National University of Singapore
School of Computing
CS3243 Introduction to AI

Tutorial 6: Mid-term Quiz Review

Issued: June 3, 2020 Discussion in: Week 4, Thursday Session

1. State if each of the following statements is True or False. Provide a rationale for your answer.
   (a) It is possible for an agent to be perfectly rational in k distinct task environments.
   (b) There exists a deterministic task environment where an agent that selects its actions uniformly at random (from a discrete set of possible actions), is rational.
   (c) Every agent is rational in an unobservable environment.
   (d) The input to an agent program is the same as the input to the agent function.

2. State if each of the following statements is True or False. Provide a rationale for your answer.
   (a) Assuming branching factor m and the shallowest solution is at depth m, there exists a case where Iterative Deepening Search produces a solution by expanding strictly fewer nodes than Breadth-first Search.
   (b) Depth-first Search is a special case of Best-first Tree Search.

3. A particular word game begins with a random allocation of 6 letters from the set of vowels (i.e., $V = A, E, I, O, U$), and 10 letters from the set of consonants (i.e., $A \backslash V$, where A is the set of the English Alphabet). In total, 16 letters are chosen and randomly placed within a 4×4 grid. All 16 letters are drawn with replacement. An example grid is shown below.

T	A	C	L
E	Z	U	L
N	O	L	O
T	O	L	D

The objective of the game is to form several words such that the sum of their scores is at least some given positive integer value k, and such that you use as few words as possible. For any particular word, w, the function $\texttt{CHECK}(w)$ returns a non-negative value. When w corresponds to an illegal word, a score of 0 is returned, while any legal word is allocated a positive integer score.

Words may only be formed:

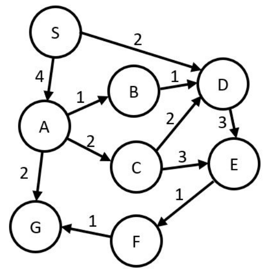
- left to right (e.g., TOLD from the bottom row in example grid above)
- right to left (e.g., CAT from the top row in the example grid above)
- top to bottom (e.g., TENT from the leftmost column in the example grid above)
- bottom to top (e.g., DOLL from the rightmost column in the example grid above)

Further, each letter may only be used in **one** word. For example, if you used the word CAT in your solution, then you may not use the word TENT since they both share a common T.

Specify the search space for the above game such that we may apply the (tree-based variant of the) Uniform Cost Search algorithm to find an optimal solution. You must assume that you will be using an unmodified version of this algorithm.

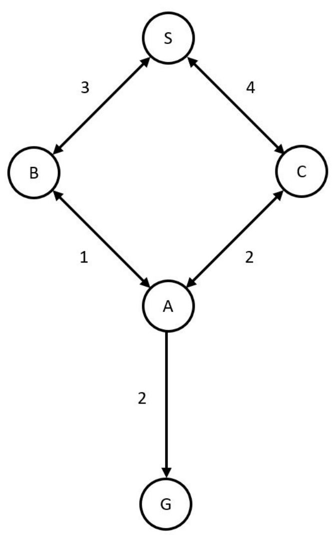
You may use plain English to describe your answer. However, as far as possible, you should formalise your answer. You must ensure that it is specific and well-defined.

4. Assuming that ties are broken based on alphabetical order, specify the order of nodes that would be explored by the following algorithms. Note that you **MUST** express your answer in the form S-A-B-G (no spaces, all uppercase letters, delimited by the dash (-) character), which, for example, corresponds to the exploration order: S, A, B, then G. You should assume tree-based implementations, and that S is the initial node, while G is the goal node.



   (a) Breadth-first Search
   (b) Uniform-cost Search
   (c) Depth-first Search

5. Consider the directed graph depicted in the Figure below. Let S be the initial state and G be the goal state. The cost of each action is as indicated.



Trace the application of the tree-based variant of the Depth-limited Search algorithm, with maximum depth $= 3$, on this graph. Assume that the depth of the start node is 0 and that ties are resolved based on alphabetical order (i.e., A before B, etc.).

6. State if each of the following statements is True or False. If the statement is true, prove it. Otherwise, provide a counterexample.

   (a) Given that a King in the game of International Chess may move one square in any of the eight possible directions (refer to Figure below for a graphical representation - note that in the Figure, K represents the King piece), the Manhattan Distance heuristic would be admissible for the problem of moving the King from an initial square, s, to a goal square, t.

$\nwarrow$	$\uparrow$	$\nearrow$
$\leftarrow$	K	$\rightarrow$
$\swarrow$	$\downarrow$	$\searrow$

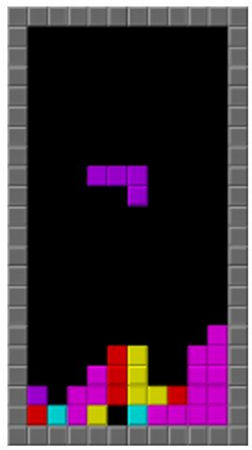
   (b) If $h_1(s)$ and $h_2(s)$ are two admissible A* heuristics, let $h_3(s) = w.h_1(s)+(1-w).h_2(s)$ for any real number w in [0, 1]. Then $h_3(s)$ must also be admissible.

   (c) If $h_1(s)$ and $h_2(s)$ are both consistent heuristics, then the heuristics $h_3(s) = min(h_1(s), h_2(s))$ and $h_4(s) = max(h_1(s), h_2(s))$ cannot both be consistent.

   (d) Given an admissible heuristic, h_1, and an inadmissible heuristic, h_2, a third heuristic, $h_3 = h_1 \times h_2$, is always inadmissible.

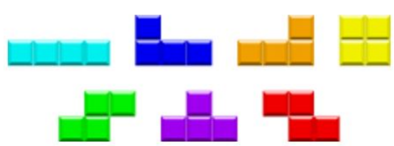
   (e) Given an admissible heuristic h, the Greedy Best-first Search algorithm may never outperform A* Search when both utilise h.

7. Tetris (fill-the-board variant) is a tile-matching game in which pieces of different geometric forms, called *tetriminos*, descend from the top of the field. During this descent, the player can move the pieces laterally (move left, move right) and rotate (rotate right, rotate left) them until they touch the bottom of the field or land on a piece that had been placed before it. The player can neither slow down the falling pieces nor stop them.



The objective of the game is to configure the pieces to fill a board completely without any surrounded gaps. A gap is defined as an empty cell on the board. A row (or column) is complete if there are no gaps in that row (or column respectively). A gap is called a blocked gap if for the corresponding column where the gap belongs to, there exists an occupied cell somewhere above that gap.

There are 7 kinds of tetriminos. Assume that we start with a fixed number of tetriminos, N (comprising some of each kind), and all are required to be used to fill the board (i.e. there exists a way to place all these tetriminos such that the board is filled).



In this problem, the **states** are different partially filled tetris fields with a tetrimino that is about to be placed in the field next (but not placed yet); the **initial state** is an empty field with a starting tetrimino; an **action** is a sequence of lateral movement(s) and/or rotation(s) of a tetrimino (assume the player has enough time to shift the tetrimino to an intended configuration before descent); the **transition model** takes in a state, applies the sequence of actions on the tetrimino that enters the field, and outputs a state where the tetrimino of that specified configuration descended onto the field; the **goal state** is a completely filled board where there are no gaps (and every tetrimino fits perfectly); the **transition cost** is 1. You may assume there exists such a goal state.

(a) Select all of the heuristics that are admissible. If you feel that none are admissible, select only the option "None of the options are admissible". For each option, briefly, but clearly explain, why it is admissible/inadmissible.

- $h_1(n)$ = number of unfielded tetriminos
- $h_2(n)$ = number of gaps
- $h_3(n)$ = number of incomplete rows
- $h_4(n)$ = number of blocked gaps
- None of the options are admissible.

(b) With reference to the heuristics in the first question of this section, select all of the following that are True. Upon making your choice, pick any **one** of the options, clearly indicate which option you picked and justify your answer for that option.

- $max(h_1, h_2)$ is admissible
- $min(h_2, h_3)$ is admissible
- $max(h_3, h_4)$ is inadmissible
- $min(h_1, h_4)$ is admissible

(c) With reference to the heuristics in the first question of this section, select all of the following that are True. Upon making your choice, pick any **one** of the options, clearly indicate which option you picked and justify your answer for that option.

- h_1 dominates h_2
- h_2 dominates h_4
- h_3 does not dominate h_2
- h_4 does not dominate $h_2/2$

8. Let G be a graph defined by:
   - Vertices $V = \{s, v_1, v_2, t\}$
   - Edges $E = \{(s, v_1, 2), (s, v_2, 4), (s, t, 10), (v_1, t, 6), (v_2, t, 3)\}$

   Define **ALL** possible admissible heuristics for G, with source s and goal t.

9. State if each of the following statements is True or False. If True, prove it. Otherwise, provide a counterexample.
   (a) For all game trees, the payoff to MAX obtained by using MINIMAX decisions against a MIN player that does not use an optimal MIN-strategy will never be lower than the payoff to MAX obtained by playing against an optimal MIN.
   (b) An optimal MINIMAX-strategy employed by MAX (rather than a suboptimal one) always results in a higher payoff when playing against a MIN player that does not use an optimal MIN-strategy.

10. The following questions refer to the adversarial search tree described below (explicit reference to the tree, if required, will be made).

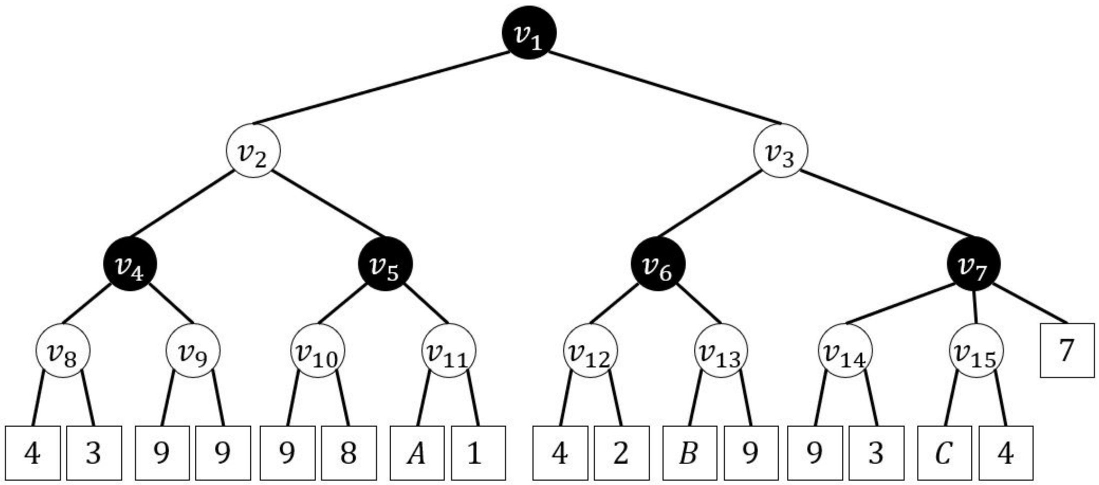
    The black nodes are controlled by the MAX player, and the white nodes are controlled by the MIN player. The square nodes are terminal nodes, with the values within corresponding to the payoff to the MAX player.

    The value of a node is the payoff to the MAX player as computed by the `MINIMAX` algorithm. When running the `MINIMAX` algorithm, we evaluate nodes from left to right.

    (a) Run the α-β pruning algorithm on the above tree. Determine the values for A, B, and C in the terminal nodes in order to ensure that a **minimal number of sub-trees are pruned**. You may assume A, B and C are positive integers.

    $A >$ _____
    $B >$ _____
    $C >$ _____

    Provide rationales to justify each of your answers.



(b) Refer to the above tree. Suppose we fix the values of A, B, and C to be the minimum positive integer that satisfies the inequalities you have specified in the first part of this question.

What is the minimax value at the root (i.e., at v_1)?

(c) Refer to the above tree. Suppose we fix the values of A, B, and C to be the minimum positive integer that satisfies the inequalities you have specified in the first part of this question.

What is the minimax value at node v_6?

(d) Refer to the above tree. Suppose we fix the values of A, B, and C to be the minimum positive integer that satisfies the inequalities you have specified in the first part of this question.

Run the α-β pruning algorithm on the above tree.

How many terminal nodes are not evaluated due to pruning?

Provide a ratioanle for your answer. If there are nodes not evaluated due to pruning, explain why they are pruned. If your answer is 0, then explain why the terminal node of value 7 connected to node v_7 is not pruned.