**Lab # 7BONUS: Chessboard moves (BONUS PROBLEM) [100 points]**

The purpose of this activity is to get you used to using lists of lists, in a 2-D matrix-like format.

Write a program that sets up a chess board with chess pieces, and lets people make moves (of one piece at a time) on the chess board. Here are the details:

* The chessboard is an 8x8 board.
* Display the chess board before every move. Each empty square should just be a period. Each square with a piece should have the piece’s identifier.
* For identifiers, use lower-case for the white pieces, upper-case for the black pieces.
  + Use P/p for pawn, R/r for rook, N/n for knight, B/b for bishop, Q/q for queen, and K/k for king.
* When a piece is moved, the location it moves from is then an empty square, and it then occupies the square it is moving to. If there was already a piece in the square it is moving to, then that other piece is eliminated from the game.
* **Important:** You do **NOT** need to enforce any rules of chess or verify moves, with one exception:
  + If someone puts in a move from a position where there is not a piece, report an error and exit the program.
  + Other than that, you don’t need to worry about the sides alternating turns, about pieces moving in ways they aren’t allowed, about landing on your own pieces, etc.
  + Only one piece moves at a time, and it can move to any position on the board.
* You can specify the system you want the user to use to specify the starting and ending location.
  + In reality, chess uses a standard system, where the columns are labeled a-h from left to right and the rows are labeled 1-8 from bottom to top. You do not have to use this system, but can if you wish.

As an example, here is what the board would look like at the very beginning of the game:

RNBQKBNR

PPPPPPPP

........

........

........

........

pppppppp

rnbqkbnr

**Remember to think through exactly what you will do before developing it!**