Ethan DiPilato, Brett Grossman, Abriana Nash

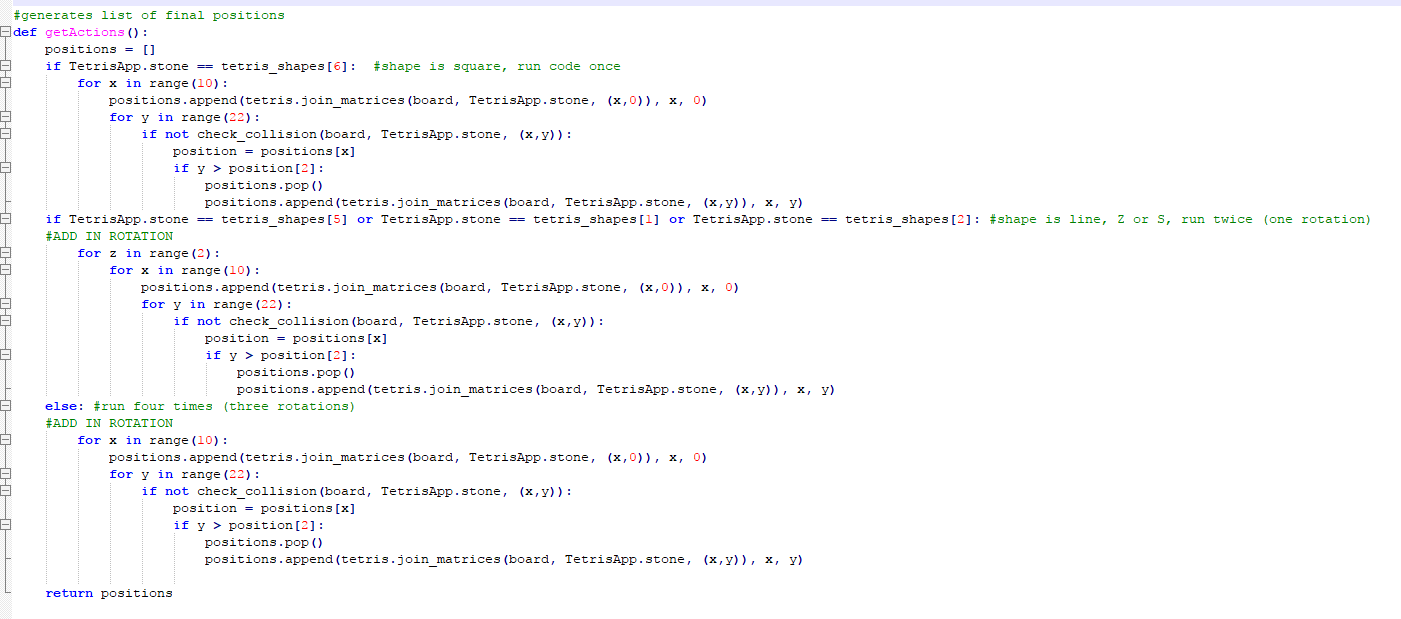
7/13/17

AI Final Project

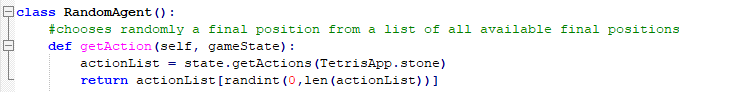
Project Update

The current priority will be to get done ASAP the replacement of keyboard inputs with directions from the solution path. The tasks left to follow are finishing the getActions function, writing the greedy agent, writing the BFS search to generate a solution path from the outputs of the various agents. The work will be divided between Brett and Ethan if it is determined that Abriana is not able to contribute for the remaining project time.

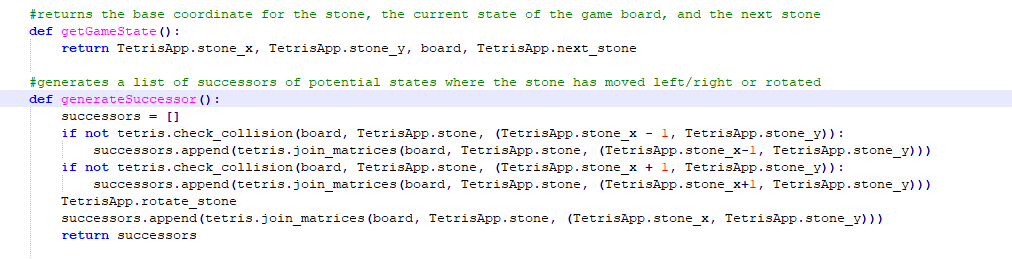
This is the getActions function, still a work in progress, it will generate a list of all final positions at the bottom of the board that the piece it is provided can be placed.



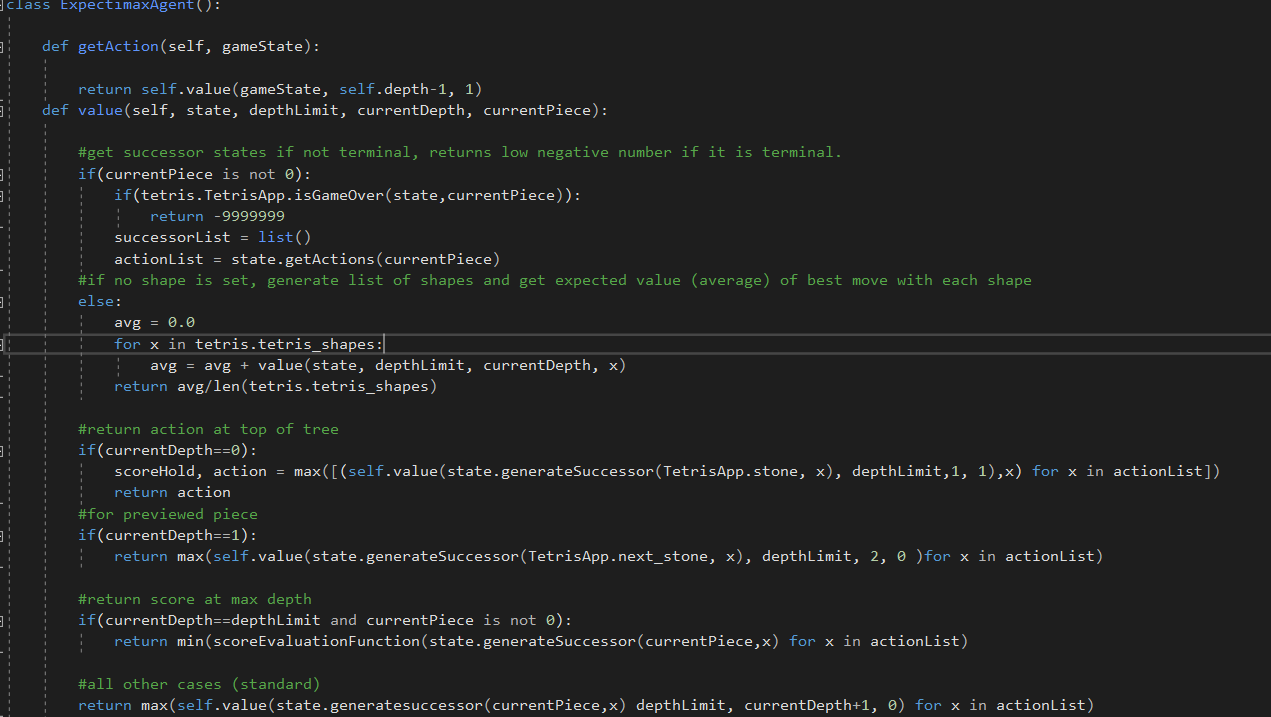
This is the RandomAgent which randomly chooses a final position for the given piece from the list provided by the getActions function.



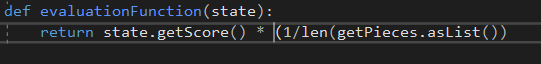
The successor and game state functions.



The expectimax agent that we hope to produce our best results with. Takes max scores at top of tree at first layer (current turn) and second turn (using previewed piece). Then uses averages of max possible scores of each shape at all future layers.



Rudimentary evaluation function that we hope to further tune once it is testable. Currently takes the game’s score and divides it by the number of filled squares on the board.



Boolean game over method added to the tetris.py file. The tetris.py we are using did not have this method, and instead checks for a game over as part of its method to start a new turn. We added this method for use in the expectimax agent in order to more easily determine terminal nodes.

