Chinese Chess - 112
Patrick Huang. CMUID: pbhuang

Project Description: The project will be a working version of Chinese Chess where two people can play on one computer and interact with the board in a 3D environment, by moving a camera to view the board from different angles.

Competitive Analysis:

At least in the 112-project gallery from 2017 to 2019, I have not seen any other student implement a version of Chinese Chess. Compared to online versions of Chinese chess (such as xahlee.info or playok.com), every online version is two dimensional and users are unable to adjust the view of the board in any way. Most of the online versions are multiplayer based in that you are able to play live against online opponents.

My project will be different in regards to providing the players the unique opportunity to view the board from different perspectives. The user will have the ability to rotate the camera so that they can view their pieces and their opponent's pieces from different angles, giving a more pleasant user experience. The main weakness of my project compared to competitors is that I am currently planning to not include multiplayer. If I have time, I also hope to implement multiplayer into my game, but the main component of my project that will differentiate it from online versions of Chinese chess is the ability of the players to adjust their three-dimensional view of the board.

Structural plan:

Each piece and the board are instances of piece and board classes. These classes are included in the global scope outside of the main game class. Functions and variables/attributes are divided between Model, View, and Controller classes. The Model class contains all the numbers and variables necessary for the game to function. It will have no methods. The Controller class contains all the functions that operate on the variables and modify game data. The View class contains all the functions responsible for visually drawing the game on a canvas or for rendering the game. Finally, there will be a main class that is a subclass of "ShowBase" (part of Panda3D) where the main game loop which reside in (i.e how a user will transition between parts of the game)

Algorithmic Plan:

The most algorithmically complicated component of my project is likely getting all the legal moves possible for a cannon piece. The cannon piece has very peculiar rules in Chinese chess. For example, the piece is able to jump over pieces to capture enemy pieces, but it can't simply jump over pieces when it is simply moving around. In that case, it moves just look a rook. I first implemented getLegalMoves. In this function, it loops through all the possibilities for drow and dcol and selects only one (it can only take straight vertically or horizontally). Then it passes in

the selected drow and dcol into getLegalMovesFromPoint. In this function, we check whether there is a “pivot” piece at some row and column in front of the piece in the current position. The direction of “in front” is based on the drow and dcol that is passed into the function. If there is a “pivot” piece, the piece goes into “attacking” mode and any enemy piece that is past the pivot piece. If the piece is not attacking, only the moves that are in the direction before hitting a “pivot” piece is a legal move. All of these are extended to a list of possible legal moves in getLegalMoves which is eventually returned after looping through all the possible drows and dcols.

Timeline Plan:

I plan to finish implementing all game rules and the turn-based nature of the game by Thursday. I then plan to port all of my game logic into a 3D environment complete with camera movement by Saturday. I then hope to implement camera intervals and preset paths, basically constituting transitions between the main menu, game mode, help screen etc…, by Monday. I then hope to clean up my animations and models on Tuesday. I then hope to continue refining my project over thanksgiving break and hopefully add some other modules.

Version Control Plan:

I am using GitHub to have version control of my project and backup up my project files online. I also have git locally for local backups of any changes that I make to my project.

Module List:

- Panda3D
- No other external modules outside of what we learned in 15-112

TP2 Update:

I have made no changes to my original design plan.

TP3 Update:

I have added camera movement functionality between the different parts of the project. Pressing “g” goes to the instructions screen. In this part, use the left/right arrow keys to switch between slides. Do not spam click the keys, wait for the one second animation to finish. Click esc to go back to the main menu or space to go to the game. In the game, the arrow keys will rotate/tilt the camera while the WASD keys move the camera around. “J” zooms in while “K” zooms out. Selected pieces will be bright green, a king in check will be purple, a legal move will be dark blue, a king in checkmate will be white, and a legal move involving taking a piece will be orange. Clicking “n” anytime in the game will start a new game by resetting the board and camera. You

can press “g” in game to go to the instructions and “space” in instructions to go straight to the game.