UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
Department of Electrical and Computer Engineering

CS 440/ECE 448 Artificial Intelligence
Spring 2020

PRACTICE EXAM 1

Monday, February 24, 2020

- This is will be a CLOSED BOOK exam. Book and notes are not allowed.
- No calculators are permitted. You need not simplify explicit numerical expressions.
- This exam is more than twice as long as the actual exam will be.
- You must SHOW YOUR WORK to get full credit.

Name: ____

netid: ____

Problem 1 (4 points)

Give one reason why a test of intelligence should evaluate action rather than thought. Give one reason why such a test should evaluate thought rather than action.

Problem 2 (4 points)

How can an agent think like a human without thinking rationally?

Problem 3 (4 points)

How can an agent think rationally without acting rationally?

Problem 4 (4 points)

Discuss the relative strengths and weaknesses of breadth-first search vs. depth-first search for AI problems.

Problem 5 (2 points)

In the tree search formulation, why do we restrict step costs to be non-negative?

Problem 6 (4 points)

What is the distinction between a world state and a search tree node?

Problem 7 (3 points)

How do we avoid repeated states during tree search?

Problem 8 (4 points)

Explain why it is a good heuristic to choose the variable that is *most* constrained but the value that is *least* constraining in a CSP search.

Problem 9 (2 points)

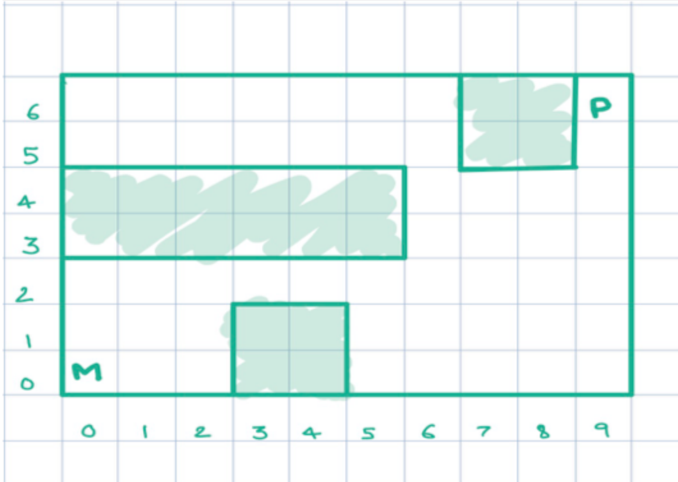
Can an environment be both known and unobservable? Give an example.

Problem 10 (2 points)

What are the main challenges of adversarial search as contrasted with single-agent search? What are some algorithmic similarities and differences?

Problem 11 (8 points)

Refer to the maze shown below. Here, M represents Mario, P represents Peach, and the goal of the game is to get Mario and Peach to find each other. In each move, both Mario and Peach take turns. For example, one move would consist of Peach moving a block to the bottom from her current position, and Mario moving one block to the left from his current position. Standing still is also an option.



(a) Describe state and action representations for this problem.

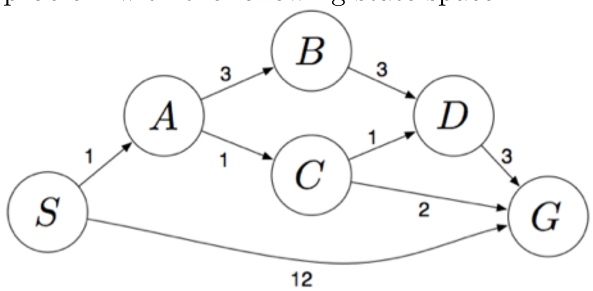
(b) What is the branching factor of the search tree?

(c) What is the size of the state space?

(d) Describe an admissible heuristic for this problem.

Problem 12 (10 points)

Consider the search problem with the following state space:



S denotes the start state, G denotes the goal state, and step costs are written next to each arc. Assume that ties are broken alphabetically (i.e., if there are two states with equal priority on the frontier, the state that comes first alphabetically should be visited first).

(a) What path would BFS return for this problem?

(b) What path would DFS return for this problem?

(c) What path would UCS return for this problem?

(d) Consider the heuristics for this problem shown in the table below.

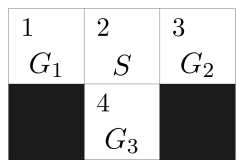
State	h_1	h_2
S	5	4
A	3	2
B	6	6
C	2	1
D	3	3
G	0	0

(i) Is h1 admissible? Is it consistent?

(ii) Is h2 admissible? Is it consistent?

Problem 13 (10 points)

Imagine a maze with only four possible positions, numbered 1 through 4 in the following diagram. Position 2 is the start position (denoted S in the diagram below), while positions 1, 3, and 4 each contain a goal (denoted as G_1, G_2, and G_3 in the diagram below). Search terminates when the agent finds a path that reaches all three goals, using the smallest possible number of steps.



(a) Define a notation for the state of this agent. How many distinct non-terminal states are there?

(b) Draw a search tree out to a depth of 3 moves, including repeated states. Circle repeated states.

(c) For A* search, one possible heuristic, h_1, is the Manhattan distance from the agent to the nearest goal that has not yet been reached. Prove that h_1 is consistent.

(d) Another possible heuristic is based on the Manhattan distance $M[n, g]$ between two positions, and is given by

$$h_2[n] = M[G_1, G_2] + M[G_2, G_3] + M[G_3, G_1]$$

that is, h_2 is the sum of the Manhattan distances from goal 1 to goal 2, then to goal 3, then back to goal 1. Prove that h_2 is not admissible.

(e) Prove that $h_2[n]$ is dominant to $h_1[n]$.

Problem 14 (10 points)

For each of the following problems, determine whether an algorithm to optimally solve the problem requires worst-case computation time that is polynomial or exponential in the parameters d and m (assuming that P $\neq$ NP).

(a) A map has d regions. Colors have been applied to all m regions, drawing from a set of m possible colors. Your algorithm needs to decide whether or not any two adjacent regions have the same color.

(b) b. A map has d regions. Your algorithm needs to assign colors to all d regions, drawing colors from a set of m possible colors, in order to guarantee that no two adjacent regions have the same color.

(c) Your algorithm needs to find its way out of a maze drawn on a d-by-d grid.

(d) Your algorithm needs to find the shortest path in a d-by-d maze while hitting m waypoints (equivalent to dots in MP1 part 1.2).

(e) Your algorithm needs to solve a planning problem in a blocks world consisting of d blocks.

Problem 15 (6 points)

The four-queens problem is a constraint satisfaction problem in which one must place four queens on a 4×4 chess board so that none of the queens is on the same row, column, or diagonal with any other queen. One way to formulate the problem is by specifying four variables, A, B, C, and D (one per row), each of which must take a different column from the set $\{1, 2, 3, 4\}$ so that no two (row,column) coordinates are on the same diagonal:

	1	2	3	4
A				
B				
C				
D				

Remember that, in choosing an evaluation sequence for the depth-first search in a constraint satisfaction problem, three heuristics are often useful: LRV (least remaining values), MCV (most constraining variable), and LCV (least constraining value).

(a) According to the LRV, MCV, and LCV heuristics, does it make more sense to choose a value for row A first, or for row B? Why?

(b) Suppose we've decided to assign row B first. According to the LRV, MCV, and LCV heuristics, should we try the value $B = 1$ first, or the value $B = 2$? Why?

(c) Suppose we have assigned $A = 2$, $B = 4$. According to the LRV, MCV, and LCV heuristics, should we try to assign row C next, or row D? Why?

Problem 16 (6 points)

A TBR (two-body robot) is a robot with two bodies. Each of the two bodies can move independently; they're connected by a wi-fi link, but there is no physical link. The position of the first body is (x_1, y_1), the position of the second body is (x_2, y_2).

The robots have been instructed to pick up an iron bar. The bar is 10 meters long. Until the robots pick it up, the iron bar is resting on a pair of tripods, 10 meters apart, at the locations $(1, 0)$ and $(11, 0)$.

(a) Define a notation for the configuration space of a TBR. What is the dimension of the configuration space?

(b) In order to lift the iron bar, the robot must reach an OBJECTIVE where one of its bodies is at position $(1, 0)$ and the other is at position $(11, 0)$. In terms of your notation from part (a), specify the OBJECTIVE as a set of points in configuration space. You may specify the OBJECTIVE as a set of discrete points, or as a set of equalities and inequalities.

(c) If the TBR touches the bar (with either of its bodies) at any location other than the endpoints ($(1, 0)$ and $(11, 0)$), then the bar falls off its tripods. This constitutes a FAILURE. Characterize FAILURE as a set of points in configuration space. You may specify FAILURE as a set of discrete points, or as a set of equalities and inequalities.

Problem 17 (5 points)

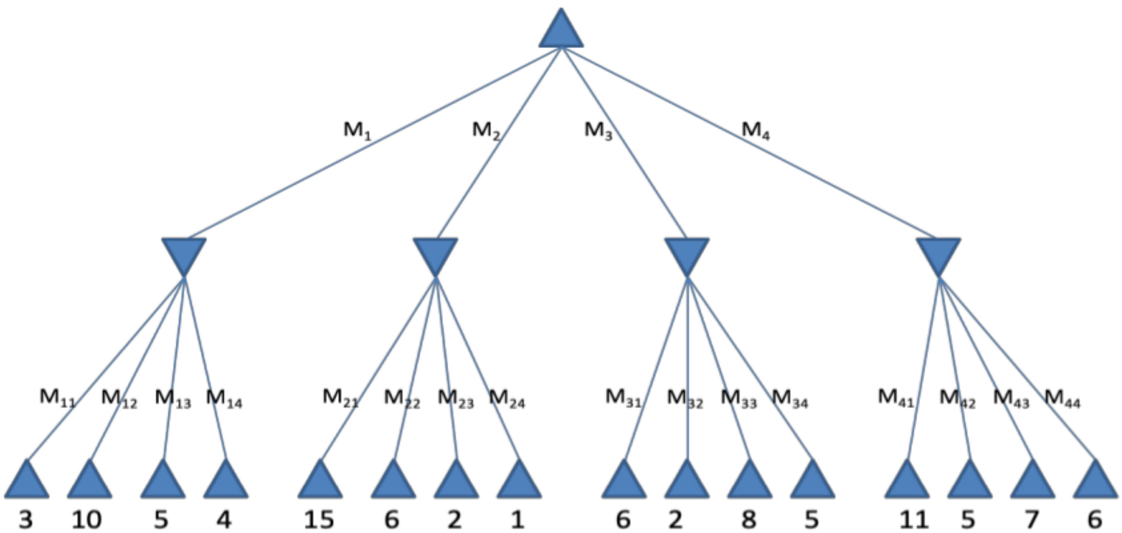
A particular planning problem is defined by a set of three variables (x, y and z) and a set of two possible values (A and B). At any given state of the planning process, each of the three variables may be set to either value, or to the value null ($\emptyset$). There are only two possible actions, called SETX and MOVE:

Action	Preconditions	Results
SETX(val)	$x = \emptyset$	$x =$ val
for val $\in \{A, B\}$		
MOVE(val,var1,var2)	var1 = val, var2 = $\emptyset$	var1 = $\emptyset$, var2 = val
for val $\in \{A, B\}$, var1,var2 $\in \{x, y, z\}$, var1 $\neq$ var2		

In the starting condition, all variables are set to null. In the desired ending condition, $y = A$ and $z = B$. Draw a breadth-first 2-level backward-chaining tree, showing all of the possible n-step predecessors of the desired ending condition for $n \leq 2$.

Problem 18 (12 points)

Consider the following game tree (MAX moves first):



(a) Write down the minimax value of every non-terminal node next to that node.

(b) How will the game proceed, assuming both players play optimally?

(c) Cross out the branches that do not need to be examined by alpha-beta search in order to find the minimax value of the top node, assuming that moves are considered in the non-optimal order shown.

(d) Suppose that a heuristic was available that could re-order the moves of both max (M_1, M_2, M_3, M_4) and min ($M_{11}, \ldots, M_{44}$) in order to force the alpha-beta algorithm to prune as many nodes as possible. Which max move would be considered first: M_1, M_2, M_3, or M_4? Which of the min moves ($M_{11}, \ldots, M_{44}$) would have to be considered?