Quiz Submissions - ECE457 A Midterm

Zahin Mohammad (username: z5mohamm)

Attempt 1

Written: Jun 25, 2020 4:31 PM - Jun 25, 2020 6:51 PM

Submission View

Released: Jul 10, 2020 4:49 PM

Question 1 **2 / 2 points**

Partially observable environments can emerge as a result of imperfections in the sensors

- ✓ (●) True
- ( ) False

Question 2 **2 / 2 points**

Algorithm A* is a type of best first search

- ✓ (●) True
- ( ) False

Question 3 **2 / 2 points**

The beginning state does not reflect on whether the problem is well-structured or ill-structured.

- ✓ (●) True
- ( ) False

Question 4 **2 / 2 points**

Better solutions are generated when using cooperative transitions in SA which is inspired from GA crossover.

- ✓ (●) True
- ( ) False

Question 5 **2 / 2 points**

Using discrete crossover operators for real-valued GAS limits creating new solutions.

- ✓ (●) True
- ( ) False

Question 6 **2 / 2 points**

SA can be applied for continuous optimization problems.

- ✓ (●) True
- ( ) False

Question 7 **2 / 2 points**

If all moves are unit cost, BFS always finds the optimal path to the goal.

- ✓ (●) True
- ( ) False

Question 8 **2 / 2 points**

1-point crossover is similar to single arithmetic crossover in terms of the changes happens in one gene.

- ( ) True
- ✓ (●) False

Question 9 **0 / 2 points**

GA is a random universal search algorithm.

- ➡ ( ) True
- ✗ (●) False

Question 10 **2 / 2 points**

Algorithm A is a type of Algorithm A*

- ( ) True
- ✓ (●) False

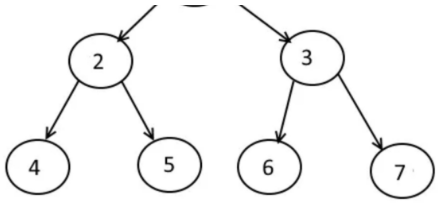
Question 11 **6 / 6 points**

Assume that h1 and h2 are two different heuristics, that both are admissible. Which of the following heuristics are admissible?

- ✓ (●) 0.7*h1+ 0.3*h2
- ( ) h1+h-2
- ( ) h1-h2
- ( ) 0.9*h1+0.5*h2

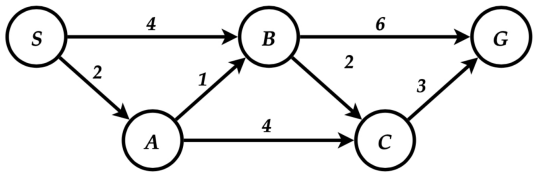
Question 12 **6 / 6 points**

DFS will visit the nodes of the tree below is given by the following order:



- 1-2-3-4-5-6-7
- 1-2-4-3-6-5-7
- ✓ 1-2-4-5-3-6-7

Question 13 **0 / 6 points**



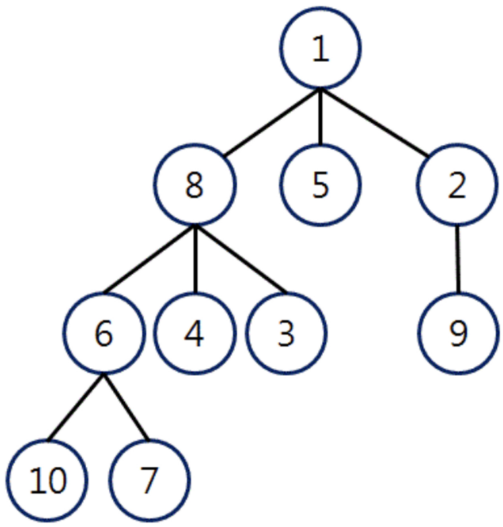
	h_1	h_2	h_3
S	8	7	3
A	3	4	0
B	7	5	4
C	2	2	0
G	0	0	0

For the above search problem, S is the start node and G is the goal node. Three different heuristic functions are defined, h1, h2 and h3. Which statement is true?

- h1 is admissible but inconsistent
- ✗ h3 is admissible and consistent
- ➡ h2 is admissible but inconsistent

Question 14 **6 / 6 points**

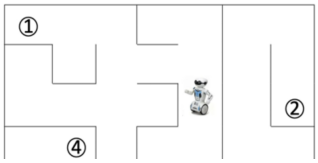
For the shown search tree, Breadth First's Search will visit the nodes of the tree in the following order:



- ✓ 1-8-5-2-6-4-3-9-10-7
- 1-8-5-9-3-4-6-7-10
- 1-8-6-10-7-4-3-5-2-9

Question 15 **0 / 6 points**

Imagine a robot who wishes to collect 4 objects located in an area configured as a maze shown below. The robot is not directional and can move one step in any direction (**n, s, e, w**) at any time step, as long as there is no wall in the way. The objects are immobile. The robot's goal is to find a plan for collecting all objects using as few moves as possible. Assume that the grid has size MxN and there are 4 objects in the maze. What is the size of the state space?



- MxNx2(to the power 4)
- MxNx2(to the power 2)
- MxN

Question 16 **3 / 3 points**

A problem with a very large search space that has a large branching factor and with possibly infinite paths. We have no heuristic. We want to find paths to the goal with minimum number of states. What is the best choice that is memory efficient and guaranteed to find the path with the minimum number of states?

- ✓ Iterative deepening.
- ✓ DFS
- ✓ Uniform Cost Search
- ✓ BFS

Question 17 **3 / 3 points**

No cost and no heuristic information are given. You know that the search space is finite and that you don't have space available to store more than one path at the time. Any solution (if more than one exist) is acceptable. What is the best search algorithm to be used?

- ✓ DFS
- ✓ BFS
- ✓ A*
- ✓ Uniform Cost Search

Question 18 **3 / 3 points**

In Tabu Search, frequency memory is

- ✓ used to encourage the diversification of the search, to force the search in nonexplored regions of the search space.
- used to keep track of recent moves/solutions
- used to keep track of bad moves/solutions to enable search diversification

Question 19 **0 / 3 points**

In Tabu Search, the tabu list is used

- to store the most recent moves/solutions for the entire duration of the search process
- to store the most recent moves/solutions for a predetermined tenure
- to store the most recent moves/solutions

Question 20 **3 / 3 points**

Suppose that you have 4 neighboring solutions (A,B,C,D) in a specific iteration of using tabu search so solve a problem. Which of these solutions will be selected if you decide to encourage more exploration? (value refers to change in cost)

A: value= -2, Tabu=No, freq memory= 12

B: value -3, Tabu=Yes, freq memory= 7

C: Value -2, Tabu=No, freq memory= 23

C: Value -1, Tabu=No, freq memory= 18

- ✓ A
- B
- C
- D

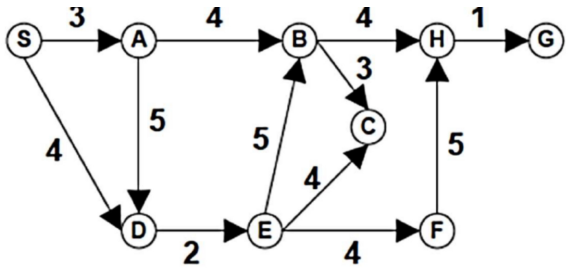
Question 21 **3 / 3 points**

No cost and no heuristic information are given. You need to find the shallowest solution (path from initial state to goal state), but you don't have memory/space available to store more than one path at the time. Select the correct statements about selecting search algorithms to be used.

- ✓ BFS is the best choice.
- ✓ Depth-Limited is the best choice.
- ✓ Iterative Deepening is the best choice.
- ✓ Uniform Cost Search is the best choice.

Question 22 **3 / 3 points**

What is the branching factor in the following tree?



- ○ 1
- ○ 2
- ✓ ◉ 3
- ○ 4

Question 23 **0 / 3 points**

Select the correct statements about best-first search:

- ➡ ✖ ☐ DFS is a special case of best-first search, when the evaluation function f(n) = -depth(n).
- ➡ ✓ ☑ BFS is a special case of best-first search, when the evaluation function f(n) = depth(n).
- ➡ ✓ ☑ Uniform Cost Search is a special case of best-first search, when the evaluation function f(n) = g(n).
- ✓ ☐ Iterative deepening is a special case of best-first search, when the evaluation function f(n) = depth(n).

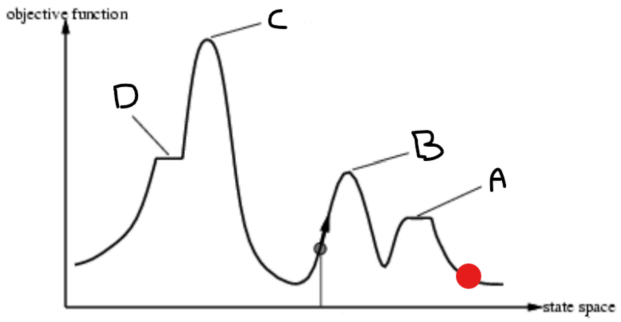
Question 24 **0 / 3 points**

A problem space with a manageable number of states but lots of cycles in the state graph. We have links of varying costs but no heuristic. What is the best choice to find shortest paths?

- ✖ ☑ Uniform Cost Search without ignoring repeated states.
- ✓ ☐ Iterative deepening.
- ✓ ☐ A* with a strict expanded list.
- ➡ ✖ ☐ Uniform Cost Search with a strict expanded list (ignoring repeated states in the closed list).

Question 25 **3 / 3 points**

Which point will the hill climbing (maximizing) algorithm find starting from the red location and given step size of the arrow?



- ✓ ◉ A
- ○ B
- ○ C
- ○ D

Question 26 **0 / 4 points**

What does it mean that a search algorithm is said to be complete?

Answer: A complete algorithm is guaranteed to find a goal node if it exists. ✖ **(A search algorithm is guaranteed to find a solution if there is one.)**

Question 27 **0 / 4 points**

Why the Iterative Deepening Search is complete?

Answer: Iterative deepening search is complete only if branching factor b is finite. This is because it solves the issue with regular DFS which is that it will get stuck if d is infinite or there are loops. Since IDS only goes to a specified depth in each iteration, it is guaranteed to not get stuck in these two cases. ✖ **(It has the potential to search all the branches within the level limit, and it will increase the level limit until it eventually finds a solution.)**

Question 28 **0 / 4 points**

What strategy SA uses to escape local minima/local maxima?

Answer: Simulated annealing allows "bad"moves. The probability of accepting bad moves is related to temperature where SA is supposed to accept more bad moves at higher temperatures.

✖ **(Using a random choice of accepting bad solutions/ non-improving solutions.)**

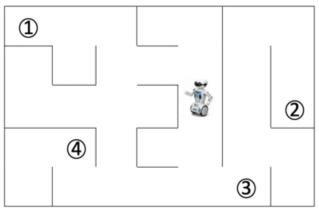
Question 29 **0 / 4 points**

Under what condition, the MiniMax algorithm is complete?

Answer: Minimax is complete only if the search tree is finite. ✖ **(when game tree is finite.)**

Question 30 **0 / 4 points**

Imagine a robot who wishes to collect 4 objects located in an area configured as a maze shown below. The robot is not directional and can move one step in any direction (**n, s, e, w**) at any time step, as long as there is no wall in the way. The objects are immobile. The robot's goal is to find a plan for collecting all objects using as few moves as possible. Assume that the grid has size MxN and there are 4 objects in the maze. Give a suitable representation of the states in this searching problem.



Answer: The state can be represented as a MxN array, where empty spaces are represented by 0's, the objects are represented as their numbers (as seen in the image), the walls can be represented as an X and the robot can be represented with an R. The path of the robot can be represented as a 1 dimensional array indicating the path it has taken to get to the current point. Combined these two array's form the current state.

✖ **({m,n,o1,o2,o3,o4}, where m in [1,M], n in [1,N] (representing the position of the target), and g1,g2,g3,g4 in [true,fales] (telling whether an object has been picked up by the robot.))**

Attempt Score:57 / 100 - 57 %

Overall Grade (highest attempt):88.5 / 100 - 88.5 %

Done