Name:
Student ID #:

CS 486 Midterm
Total: 150 pts

Instructions: You will have 105 minutes to complete this midterm. The test is closed book; no notes, computers, or calculators are allowed. Please show all relevant work and calcuations in order to receive partial credit (when appropriate). Many of the questions on this test are short answer, and are designed to get you to think about or explain the topics we have covered. You should answer these at an appropriate length and explain your reasoning when asked to do so. A one-word answer is usually too short; a one-page answer is usually too long. You may use the backs of the pages if you need extra space.

1 True/False, plus corrections

35 pts, 5 pts each

Each of the following statements is either true or false. If it is true, mark it true. If it is false, correct the statement so that it is true. **Note:** Adding "not" or otherwise negating the sentence is not acceptable. You must change the facts in the sentence if it is false. For example:

Question: The Turing Test is a test of whether a computer program is rational.

Bad answer, no credit: The Turing Test is *not* a test of whether a computer program is rational.

Good answer: The Turing Test is a test of whether a computer program is indistinguishable from a human.

- A stochastic environment is one in which an action may have more than one result.

- The Schema Theorem says that more fit bitstrings will be selected with exponentially increasing likelihood.

- A rational agent is an agent that does what a human would do in any situation.

- A toy problem is a problem used to illustrate or study the performance of an algorithm.

- A complete algorithm is one that always finds all possible solutions to a problem.

- Mutation in genetic algorithms and temperature in simulated annealing both improve performance by eliminating suboptimal solutions.

- A Goal-based agent selects actions that are likely to maximize its utility.

2 Agents and environments

a. 24 pts, 6 pts each

For each of the following agents and environments, characterize the environment according to the six properties used in R & N (static vs dynamic, discrete vs. continuous, fully vs. partially observable, deterministic vs stochastic, episodic vs. sequential, single-agent vs. multi-agent). If necessary, you may want to give a **brief** justification of your answers.

- The game of checkers.

- An autopilot that successfully lands a plane.

- An handwriting recognition agent that reads the addresses on envelopes and places each of them in the appropriate mailbox.

- A music-playing accompanist agent that can play along with a human piano player.

b. **6 pts**

Consider the following problem: We want to build an agent that can play word games.

For our first task, we would like to build an agent that can recognize palindromes. (A palindrome is a word or phrase that reads the same backwards as forwards excpeting punctuation, such as "Anna" or "Madam, I'm Adam", or "A man, a plan, a canal - Panama!")

Assume that our agent receives its input one letter at a time. That is, at time t=0 it sees 'a', and and time t=1 it sees 'n', and so on. Once the string is seen completely, the agent must output 'yes' or 'no'.

Can a reflex agent solve this problem? If so, provide pseudocode for how such an agent might work. (You may use the back of the page if needed.) If not, explain **precisely** why it cannot solve this problem.

For our second task, we want to build an agent that can find anagrams. That is, rearrangements of a phrase that are also a phrase. For example, "scab" can be rearranged to "cabs", and "artificial intelligence" can be rearranged to make (among other things): "Intergalactic lie? Fie! Nil!" (punctuation added).

Let's assume that we want to build a goal-based agent that will use search to find an anagram for an arbirtary input phrase. Your task in this problem will be to formulate the problem precisely enough so that it can be implemented. In grading this, I'll ask two questions. 1) Is the answer correct? 2) Could I write a program from this answer without any additional information?

a) **4 pts** Give an initial state for this search.

b) **4 pts** Give a description of how a goal test could be implemented.

c) **4 pts** Describe what the successor function does. In other words, for a given input, what is produced?

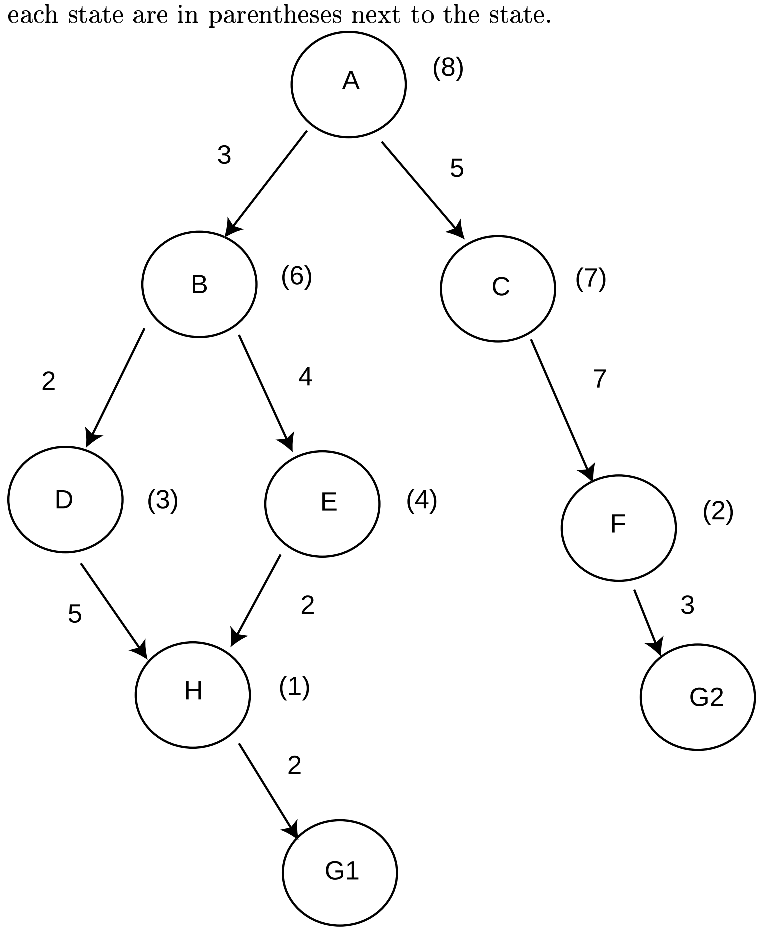
3 Search

a) **(4 pts each)** Give the **time and space** requirements in big-O terms for each of the following algorithms. Assume that the search tree has a branching factor of b, the solution is at depth d, and the tree has a maximum depth of n.

- Breadth-first search
- Depth-first search
- Depth-limited search (depth $= d$)
- Iterative deepening search

b) **18 points - 6 points for each search**

Trace BFS, DFS, and A* on the following graph. Uninformed search methods should expand nodes from left-to-right. (B should be visited before C.) Both g1 and g2 are goal states; your algorithm may stop as soon as either is expanded. Show the order in which nodes are visited. For A*, also show the f,g, and h costs for each node visited. For A*, the costs for each edge are indicated next to the edge, and the heuristic costs for each state are in parentheses next to the state.



4 Heuristics

a. **4 pts**

Consider the Tower of Hanoi problem. There are three towers, with N disks of increasing size stacked on the leftmost tower. The problem is to move the disks from the left-hand to the right-hand tower, with the constraint that a larger disk can never be placed on top of a smaller disk.

To begin, let h be the number of disks that still need to be moved. Is this heuristic admissible? If so, prove it. If not, provide a counterexample.

b. **6 pts**

Next, consider the following heuristic:

Number the disks according to size, where 1 is the smallest disk and n is the number of the largest disk. h is the sum of the numbers of each misplaced disk.

Which heuristic should be used? Justify your answer with a precise technical argument, using the appropriate terminology.

5 pts

c. Use an admissibility argument to show that uniform cost search is optimal.

5 Genetic Algorithms

a) **10 pts**

Consider the following fitness function:

$fitness = 5a + 3b * c + -d + 2e$

where a-e are indices of the bitstring. Compute the fitness of each of the members of the initial population below, as well as the probability that each string will be selected using roulette selection.

a b c d e	fitness	selection probability
1 1 0 1 1		
0 1 1 0 1		
1 1 0 0 0		
1 0 1 1 1		
1 0 0 0 0		

b) **4 pts**

Assuming the first two members of the population are selected for reproduction, and the cross-over point is that between the b and the c, show the resulting children:

6 Adversarial Search

10 pts

For the following tree, give the backed-up minimax value for each node, assuming that the Max player chooses first at node A, and can choose B or C. What is the optimal sequence of plays?