Student Name:

Student Number:

2019 CAB320 Mid Semester Exam Instructions

Do NOT remove the staple from this booklet. Write your name and student number at the top of the first page. Attempt all the questions. All calculators are allowed. This is individual work. **No notes allowed**. If you have any question, call the invigilator. Marks for each question as indicated. **Write your answers in the boxes of the corresponding questions (or on the dots standing for missing code)**. Good luck!

Question 1

Two mobile robots *R1* and *R2* located in an *NxN* grid maze have to meet. It does not matter in which cell of the maze they meet. In each time step, *R1* and *R2* simultaneously take an action; either STOP or move in one of the following directions: {NORTH, SOUTH, EAST, WEST}. You must devise a plan which positions them together, somewhere in the grid, in as few time steps as possible. Passing each other does not count as meeting; they must occupy the same cell at the same time.

(1.a) Formally state this problem as a single-agent state-space search problem.

States: express the state space in set comprehension form

Size of the state space:

Branching factor of the search tree:

Goal test function:

(4 marks)

(1.b) Give a non-trivial (different from zero) admissible heuristic for this problem. Assume that each time step costs *1*. That is, the total cost of moving the two robots has a value of *1* even if one of the robot is not moving.

(3 marks)

(1.c) Mark with a tick all of the following **tree search** methods which are guaranteed to output optimal solutions to this problem:

(i) DFS

(ii) BFS

(iii) UCS

(iv) A* with an admissible heuristic

(v) A* with heuristic that returns zero for each state

(vi) Greedy search with an admissible heuristic

(6 marks)

(1.d) Assuming that *h1* and *h2* are admissible, mark with a tick all of the following expressions which are also guaranteed to be admissible:

(i) *h1* + *h2*

(ii) *h1* ∗ *h2*

(iii) max(*h1*, *0.3*h2*)

(iv) min(*h1* , *3*h2*)

(v) *0.94*h1* + *0.08*h2*

(5 marks)

Question 2

You are designing a menu for a special event. There are several choices, each represented as a variable: (**A**)ppetizer, (**B**)everage, main (**C**)ourse, and (**D**)essert. The domains of the variables are as follows:

A: (**v**)eggies, (**e**)scargot

B: (**w**)ater, (**s**)oda, (**m**)ilk

C: (**f**)ish, (**b**)eef, (p)asta

D: (**a**)pple pie, (**i**)ce cream, c(**h**)eese

The menu has to obey the following dietary constraints:

(i) Vegetarian options: The appetizer must be veggies or the main course must be either pasta or fish.

(ii) Total budget: If you serve the escargot, you cannot afford any beverage other than water.

(iii) Calcium requirement: You must serve at least one of milk, ice cream, or cheese.

(2.a) Draw the constraint graph over the variables A, B, C and D.

(2 marks)

(2.b) Assume we first assign *A*=*e*. List the domains of the variables after forward checking.

A in { e }

B in { }

C in { }

D in { }

(3 marks)

(2.c) Again assume we first assign *A*=*e*. List the domains of the variables after arc consistency has been enforced.

A in { e }

B in { }

C in { }

D in { }

(3 marks)

(2.d) Give a solution for this CSP or state that none exists.

(1 mark)

(2.e) What is the running-time of an efficient solver for tree-structured constraint satisfaction problems given that *n* is the number of variables and *d* is the maximum size of any variable's domain?

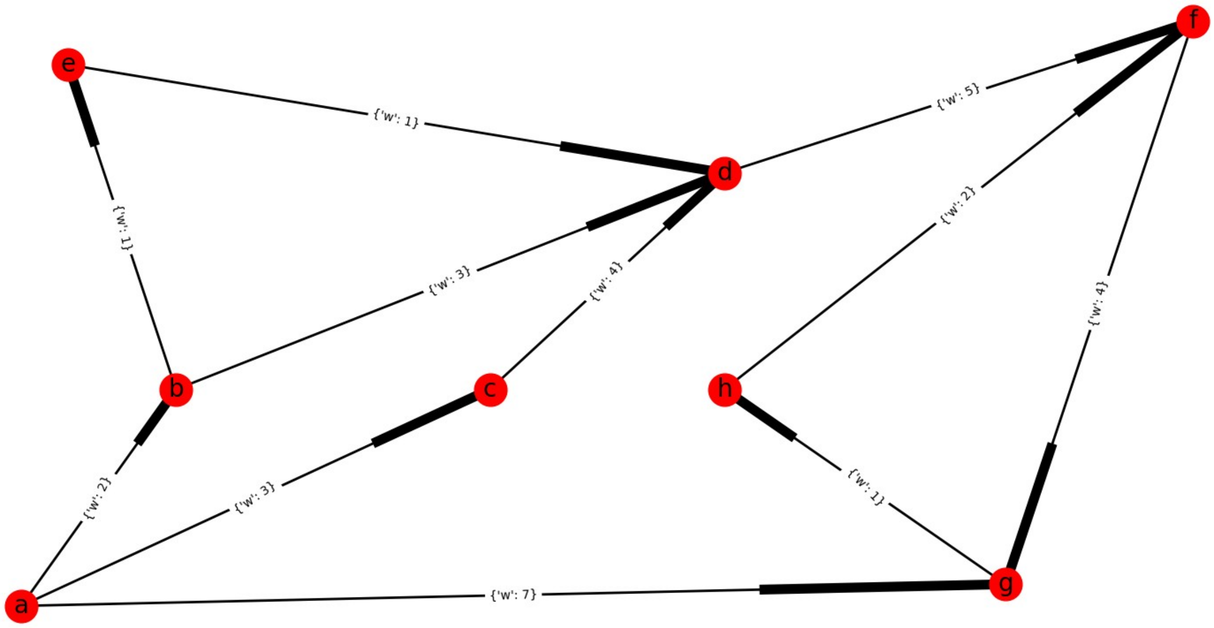
(3 marks)

(2.f) Name and briefly describe a standard technique for turning nearly tree-structured CSPs into tree-structured ones.

(3 marks)

Question 3

Consider the state graph below whose vertices represent states and whose arcs represent transitions between states. The head of the arc is indicated by the wider segment of the edge. For example, the state 'a' connects to state 'c with an arc directed from 'a' to 'c'. The cost of using this arc is the weight 'w=3'. We assume that the **successors of a state** are **generated in alphabetical order**.



(3.a) How many nodes are in the **complete search tree** for the graph above when the initial state is **A**? Note that no goal state has been specified yet.

(2 marks)

(3.b) Recall that a subgraph induced by a subset *S* of vertices of a graph *G* is the graph obtained from *G* by removing all vertices not in *S* and their incident arcs. How many nodes are in the **complete search tree** for the subgraph induced by the subset of vertices {A, B, C, D, E} when the initial state is **A**?

(2 marks)

(3.c) Consider a depth-first search on the state graph displayed at the beginning of Question 3. Assume that the initial state is **A**, and that **D** and **F** are goal states. Enter in the box below the final path returned by a **depth-first tree search**. Your answer should be 'no solution' or a string with 'A' as the first character and the last character should denote one of the goal states.

(4 marks)

(3.d) Consider a breadth-first search on the state graph displayed at the beginning of Question 3. Assume that the initial state is **F**, and that **E** and **D** are goal states. Enter in the box below the final path returned by a **breadth-first tree search**. Your answer should be 'no solution' or a string with 'F' as the first character and the last character should denote one of the goal states.

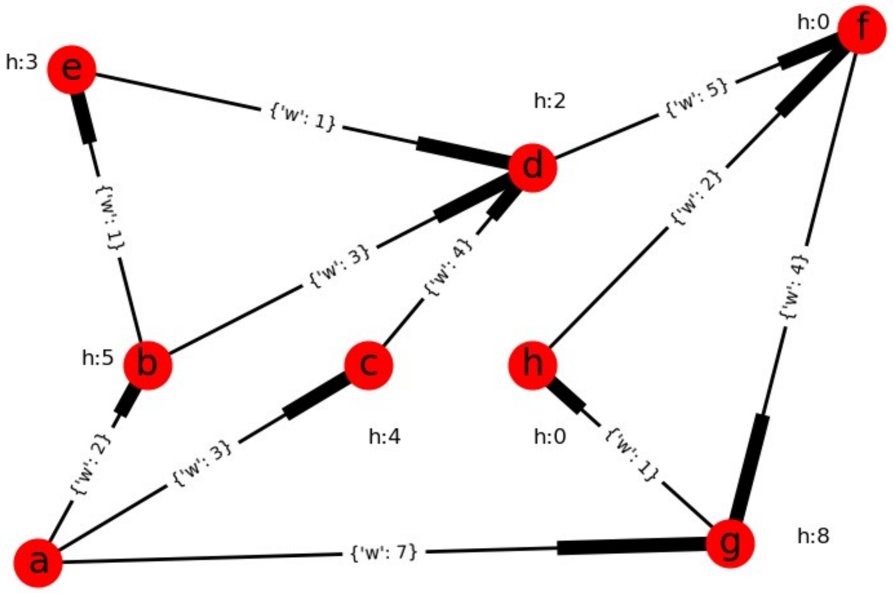
(4 marks)

(3.e) Consider a uniform cost search on the state graph displayed at the beginning of Question 3. Assume that the initial state is **A**, and that **F** is the goal state. Enter in the box below the final path returned by a **uniform cost tree search**.

(5 marks)

Question 4

Consider the search space below, where **A** is the start state and **F** and **H** are the only states that satisfy the goal test. Arcs are labeled with the cost of traversing them and the estimated cost *h* (heuristic) to a closest goal is indicated next to the vertices.



For each of the following **tree search** strategies, indicate which goal state is reached (if any) and list, in order, all the states popped out of the frontier. Assume that the **successors of a state are generated in alphabetical order**. In case of a tie, pick the element that comes first in the alphabetical order.

Iterative deepening depth first search

Goal state reached:

States popped out of the frontier:

A* tree search

Goal state reached:

States popped off of the frontier:

(4+5 marks)

Question 5

We write $h^{*}(s)$ for the true cost from a state s to the closest goal state of a state graph *G1*. We write *g1(n)* for the cost of a path from the root node to node *n* in the search tree associated with *G1*. Let *h1* be an admissible heuristic, and consider *h2 = 7*h1*. Consider the state graph *G2* derived from *G1* by multiplying the weight of each arc by 7.

(5.a) Do the optimal paths in *G2* correspond to the optimal paths in *G1*? Justify your answer. Provide a counter-example if needed.

(2 marks)

(5.b) Is *h2* an **admissible** heuristic for *G2*? Justify your answer.

(3 marks)

(5.c) What can you say about the cost of the returned path when running A* tree search on *G1* with *h2* as the heuristic? Relate the cost of the returned path to the cost of an optimal path.

(4 marks)

End of Paper

Scribble Pages [will be ignored by the marker]