Chess APCS1 pd 5
Fabiha Ahmed, Kenny Chen, Jasper Cheung 2017-01-10

Final Project Proposal

Chess

We will make the popular board game known as Chess playable in the terminal. It will be a 2-player game (no AI). If time allows, we plan to add Chess variants such as Progressive chess, Chess960, Atomic chess, etc.

Solidify:

- Class hierarchy (types of Piece)

Showcase:

- Use of arrays/arraylist:
    - a 2-d array for grid, where each element of each array in the 2-d array corresponds to a square.
    - ArrayList can also be used to keep track of moves.
- Abstract class (Piece):
    - It doesn’t make sense for Piece to be instantiated, and it contains instance variables (such as color). Thus, Piece should be an abstract class.
- Class hierarchy (types of Piece):
    - King, Rook, Pawn, etc., are all pieces; they define implementations of Piece abstract methods.
    - We use subclasses because that eliminates the need to redefine variables that all pieces share, such as their color.
- Take in user input: User inputs commands/moves through the terminal

Stretch:

- Robust responses to user input:
    - For example: A illegal move is played because the king is in check. Instead of just outputting a generic error message, the message would be specific to the move.

MVP: Pieces on the grid, ability to print out the grid in terminal

Example terminal output:

```
8 | r n b q k b n r
7 | p p p p p p p p
6 |
5 |
4 │
3 │
2 │ P P P P P P P P
1 │ R N B Q K B N R
  └────────────────
    a b c d e f g h

White’s turn:
Enter command or coordinate of piece:   thaowih
Unrecognized command or coordinate of piece "thaowih"
Enter command or coordinate of piece:   a3
No piece at a3.
Enter command or coordinate of piece:   a1
No legal move for R at a1.
Enter command or coordinate of piece:   b1
N at b1; legal moves:   a3, c3
Enter move:   a3

(prints board)

Black’s turn:
Enter command or coordinate of piece:   history
1. Na3
...
```