

## A Chess Game: *Othello*

The project I plan to do is the implementation of a chess game called *Othello* (also known as the modern version of *Reversi*). I was once asked to build this game using Python as a homework assignment from the AI class last year. The reason why I am interested in this particular project is because I remember how my Python Bot acted poorly in the end. Specifically, with a timeout of 10 seconds (that is, a bot needs to make a move within 10 seconds), it is only able to produce a depth of ~20 Minimax tree (i.e. think ahead ~10 moves) even with the optimization of alpha-beta pruning. I can imagine that it is not an unbeatable “master” and there should be many other ways to improve the chess bot.

I have some knowledge of game playing algorithms and I think it would be cool to implement this game again using Haskell. I would like to see how better performance could parallel programming offer to this game.

I plan to do this project by myself and personally I wish to focus more on fields like rules are enforced correctly and the chess bot is trying hard to win in my project other than fields like fancy user interface or visualization. But I am open to any potential change.