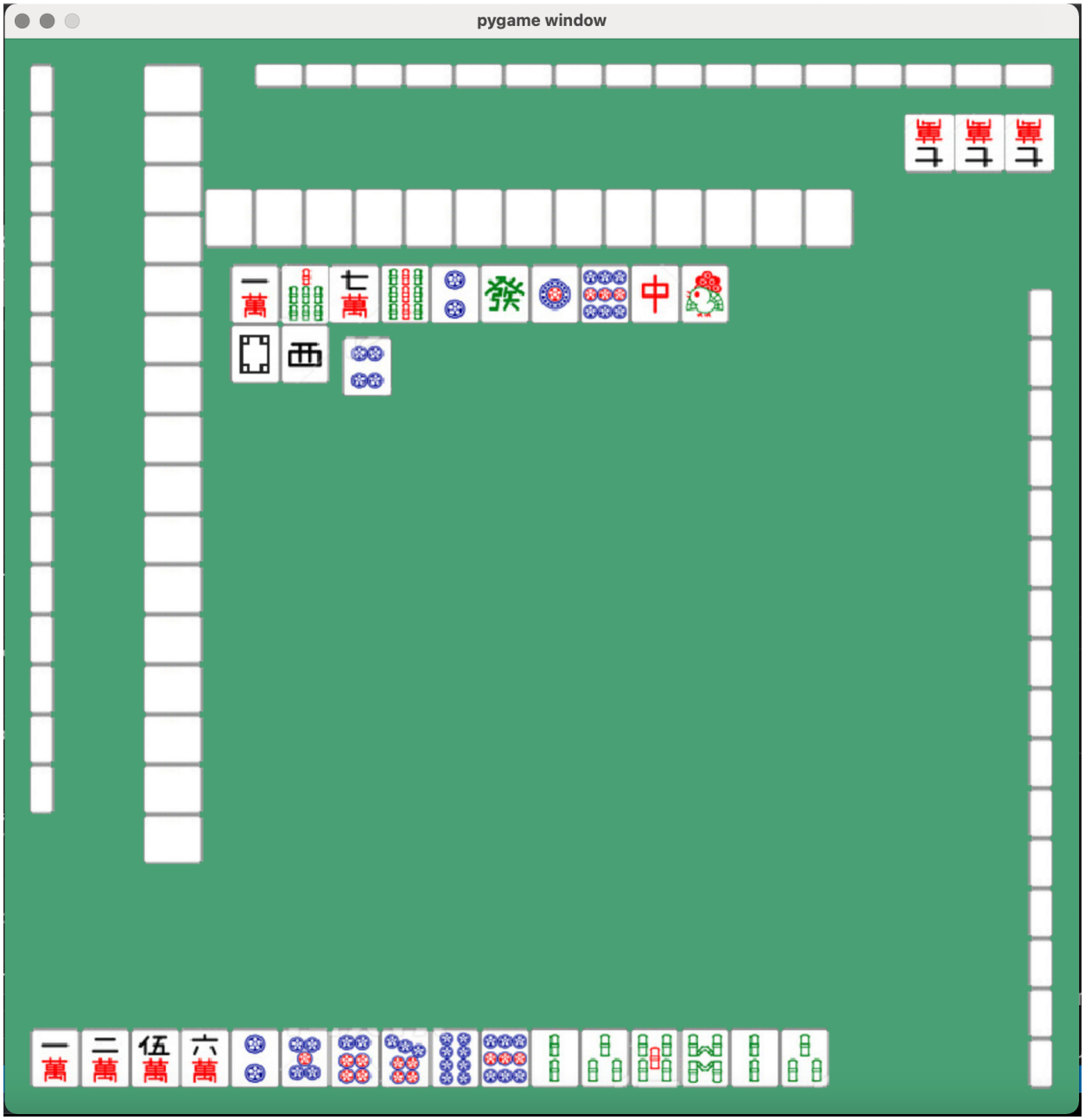
Proposal

TP2 UPDATE:

I've moved things around a bit since TP1, below is what I have right now:



The thin lines on the sides are the other player's (AIs at the moment) hands, and the lines of face down tiles are the deck. The face up tiles in the middle are the 'dead' tiles (tossed tiles that are no longer accessible), and the tile next to the dead tiles is the most recent tossed tile that can still be accessed.

Another update is I made tile drawing automatic if a player cannot perform an action.

I also plan on incorporating a sidebar on the right side of the screen that'll have buttons to allow the player ro pong / chi / etc., and a text box on the top to say when players perform actions.

Mahjong - I'll be making a network-based mahjong app that will allow users to play with each other over a network, or play against an AI

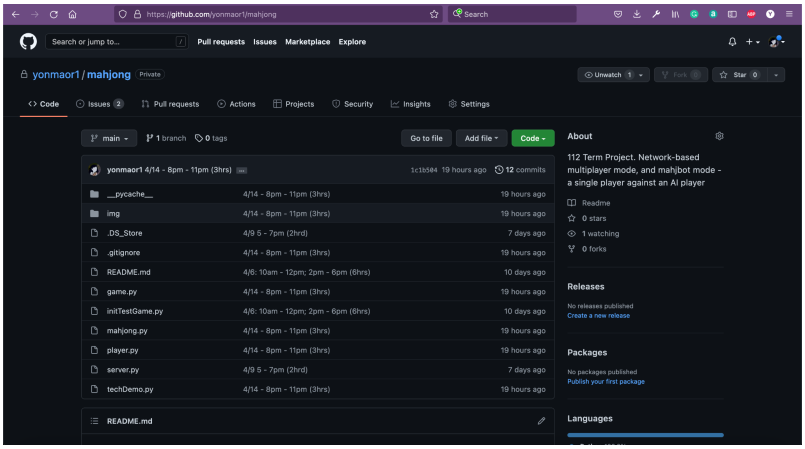
Competitive analysis: I found around 4 other mahjong term projects, many of which had some of the features I'm thinking of integrating into mine, others had features or techniques I dont like and want to make sure I avoid. One big thing I've seen and don't want to do is have a solely local single player, where the computer has to be passed around in order to make moves. I think this is clunky and not very elegant. That's why my priority is to first implement a single player AI mode, then do network based multiplayer. Many of the other mahjong apps I've seen allow players to easily select tiles by pressing on them, which I like and will implement into my mahjong. One thing I didn't see in any other mahjong apps is sophisticated graphics. I'd like to animate the tiles sliding from their original location to their destination, coupled with a sliding sound, because I feel like the tile sliding and clanking is a pretty essential part of playing mahjong.

Structural Plan: I am implementing the players as objects, and one of their attributes is a 'hand' which contains their tiles, in the form of tuples (value, suit). The player class is in its own file (player.py), mahjong.py contains game play functions, then startGame.py runs the game and the pygame graphics. I will also need a server file once I begin to work on a network based mp.

Algorithmic Plan: There are 3 algorithmically complex components of my project - the AI, win detection, and sliding. I've already written the AI and win detection. The AI as of now always reveals sets if it's able to with a tossed tile, then goes through its hand and finds the first tile that isn't in a pair, or set to toss out. The win detection uses backtracking to try and assemble a hand into complete sets (it's a bit more complicated as it does this backtracking algorithm on every possible pair in the hand). For sliding, I plan to look into bezier splines and animating the tiles to follow a spline from point A to point B while not going through any other tiles on the table.

Timeline Plan: I currently have working a local multiplayer mahjong, with an AI option, no graphics. I hope to be able to have a basic graphical interface (no sliding, just displaying everything that needs to be displayed and graphics-based tile selection) by the middle of this week (4/20), then implement network code by the end of that week (around TP2), then work on sliding between that and TP3.

Version Control: I'm using github to backup my files. I use github desktop to commit my edits.



Module List: pygame, sockets, threading