Collaboration policy for homework: This homework is to be done by yourself. Please do not collaborate with others, look for answers online, or post homework problems or solutions to websites or discord servers. **We encourage you to come to office hours for help.**

For the code, modify the provided `hw3.py` and submit that.

Problem 1. (30 points)

For problem 1, you will put answers in your writeup for a - d. Also make alterations to the given python module.

a. Run the `example_problem` code. You will be prompted to play tic-tac-toe against a minimax agent. Common wisdom suggests that the best first move is the center square. The agent doesn't begin by playing in the center square. Why do you think the agent plays where it does instead of the center?

b. Run `example_problem_2`. You will be prompted to play connect-4 against an alpha-beta with cutoff agent that is using a terrible evaluation function. See if you can defeat the agent in a few games. What is your best plan to defeat it? Why?

c. Write a new evaluation function that isn't terrible. Using this function, write code that will play an alpha-beta agent with depth cutoff 4 using your evaluation function against a random agent. Describe your evaluation function in your writeup and explain why you think it should do well.

d. In the function `problem_1d`, run 20 games of a random agent versus your alpha-beta-cutoff agent, with each agent playing 10 times as X and 10 times as O. Report your results in your writeup.

Problem 2. (30 points)

For problem 2, you will put answers in your writeup for a - b. Also make alterations to the given python module.

a. Find the definition for Gomoku (5-in-a-row) in the aima code. Build an evaluation function for gomoku. In your writeup, briefly explain what your evaluation function is doing.

b. In the function `problem_2b`, run 20 games of a random agent versus an alpha-beta-cutoff agent using your evaluation function, with each agent playing 10 times as X and 10 times as O. Report your results in your writeup.

Problem 3. (40 points)

For problem 3, you will put answers in your writeup for a - d.

a. Construct a game tree with branching factor of 2 with 15 nodes that would be maximally pruned by alpha-beta pruning. Show which nodes would be pruned.

b. Suppose you have an oracle, OM(s), that correctly predicts the opponent's move in any state. Using this, formulate the definition of a game as a (single-agent) search problem. Describe an algorithm for finding the optimal move.

c. In this question and the following, an "expectimax" tree consists of a max node at the root with alternating layers of chance and max nodes. At chance nodes, all outcome probabilities are nonzero. The goal is to find the value of the root with a bounded-depth search.

   If leaf values are all nonnegative, is alpha-beta pruning ever possible in an expectimax tree? Give an example, or explain why not.

d. If leaf values are all in the range [0,1], is alpha-beta pruning ever possible in an expectimax tree?