Minor Exam: Sunday, 19th September, 2021

The exam is intended for both COL333 and COL671 students.

Please write answers legibly using dark blue/black ink.

Please write your name, entry number and page number on all sheets. The response for each question must begin on a fresh page and that all sub-parts of a question must appear together.

Responses are to be handwritten, scanned as a single PDF file and uploaded as per submission instructions. The space requirement for any questions is typically under a single page. Please do not exceed two pages for any question.

Clarifications will not be provided during the exam. The necessary information is provided for each question.

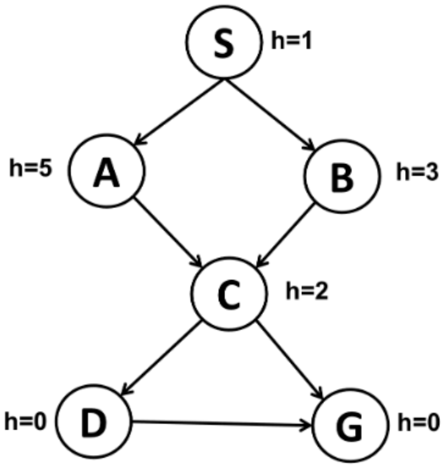
Please work individually and submit responses based only on your own efforts.

Wishing you good luck for the exam.

Name and Entry No.: ____________________

Question	Points	Score
1	12	
2	12	
3	8	
4	8	
5	10	
6	10	
7	10	
Total:	70	

1. (12 points) Consider the search problem below with the start state S and the goal state G. The heuristic values are shown for each node in the graph. Unfortunately, we do not know the transition costs for any of the edges in the graph. We know that the given heuristic is *admissible*. Furthermore, we know what was the *priority queue* of A* (graph-search version) after each node expansion. The state of the priority queue is shown below as a list of nodes and their priority as determined by the f- value. The priority increases from left to right in the queue.



Iteration	State of Priority Queue
1.	{ (S, f=1) }
2.	{ (A, f=5), (B, f=6) }
3.	{ (A, f=6), (C, f=7) }
4.	{ (C, f=6) }
5.	{ (D, f=5), (G, f=7) }
6.	{ G, (f=6) }

(a) Determine the transition costs for *all* the edges in the graph from the information above. Please draw the graph and show the transition cost for each edge in the graph. Show working to arrive at your answer.

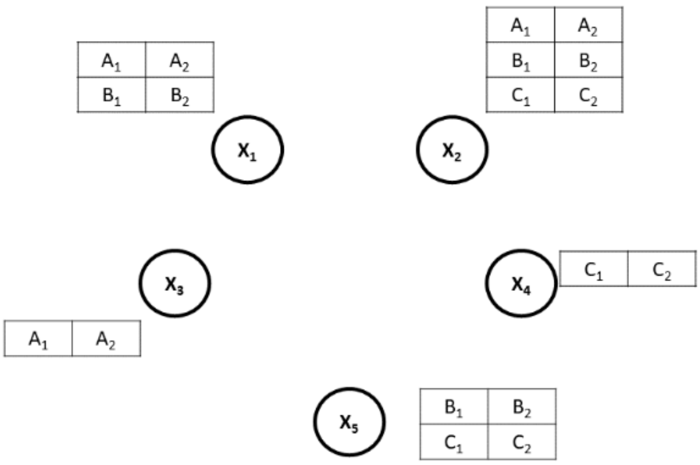
(b) Consider the following claim regarding the general setting for the example above: *If we are given a search problem for which we know: (i) that the heuristic is admissible, (ii) the heurstic values for all the nodes, (iii) the state of the priority queue (including f-values) then the transition costs for all the edges in the search problem can be uniquely determined.* Is this claim correct? Yes or No. Explain briefly in 2-3 lines.

Name and Entry No.: ______________________________

2. (12 points) Consider the following allocation problem. We have three robots (A, B, C) that must complete five tasks (1, 2, 3, 4, 5) in a total time duration of two hours. Each task takes one hour to complete. Each robot can work on only one task at a time and only one robot may work on a task at a time. Further it is required that task 1 must be completed before task 2, and task 3 must be completed before task 5. Note that each robot can perform only specific tasks. Robot A can perform tasks 1, 2 and 3. Robot B can perform tasks 1, 2 and 5. Robot C can perform tasks 2, 4 and 5.

   We formulate this problem as a CSP as follows. A variable is instantiated for each task as X_1, X_2, X_3, X_4 and X_5. Each variable can potentially take the following values in its domain $\{A_1, A_2, B_1, B_2, C_1, C_2\}$. In this formulation, assigning the variable X_5 a value of C_2 implies that robot C will complete task 5 in the second hour. Also, let $Time(X_5)$ denote the hour when the task X_5 is scheduled (which is 2 in the above example).

   The following diagram shows the incompletely-drawn constraint graph where the variables and certain permitted domain values are shown but the constraints are missing.



   (a) Please identify and write the *constraints* for this problem. Add the constraints in the diagram above and draw the complete constraint graph in your answer.

   (b) Enforce *arc consistency* in the *entire* CSP by running AC-3 on the constraint graph obtained in part (a). For each variable which values will be removed (if any)? Draw the resulting constraint graph by crossing out the values for each variable that will be removed.

   (c) Consider the CSP obtained after part (b). Solve the CSP using *backtracking search* (without forward checking). Use the *most constrained variable* heuristic for ordering variables and break ties among variables using numerical order. Use the *least constrained value* heuristic for picking values and break ties among values according to their alpha-numeric value. Identify the first three variables and their assigned values during backtracking search. Fill the following table and show working (1-2 lines each) to arrive at your answer.

Order	Variable	Assigned Value
1.		
2.		
3.		

3. (8 points) Please find below an implementation of A* (graph search) which may be incorrect. Here, the *fringe* is a priority queue and nodes are inserted into the fringe using the standard key for A*, namely $f = g + h$. The heuristic h can be assumed to be consistent.

```
function A*-SEARCH(problem, fringe)
    closed ← an empty set
    fringe ← INSERT(MAKE-NODE(INITIAL-STATE[problem]), fringe)
    loop
        if fringe is empty then
            return failure
        end if
        node ← REMOVE-FRONT(fringe)
        if STATE[node] IS NOT IN closed then
            ADD STATE[node] TO closed
            for successor IN GETSUCCESSORS(problem, STATE[node]) do
                fringe ← INSERT(MAKE-NODE(successor), fringe)
                if GOAL-TEST(problem,successor) then
                    return successor
                end if
            end for
        end if
    end loop
end function
```

Consider each of the following assertions about the algorithm implemented as above. Write if each of the assertions is True or False. Provide a brief explanation (1-3 lines each).

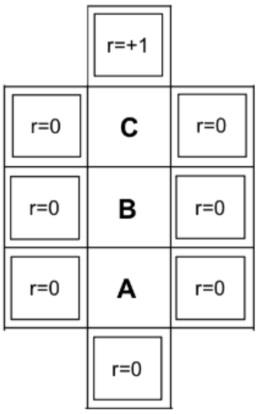
(a) The algorithm may expand nodes more than once.

(b) The algorithm can return a non-optimal path. Here, non-optimality relates to returning a higher cost path than the lowest cost path.

(c) In this algorithm, the fringe will grow to a maximum size of half the size of the state space.

(d) The algorithm is complete.

5. (10 points) Consider a grid-world MDP shown in the following figure. There are two types of states. The states marked with a double rectangle are *exit* states where the only action available is *Exit*. The *Exit* action taken in an *exit* state and moves the agent to a terminal state X (not shown in the figure). The states labeled as A, B and C are the *non-exit* states where the agent can choose either the *GoUp* or the *GoDown* action. The actions *GoUp* and *GoDown* are stochastic and succeed in reaching the intended state with a probability of 0.5. With a probability of 0.25 each the agent can land up in the grid cell to the left or to the right. When the *Exit* action taken in any of the *exit* states, then the agent receives immediate reward or $r = 0$ or $r = +1$ shown within the grid cells for the *exit* states. The reward for all other transitions is assumed to be zero. Assume a discount factor $\gamma = 1$. The agent's starting state is A.



(a) Consider value iteration algorithm for this MDP. The values for each state are initialized as $V_0(s) = 0$ for all states. Perform value iteration till the value function for the start state becomes *non-zero*. At each iteration, draw the entire grid by writing the value of each state in the corresponding grid cell. Show working to arrive at the answer (only for states updated with non-zero values in any iteration).

(b) Determine the optimal value function for this MDP. What are the values for $V^*(A)$, $V^*(B)$ and $V^*(C)$?

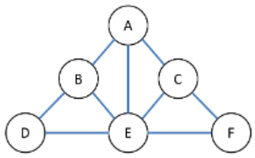
(c) How many policies are possible for this MDP? Draw the optimal policy.

Name and Entry No.: ______________________________

6. (10 points) Please write the correct option(s) for the questions below. Note that more than one option may be correct. Write the text for the option(s) you select or write that no option is correct. A correct answer will receive 2 point each and no points otherwise. No negative marks in this question.

   (a) Which of the following AI systems showed language understanding in a simulated computer world?
   - ◯ Alpha Go
   - ◯ SHRLDU
   - ◯ DENDRAL
   - ◯ General Problem Solver

   (b) Which of the following node can be considered as a cut set for the following constraint graph?



   - ◯ A
   - ◯ B
   - ◯ F
   - ◯ E

   (c) Given two admissible heuristics h_A and h_B then which of the following heuristics are admissible?
   - ◯ $(h_A + h_B)$
   - ◯ $max(h_A, h_B)$
   - ◯ $min(h_A, h_B)$
   - ◯ $(h_A * h_B)$

   (d) In the context of adversarial search which of the following are true for (α, β)-pruning?
   - ◯ effectively halves the depth of the search
   - ◯ effectively doubles the depth of the search
   - ◯ may lead to a different decision at the root node compared to minimax
   - ◯ leads to the same decision at the root node compared to minimax

   (e) A single iteration of Value Iteration for one state in an MDP has the worst case time complexity of:
   - ◯ $O(|States||Actions|)$
   - ◯ $O(|States|^2|Actions|)$
   - ◯ $O(|States|^2|Actions|^2)$
   - ◯ $O(|States||Actions|^2)$

7. (10 points) Please write if the following statements are True or False. A correct answer will receive (+1.0) points each, an incorrect answer will receive a negative (−0.25) point and no points for leaving blank.

(a) In an episodic task, the future actions are independent of actions taken previously.

○ True
○ False

(b) When enforcing arc-consistency in a CSP, the set of values for variables which remain when the algorithm terminates does not depend on the order in which arcs are processed from the queue.

○ True
○ False

(c) Using hill-climbing search requires that you have a formula for the gradient of the function you are trying to optimize.

○ True
○ False

(d) In an MDP, the larger the discount factor γ, the more strongly favoured are the short term rewards over long-term rewards.

○ True
○ False

(e) Forward checking propagates information between unassigned variables.

○ True
○ False

(f) If a binary CSP has a tree-structured constraint graph, then we can find a satisfying assignment (if it exists) in time that is linear in the number of variables.

○ True
○ False

(g) IDA* uses the f-value to limit the search.

○ True
○ False

(h) The cost of the optimal solution in a relaxed search problem is an admissible heuristic for the original problem.

○ True
○ False

(i) In an MDP with a finite number of states and a finite number of actions, the number of possible policies is finite.

○ True
○ False