Project Architecture

* Components
  + Cell: stores cell contents/adjacency value and cell status tag; controls display format for terminal interface
  + Board: manages grid as 2D array of cell objects; handles back-end game tasks such as placing the mines, enacting the effects of a user click, revealing and recursively revealing cells, and tracking if the game is lost
  + Main: operates upper-level tasks and functionality for terminal interface; takes user input for game conditions and user moves; manages game state and display updates
  + GUI: operates upper-level tasks and functionality independently from main, for use in GUI; creates interface, gets user input, checks for end state conditions, updates display with changes
* Data Flow
  + User input (click): GUI processes input and makes appropriate call to board to update game state
  + After call to board, GUI redraws game display with new game state
  + User input (parameter change): GUI processes input, changes board parameters, resets game state to new game, redraws display with new parameters and game state
* Data Structures
  + Cell (in cell module): stores cell contents/adjacency value and cell status tag
    - Val: 9 = mine; 0-8 = number of adjacent mines
    - Tag: 0 = hidden; 1 = cleared; 2 = flagged; 3 = triggered
  + Array (in board module): NxN matrix of cell objects
  + Board (board module): holds array, alive status indicator, size of board
* Board Parameters
  + Board size defaults to 10x10, can be set to any size between 4 and 20
  + Number of mines defaults to 10, can be set to any number between 1 and board size squared minus one