Project 2

Objective

Write a chess AI with traditional techniques like minimax search and alpha-beta pruning

Prerequisites

python2, pip

Instructions

Download handout.tar from the ftp or from HERE, extract it and you will fild `cli.py' and `agent/' folder. Modify the `agent/src/agent.py', and fill your code in the function `respond_to' which receives a string in FEN(Forsyth-Edwards Notation), and returns a move in string. Please refer to the `Move format' section of the UCI(Universal Chess Interface) specification about the movement presentation. Once you are convinced that you have a working agent, you should upload it to our server as following:

- Make sure you are using python2, check the output of `python -V', it should be 2.7.x or 2.6.x
- Install the dependencies, `pip install requests click' if there is an error, add sudo in the front
- `python cli.py deploy <YOUR STUDENT NUMBER> [THE PATH TO YOUR AGENT FOLDER]', the path can be ommited, *please try to kill your agent first before you deploy a new version*
- `python cli.py launch <YOUR STUDENT NUMBER>'
- open `http://10.141.209.144:54321/board/<YOUR STUDENT NUMBER>' in your browser to test your agent.

Notice

- Your agent must respond in *3* seconds, otherwise you lose the match
- Your agent must make legal moves, otherwise you lose the match
- You must NOT upload any malicious code, otherwise you get ZERO points

Library Reference

You are allowed to use the library `python-chess' whose document can be found HERE

Deadline

Dec. 24, 2017