Mini Checkers Report

Dhaval Pathak
dhp287 / CS6613

Instructions to compile: -
Used python 3 for developing the game. Used pygame for developing UI.
To compile and run the program type the following on your command prompt: -
python filepath/filename.py

High level description: -
Designed and implemented mini-checkers for a human to play against a computer (artificial agent). The black pieces are for humans and white pieces are for the computer. The program begins with selection of level at which human wants to play the game. Level 1 is easy, level 2 is intermediate and level 3 is difficult.

After selecting the level, the game begins, first turn is of human. Human selects a piece to move, as soon as the piece is selected it is checked whether that particular piece has any legal moves if the piece doesn't have any legal moves an error is displayed on the pygame window and the human is asked to select a piece again, if the piece has legal moves then human is asked to select an end position for the selected piece. As soon as the end position is selected for the piece program will check if that particular move is valid, if the move is invalid an error will be displayed on pygame window and human will have to select the piece and end position again, if the move is valid and it is a regular move the piece will be moved to its new position, if it's a capture move then the piece will be moved to its new position and the opposition piece in between will be eliminated.

After human has made his/her move its computers turn to make a move. The program will call the alpha_beta function on the current state of the board. Alpha_beta function will search for all the legal moves that the artificial agent can make and the cutoff time will be divided among all the legal moves i.e. for example the cutoff time is 15 seconds and there are 3 legal moves then the alpha beta pruning algorithm will run for 5 seconds (15 / 3 = 5) for each of those 3 moves and will select the best move based on the scores returned by each move. After the legal moves are searched the alpha_beta function will call the minimum function on the first legal move this minimum function will search for a move associated with the black piece i.e. human player, after selecting the move the minimum function will call the maximum function and the maximum function will select the next move for the artificial agent and this will continue until the board reaches its terminal state or the cutoff time is reached, at this point the program, will calculate the utility score of that particular move and return that utility score. This will happen for all the legal moves and the move with best score will be selected.

After the artificial agent makes its move the control will gain be transferred to the human, this will continue until the board reaches its terminal state and the game is over, at this point the program will count the number of pieces of each player and if the number of pieces are equal it is a draw else the player with more pieces on board will win. If in middle of the game of the game if all the pieces of a player are captured, then at that instant the program will terminate, and the that payer will lose.

Mini Checkers Report

Dhaval Pathak
dhp287 / CS6613

Terminal states: -

The terminal state of the game is reached when there are no legal moves left for both the players or there are no black pieces left or there are no white pieces left.

If the game reaches a terminal state where there are no legal moves left for both the players then the program will assign (10 + row number of that piece) points to each of the white pieces, (10 + row number of that piece) points to each of the black pieces and return the difference of white piece score and black piece score.

If there are no black pieces left the artificial agent automatically wins and if there are no white pieces left, then the human automatically wins.

Evaluation function: -

Whenever the we reach the cutoff value the program will call an evaluation function, evaluate function calculates the utility score of the current state of the board and returns the score. It calculates the score of the white pieces and the score of the black pieces and returns the difference of both the scores.

If a piece cannot be captured i.e. no opposition piece is left in front of that piece, then it gets 30 points, if a piece can capture the opposition piece it gets 20 points and if a piece is not in the above two categories it gets (10 + row number of that piece) points.

Difficulty Level: -

The difficulty is managed by the cutoff level of the alpha beta pruning algorithm tree, if the difficulty level is 1 then the cutoff is set to 0 seconds, if the difficulty level is 2 then the cutoff is set to 8 seconds and if the difficulty level is 3 cutoff is set to 15 seconds.