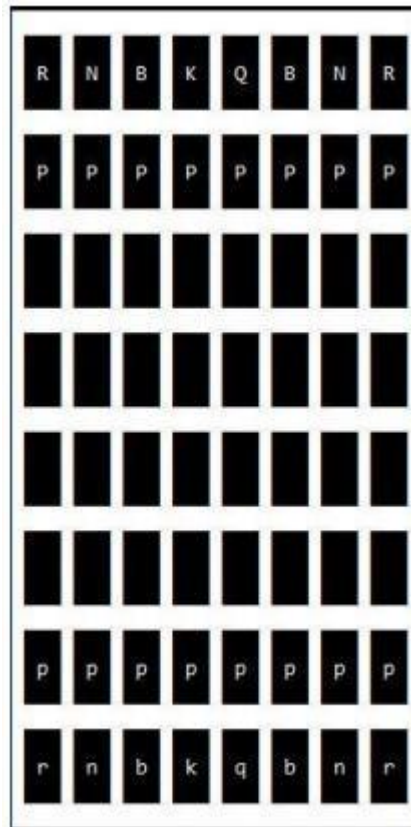


## Project - Chess

- 1) Chessboard of size  $8 \times 8$ .
- 2) Players belong to two different camps, say small letters and large letters, and the small pieces move first.
- 3) The chessboard is oriented, small letters are at the bottom of the board, and larger letters are at the top of the chessboard.
- 4) There are six types of pieces: king (K), queen (Q), bishop (B), knight (N), rook (R), and pawn (P). Each Player starts with **one king, one queen, two bishops, two knights, two rooks, and eight pawns**.
- 5) The chess board implemented is shown below:



- 6) A **King** may move in any direction including diagonally, but may only move one square. A king may not move through other pieces. Also, the king cannot move into the square controlled by enemy's piece, otherwise, count as the foul. The king will be "**checked**" by other pieces, so we should code the special judgement for king.
- 7) A **Queen** may move any number of spaces in one direction, including diagonally. A queen may not move through other pieces.
- 8) A **Bishop** may move diagonally only, but may move any number of spaces. A bishop may not move through other pieces.

- 9) A **Knight** may move in an L-shape, of length either two-by-one or one-by-two. The knight is the only piece that is not stopped by other pieces in its way (i.e., it can move through other pieces to get to an open square).
- 10) A **Rook** may move any number of squares, but only in a straight line which is not diagonal on the board. A rook may not move through other pieces.
- 11) A **Pawn** can move only forward towards the opponent's side of the board, but with restrictions. On its first move of the game, a pawn may move forward either one or two squares; on subsequent moves, a pawn may only move forward one square. A pawn may not move through other pieces. Moreover, the pawn may not use this forward move to capture the opponent player's piece. Instead, the pawn moves one square diagonally forward to capture a piece and have that piece removed from the board.
- 12) Only the player of the opponent team can be killed, keep in mind while implementing that it doesn't kill its own pieces.
- 13) You need to implement "**File Handling**" of the project, so if player wants to load the game the game at the point where he/she left it so that he/she can continue the game.