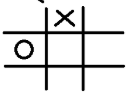
**Final Exam CS2400 Spring 2020 70 points Name:** David Schulz

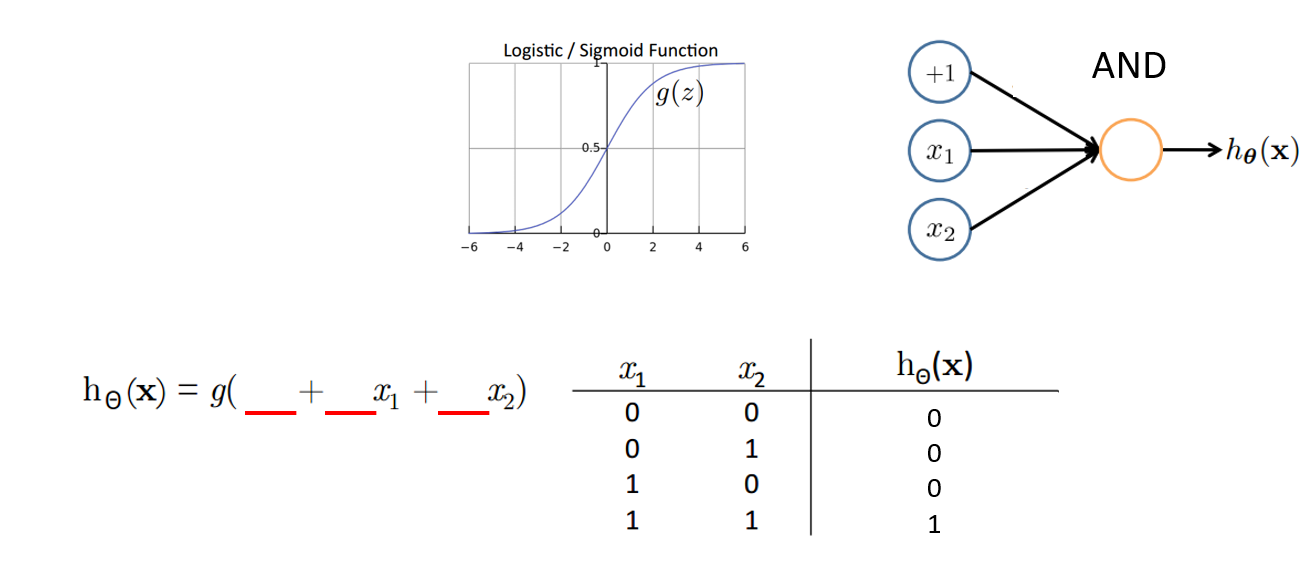
**Instructions:** This is a take home exam, and you have until 5/22 at 11:59PM to submit the exam on Blackboard. Please read through ALL the questions before starting to answer any and reach out to the instructor with questions as early as possible. The instructor will not be responding to email or Teams messages after 4PM on 5/22, so plan accordingly. The instructor will be available during the “lecture” time of the exam on 5/22 from 2-2:50PM, so you are strongly encouraged to work on the exam during that time and reach out via Teams when you have questions.

**Academic Honesty:** You are welcome to use any materials from class or online on this exam to help you develop your answers, but your answers must be your own words/diagrams (no copying/pasting from the book or internet, although paraphrasing is okay). No interaction with another human (other than the instructor) is allowed **including question clarifications**. You are welcome to discuss the questions on the exam AFTER the due date with your peers. You MUST type your name in at the bottom of this document to certify that you met these expectations for your exam to be accepted for credit.

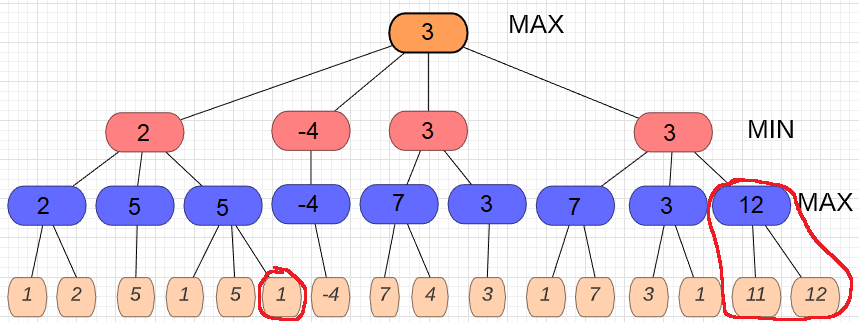
1. (3 points) Describe how you would identify whether a problem is better solved as a classic search problem or an adversarial search problem.
   1. If the problem involves two individuals and one benefits from the other’s loss and vice versa, then it would be considered an adversarial search problem. Otherwise, it’s a classic search problem.
2. (2 points) Why does the depth-first search algorithm keep track of previously-visited nodes?
   1. To make sure it doesn’t get caught in a loop checking the same states over and over.
3. (5 points) Describe a search problem where the inefficiencies of iterative deepening are acceptable. Justify why your chosen situation describes an acceptable tradeoff. Estimate the branching factor and give the approximate time penalty numerically (include your work).
   1. A problem where the inefficiencies are acceptable could be a game of tic tac toe because the branching factor is fairly low. A lower branching factor reduces the time penalty. The branching factor of tic tac toe decreases as the game progresses because the number of possible moves decreases, but the starting branching factor is 9. The approximate time penalty might be somewhere around 0.25.
4. (3 points) What is the heuristic for the following tic-tac-toe board, assuming X went first?
   1. (MAX placing Xs): 6 – 6 = 0



1. (5 points) Why is a sigmoid function generally preferable to a linear activation function? What types of data are sigmoid activation functions typically well-suited for?
   1. It generally classifies the input into one of two distinct outcomes: closer to 0 or closer to 1. A linear activation function doesn’t make that kind of adjustment. It generally keeps the distribution of the outputs the same as the inputs. The sigmoid function typically works for neural networks that are determining one of two possible outcomes, like a boolean algebra problem.
2. (10 points) Identify weight and bias values (3 red underlines) for the AND network shown below:
   1. w1 = 8
   2. w2 = 8
   3. b = -12



1. (5 points) For a neural network with 5 input neurons, 10 hidden layer neurons, and 3 output neurons, how many weights are there to be tuned?
   1. (5\*10) + (10\*3) = 80 weights
2. (5 points) Which will typically converge faster (less wall time): Gradient descent or stochastic gradient descent? Why?
   1. Stochastic gradient descent because it trains with multiple smaller subsets of the whole training data instead of training with the entire set all at once, making it less computationally expensive.
3. (2 points) The output probabilities for an MDP from any state must sum to:
   1. 1.0
4. (5 points) Describe an effective reward function for an agent playing a game of tic-tac-toe. Why shouldn’t a reward function include any false positive results?
   1. The reward function should increase either when the agent places its Xs or Os together in any of the three directions or when it blocks the other player from getting a win. The reward function should also be negative when the opponent wins. It shouldn’t include false positives because it’s possible false positives may still be beneficial for future moves.
5. (5 points) What is a discount factor in RL? What is it intended to model/replicate?
   1. A discount factor is used to reduce the reward as it searches more and more steps ahead of the current state. It models the reduction in benefit as something takes longer to accomplish.
6. (5 points) If a Genetic Algorithm was used to solve the nurse scheduling problem, what would an example evaluation function be?
   1. By “evaluation” function, I’m assuming you mean the “fitness function”. A possible idea for a fitness function could be the average number of shifts each of the n nurses have and comparing it to 21/n, with 21 representing the 3 shifts a day for 7 days in a week. The closer the average is to 21/n, the more “fit” the solution is.
7. (1 point) T/F Genetic algorithms do not support multi-objective optimization evaluation functions.
   1. T
8. (1 point) T/F Mutation requires 2 “parents”
   1. T
9. (1 point) T/F Reproduction in Genetic Algorithms can include either or both mutation and crossover.
   1. F
10. (2 points) T/F Genetic Algorithms can often solve the same problems as A\*. Give an example.
    1. T. With our rat dungeon crawler lab, a genetic algorithm could be used to find the optimal path through the rooms through mutations, crossovers, and selections rather than with heuristics from A\*.
11. (10 points) Please trace the minimax algorithm for the tree below. Give the resulting weight at each node of the tree. Indicate the nodes that would be pruned if using Alpha-Beta pruning.



**I certify that I have met all the academic honesty expectations outlined at the beginning of this exam by typing my name (to serve as a proxy for a signature) here:** David Schulz