Midterm 1 Exam

CSCI 561 Spring 2019: Artificial Intelligence

Please print neatly.

Exam Version 1.1 with Solution

Instructions:

1. Date: **2/12/2019 from 7:30 pm - 8:50 pm**
2. Maximum credits/points/percentage for this midterm: 100
3. The percentages for each question are indicated in square brackets [ ].
4. **No books, cell phones, calculators** (or any other material) are allowed.
5. **Write down your name, student ID and USC email address.**
6. **Write answers ONLY in designated areas.**
7. **Do NOT write on the 2D QR code.**
8. **The back of the pages will NOT be graded. You may use it for scratch paper.**
9. No questions during the exam. **If something is unclear to you, write it down on your exam paper.**
10. **Be brief: a few words are often enough if they are precise.**
11. When finished, raise completed exam sheets until approached by a proctor.
12. **Adhere to the Academic Integrity Code.**

Problems	**100 Percent total**
General AI Knowledge	10
Multiple Choice	10
Problem Formulation	16
Intelligent Agents	4
Classical Search	14
Adversarial Search	10
CSP	20
Problem Solving	16

1. [10%] General AI Knowledge

True/False

For each of the statements below, answer **T** if the statement is **always true**, or **F** otherwise.

T	1. An agent does not always perceive its environment through sensors.
F	2. Alpha-beta pruning will always be faster than Minimax without any pruning.
F	3. Simulated Annealing is an optimal algorithm that guarantees convergence to a global minimum (or maximum) solution.
T	4. Genetic algorithms are a variant of stochastic beam search.
T	5. An optimal search algorithm is necessarily complete.
F	6. A Depth-First Search algorithm is guaranteed to find an optimal solution if such a solution exists.
T	7. All consistent heuristics are admissible.
T	8. Every CSP can be written using only binary constraints.
F	9. A* graph search is always optimal when it is implemented using a heuristic that is within 10% of the actual cost.
F	10. Uniform Cost Search is a special case of Depth-First Search.

2. [10%] Multiple Choice

Each question has <u>one or more</u> correct answers. Circle all correct answers.
Please note that there will be no partial credit. You will receive full credit if and only if you choose all of the correct answers and none of the wrong answers.

1. Which statement is false?
   A. BFS is always optimal if the path cost is finite.
   B. BFS is always optimal if the path cost is a decreasing function of the depth of the node.
   C. BFS is always optimal if the state space is infinite.
   D. DFS is always optimal if the state space is finite.
   E. DFS is neither complete and optimal.

2. If MAX uses Minimax, but MIN does not, then a game between them will give MAX a final score that _____________ its minimax value:
   A. will always be higher than
   B. might be higher than
   C. will always be the same as
   D. might be lower than
   E. will always be lower than

3. You need to find the shortest path between two vertices (x_S,y_S) and (x_G,y_G). Every edge $E((x_1,y_1), (x_2,y_2))$ has a cost C such that $D < C < 2D$, where D = Euclidean_Distance(x_1,y_1,x_2,y_2), defined as the Euclidean (straight-line) distance between the edge's endpoints. Which of the following heuristic functions is consistent?
   A. $h(x, y) = 0$
   B. $h(x, y) = -1$
   C. $h(x, y) = ½ \cdot$ Euclidean_Distance(x,y,x_G,y_G)
   D. $h(x, y)$ = Euclidean_Distance(x,y,x_G,y_G)
   E. $h(x, y) = 2 \cdot$ Euclidean_Distance(x,y,x_G,y_G)

4. Select the techniques that can be used to invent admissible heuristics:
   A. Derive from exact solution of a relaxed version of the problem
   B. Derive from exact solution of a harder version of the problem
   C. Learn from experience
   D. Derive from exact solution of subproblems

5. Consider the equation in which each letter represents by a digit 0-9:

EAT + EAT = LIVE

A solution to the above equation consists of a mapping from letters to digits, so that the sum is arithmetically correct (i.e., “LIVE” will be replaced by a four-digit number that is equal to twice the three-digit number that “EAT” is replaced with).

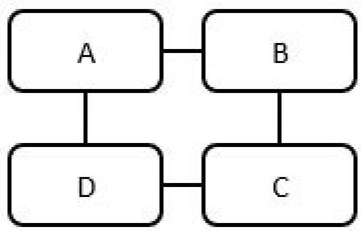
Suppose that: (1) All of the letters (A, E, I, T, and V) are different digits, and (2) the first digit of a number cannot be 0. Which of the following statements are true:

A. There is at least one solution where A = 0 and E = 8
B. There is at least one solution where A = 1 and E = 8
C. There is at least one solution where A = 2 and E = 8
D. There is at least one solution where A = 3 and E = 8
E. There is at least one solution where A = 4 and E = 8
F. There is at least one solution where A = 5 and E = 8

3. [16%] Problem Formulation

In the near future, a planet immigration task is assigned to a group of four intelligent robots. They must all travel from the Earth to Mars by spacecraft. Three restrictions make this task a little intractable:

1. First, there is only one spacecraft, whose engine is too weak to transport more than two robots in a single-trip voyage.
2. Second, only robots *A* and *B* are equipped with a spacecraft navigation program. So at least one of them must be on board the spacecraft for any voyage.
3. Third, no matter whether it is on Earth, Mars, or the spacecraft, each robot must be able to synchronize with any other robots that are in the same place. However, some of the robots have incompatible communication protocols. For example, *A* can only directly synchronize with *B* and *D*. If *A* wants to synchronize with *C*, either *B* or *D* must stay nearby as an interpreter. The links in the following graph show compatibility between robots:



Can you help schedule a sequence of voyages that will get all four robots from Earth to Mars?

1. [3%] Write down the initial and goal states. Use variables (*A, B, C, D*) where *A, B, C, D* ∈ { *E, S, M* } represent the locations of the robots.

Initial state: {A=E, B=E, C=E, D=E} or (E, E, E, E) for robot A, B, C, D on Earth

Goal state: {A=M, B=M, C=M, D=M} or (M, M, M, M) for robot A, B, C, D on Mars.

Note:

The following representation is also acceptable:

Initial state: E = {ABCD}, S = { }, M = { },

Goal state: E = { }, S = { }, M = {ABCD}

2. [3%] For each restriction (1, 2, and 3 in the list above), write down two illegal states that violate it (6 illegal states in total).

(1)

(2)

(3)

Pick 2 from each category

(1):	(2):	(3):
(S, S, S, S)	(*, *, S, S)	(M, *, M, *)
(S, S, S, *)	(*, *, S, *)	(*, M, *, M) For * = E | S
(S, S, *, S)	(*, *, *, S)	(S, *, S, *)
(S, *, S, S)	For * = E | M	(*, S, *, S) For * = E | M
(*, S, S, S)		(E, *, E, *)
For * = E | M		(*, E, *, E) For * = S | M

For example, a correct answer can be:

(1): (S, S, S, E) , (S, S, M, S)
(2): (E, E, S, S), (E, E, E, S)
(3): (S, E, S, M), (E, S, E, M)

Note:
There are 3 restrictions given in this problem.
The 1st infers that it cannot have >2 robots on spacecraft.
The 2nd infers that robots C or D should not be on spacecraft without robot A or B.
The 3rd infers that A and C, as well as B and D, should not stay in a place without the others.

3. [2%] Write down the action(s) you can use to solve this problem.

Move one or two robots from one place to another place.

Note

1) The domain of place is {Earth(E), Spacecraft(S), Mars(M)}
2) Answers with the same meaning are acceptable.

4. [6%] Provide a possible sequence of actions that solve this problem.

Pick one from the following

#1: A, B: E->S B: S->M D: E->S A, D: S->M B: M->S C: E->S B, C: S->M	#3: A, D: E->S D: S->M A: S->E B, C: E->S C: S->M A: E->S A, B: S->M	#5: B, C: E->S C: S->M B: S->E A, D: E->S D: S->M B: E->S A, B: S->M
#2: A, B: E->S A: S->M C: E->S B, C: S->M A: M->S D: E->S A, D: S->M	#4: A, D: E->S D: S->M B: E->S A: S->M C: E->S B, C: S->M	#6: B, C: E->S C: S->M A: E->S B: S->M D: E->S A, D: S->M

Note:
Students may have different representations of the answer.
Graders, please first understand the answer. Also refer to their definition of action in Q3.4

5. [2%] Provide the corresponding sequence of state transitions resulting from your answer to the previous question (Q3.5).

Pick the one which corresponds to the # in Q3.5

#1: (E, E, E, E) (S, S, E, E) (S, M, E, E) (S, M, E, S) (M, M, E, M) (M, S, E, M) (M, S, S, M) (M, M, M, M)	#3: (E, E, E, E) (S, E, E, S) (S, E, E, M) (E, E, E, M) (E, S, S, M) (E, S, M, M) (S, S, M, M) (M, M, M, M)	#5: (E, E, E, E) (E, S, S, E) (E, S, M, E) (E, E, M, E) (S, E, M, S) (S, E, M, M) (S, S, M, M) (M, M, M, M)
#2: (E, E, E, E) (S, S, E, E) (M, S, E, E) (M, S, S, E) (M, M, M, E) (S, M, M, E) (S, M, M, S) (M, M, M, M)	#4: (E, E, E, E) (S, E, E, S) (S, E, E, M) (S, S, E, M) (M, S, E, M) (M, S, S, M) (M, M, M, M)	#6: (E, E, E, E) (E, S, S, E) (E, S, M, E) (S, S, M, E) (S, M, M, E) (S, M, M, S) (M, M, M, M)

4. [4%] Intelligent Agents

1. [2%] Consider a Roomba vacuum that is responsible for cleaning a home. If the cleaning path is programmed by the house owner, is this robot operating in a deterministic or stochastic environment? Why or why not?

Even though the owner programmed the cleaning path, the Roomba is still operating in a stochastic environment. The "why" part should include a few of the following keywords:

In a deterministic environment any action has a single guaranteed effect, and no failure or uncertainty. In a non-deterministic environment, the same task performed twice may produce different results or may even fail completely.

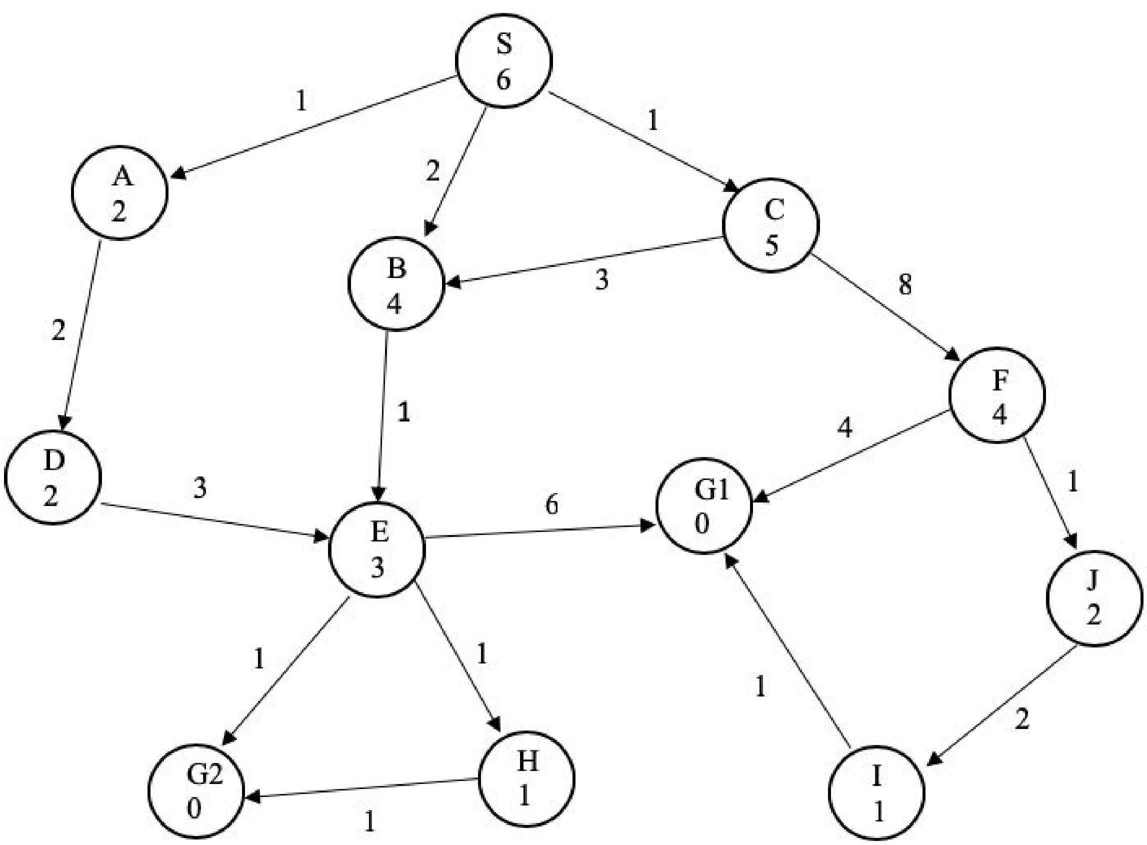
2. [2%] Consider an antivirus software on computer. The software scans a computer for virus, and removes a virus once it's identified. Is the antivirus software episodic? Why or why not?

In an episodic environment, each agent's performance is the result of a series of independent tasks performed. There is no link between the agent's performance and other different scenarios. In other words, the agent decides which action is best to take, it will only consider the task at hand and doesn't have to consider the effect it may have on future tasks.

1. The virus scanner is episodic if the student does not mention that the scanner learns from past filtering, etc.
2. If the student claims that the antivirus agent is not episodic then the answer must include learning or the virus is removed.
3. Note that some students might suggest that the antivirus scanner is episodic because it is heuristic. This alone is not sufficient, it must be heuristic + (learning or virus is removed)

5. [14%] Classical Search

Spring break is coming! Tom and Jerry start to plan their itinerary. They live in city *S*, and they want to visit G_1 or G_2. The map of the country is shown in the graph below, where each node is a city, and the number below the city name is a heuristic estimate of the cost of going from this city to the destinations (G_1 and G_2); Each edge is a path from one city to the other, and the number next to the path is the actual distance between the two cities. Using graph search, for each of the following search algorithms, show the order in which the nodes are expanded starting from *S* and the path found to the goal (or "none" if no path is found). In case of a tie in the frontier, pick the nodes in alphabetical order. G_1 has higher priority than G_2 if the two goal nodes are in a tie. Terminate the search if any goal node is chosen from the frontier for expansion.



1. [2%] Breadth-First Search
Nodes explored:
Path to goal:

Nodes expanded: S A B C D E F G1
Solution: S B E G1

2. [2%] Depth-First Search
Nodes explored:
Path to goal:

Nodes expanded: S A D E G1
Solution: S A D E G1

3.[2%] Uniform Cost search
Nodes explored:
Path to goal:

Nodes expanded: S A C B D E G2
Solution: S B E G2

4. [2%] Greedy Best-First Search
Nodes explored:
Path to goal:

Nodes expanded: S A D E G1
Solution: S A D E G1

5. [2%] A* Search
Nodes explored:
Path to goal:

Nodes expanded: S A D B C E G2
Solution: S B E G2

6. [4%] Search Questions:

a) Is the heuristic for the given graph admissible? Briefly explain why or why not.

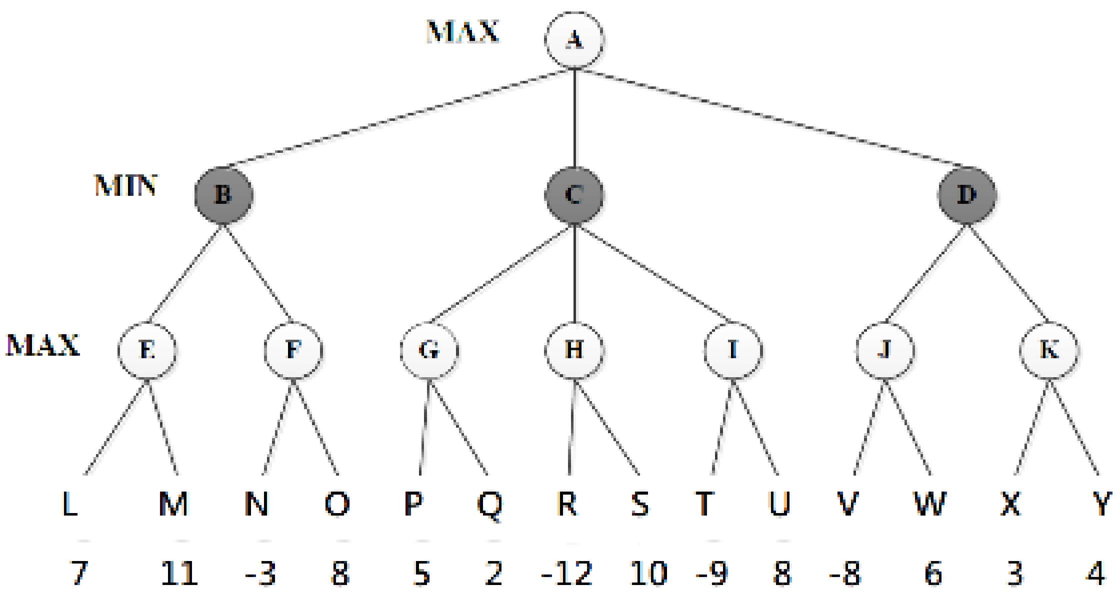
NO. Actual cost for node E is 1, but heuristic cost is 3 which is overestimated

b) Is the heuristic for the given graph consistent? Briefly explain why or why not.

NO. Heuristic cost for node F is 4, heuristic cost for node J is 2, actual cost from F to J is 1. Triangle inequality is not satisfied.

6. [10%] Adversarial Search

Given the following tree, please answer the following questions regarding adversarial search conducted on this tree.



1. [3%] Compute the backed-up values of nodes A to K computed by the minimax algorithm.

A	B	C	D	E	F	G	H	I	J	K
8	8	5	4	11	8	5	10	8	6	4

2. [4%] List all of the nodes which will not be examined when using alpha-beta pruning (pruning from left to right).

HRSITUKXY
(HIK partial score)

3. [3%] In what order should nodes B, C, and D be explored to minimize the number of nodes examined when using alpha-beta pruning (pruning from left to right).

BCD

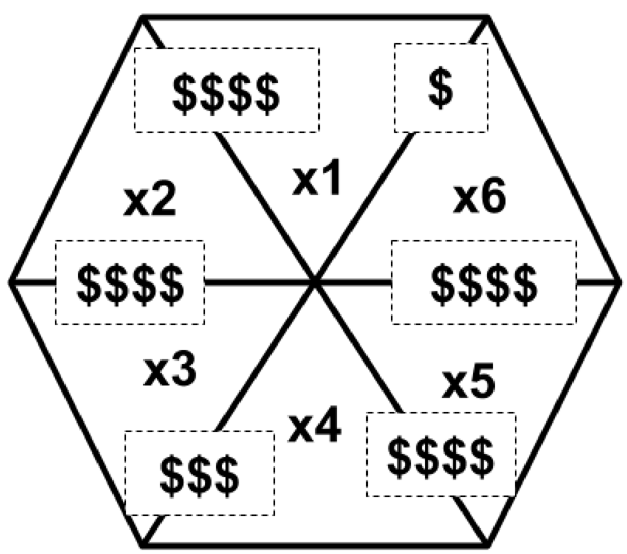
Note: BDC is also correct.

7. [20%] CSP

The Computer Science Department is hosting its annual Lunar New Year Party. Because the department is renowned for its inclusive philosophy, it has decided to invite students from all academic departments. The menu consists of: Toasted Avocado (A), Dry Dates (D), Grilled Broccoli (B), and Kale Tacos (T). Below is a table of relative cost of food.

Food	**Price Label**
Toasted Avocado (A)	$
Dry Dates (D)	$$
Grilled Broccoli (B)	$$$
Kale Tacos (T)	$$$$

Attendants are expected to find their favorite food by solving a brain teaser. So, the department arranged food in a hexagon-shaped tray but kept the food covered. Clues are provided with the food's price label.



To make things more zesty, the price tags are placed in between food compartments, with the price on the tag coming from the more expensive of the two food compartments it is placed between. Because party goers can make only one selection on each trip through the food line, they have to choose wisely.

Your brought a friend who loves Dry Dates (D). Can you help your friend identify the dates?

1. [2%] Write down the variables and their domains in this CSP.

Variables	x1, x2, x3, x4, x5, x6
Domains	{A,B,T,D}

2. [6%] Write down the constraints implied by the prices shown between each food compartment.

Unary	Binary
x1 != B x1 != T x1 != D (or just x1 == A) x3 != T x4 != T x6 != B x6 != T x6 != D (or just x6 == A)	x1 == T or x2 == T x2 == T or x3 == T x3 == B or x4 == B x4 == T or x5 == T x5 == T or x6 == T

3. [6%] You decide to narrow down your domains by enforcing node and arc consistency. What are the remaining domains of each variable after node and arc consistency are enforced?

x1 [A]
x2 [T]
x3 [B D]
x4 [B D]

x5 [T]
x6 [A]

4. [4%] List all solutions to this CSP or state that none exist

x1	x2	x3	x4	x5	x6
A	T	B	D	T	A
A	T	D	B	T	A

5. [2%] If you can pick out only one compartment at a time, how many trips through the line will it take to guarantee you get the dry dates?

2 trips

8. [16%] Problem Solving

Consider a tile puzzle with six tiles (two labeled [1], two labeled [2], and two labeled [3]) in a linear tray which can hold seven tiles. The following depicts the initial state:

[1][1][2][2][3][3][E]

where [1], [2] and [3] are numbered cells, and [E] is an empty cell. The puzzle has the following two legal moves:

(1) A tile may move into an adjacent empty cell with unit cost;
(2) A tile may hop over one or two tiles into the empty cell with a cost equal to the number of tiles hopped over.

Thus, the initial state has the following three immediate successors:

[1][1][2][2][3][E][3] (cost = 1)
[1][1][2][2][E][3][3] (cost = 1)
[1][1][2][E][3][3][2] (cost = 2)

The goal is to have all of the tiles arranged such that it is either less than or equal to the tile to its left. It is unimportant where the empty cell is.

1. [6%] One possible goal state is [3][3][2][2][1][1][E]. Please list the remaining goal states.

[3][3][2][2][1][E][1]
[3][3][2][2[E][1][1]
[3][3][2][E][2][1][1]
[3][3][E][2][2][1][1]
[3][E][3][2][2][1][1]
[E][3][3][2][2][1][1]

2. [2%] Calculate the size of the state space by assuming the tiles of the same values are interchangeable or equivalent.

7! /(2! 2 !2!) = 630.

3. [4%] What is the maximum number of successors that a state can have, by considering all possible positions of the empty cell? What's the minimum? Calculate the branching factor (that is the average number of successors over the entire state space).

Max: six;
Min: three;
Branching factor: (3+4+5+6+5+4+3)/7 or ~ 4.3 or 30/7

4. [3%] Let *h*(*s*) be the following heuristic: *h*(*s*) = the number of tiles that would have to be moved to a different slot (by any number of spaces) for a state "*s*" to become a goal state. For example:

h(22E3113) = 4 because the 3's and 2's must move to make a goal state.
h(33E2121) = 1 because only the rightmost 2 needs to be moved to make a goal state.
h(332E211) = 0, because this is one of the goal states.

What are the *h* values for the immediate successors of the initial state?

h(11223E3) = 4
h(1122E33) = 4
h(112E332) = 5

5. [1%] Is this heuristic admissible?

Yes, because the heuristic is less than or equal to the cost.