveals all of the cells immediately around each of the clear cells until all of the necessary cells are revealed. With Depth First Search, the cells are exposed starting with the bottom right, which each cell being revealed in a line until it can't go in that direction any more. It then checks all around the last exposed cell on the line, extending as far away from the clicked cell as it can before doubling back around and exposing a closer one. BFS checks all the neighbors around, then all of the neighbors' around those neighbors, etc. DFS checks out a very specific neighbor's neighbor, which checks out a specific neighbor's neighbor, until all of the neighbor's are checked. Fun!