United International **University (UIU)**

Dept. of Computer Science & Engineering (CSE)

Mid Exam: Spring 2020

Course Code: CSI 341, Course Title: Artificial Intelligence

Total Marks: 30 Duration: 1 hour 45 minutes

Answer all questions. Marks are indicated in the right side of each question.

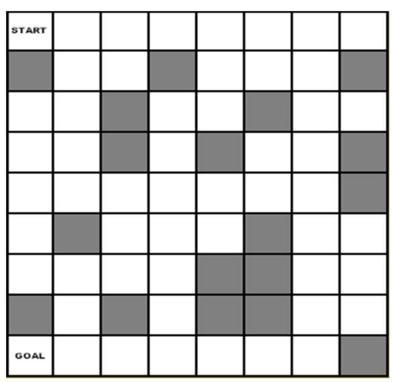
---

1. a) Suppose you are going to design an intelligent agent for cleaning water surface of river. The objective is to collect the floating waste. To do that the robot first detects the waste using its camera, determines the distance through the depth sensors, moves close to the waste, and finally collects the waste to the collector bin. The robot can detect waste within 60 meters. The scenario of floating waste in the water varies from time to time.

   Now provide the **PEAS** description of this water surface cleaner robot and also detect whether the agent's environment is *Fully or Partially observable, Single or Multi-agent, Deterministic or Stochastic, Episodic or Sequential*. **[1.5+2]**

   b) Consider the following *Maze Problem* where only the allowed moves are *{Left, Right, Up, Down}*. You can't move to the colored cells and at a time you can move exactly 1 step. Your goal is to reach to the state "GOAL" starting from the state "START". **[1.5]**

   Now give a formal description of this search problem.



2. a) Suppose the search tree corresponding to your search problem has branching factor, $b = 8$ and the optimal solution depth, $d = 4$ and the maximum depth, $m = 100$.

   Now calculate the total number of nodes that will be generated during Iterative Deepening Depth-First Search. **[1]**

   b) "Iterative Deepening DFS is not optimal when all the step costs are not identical" – draw a search tree that supports this statement. **[1]**

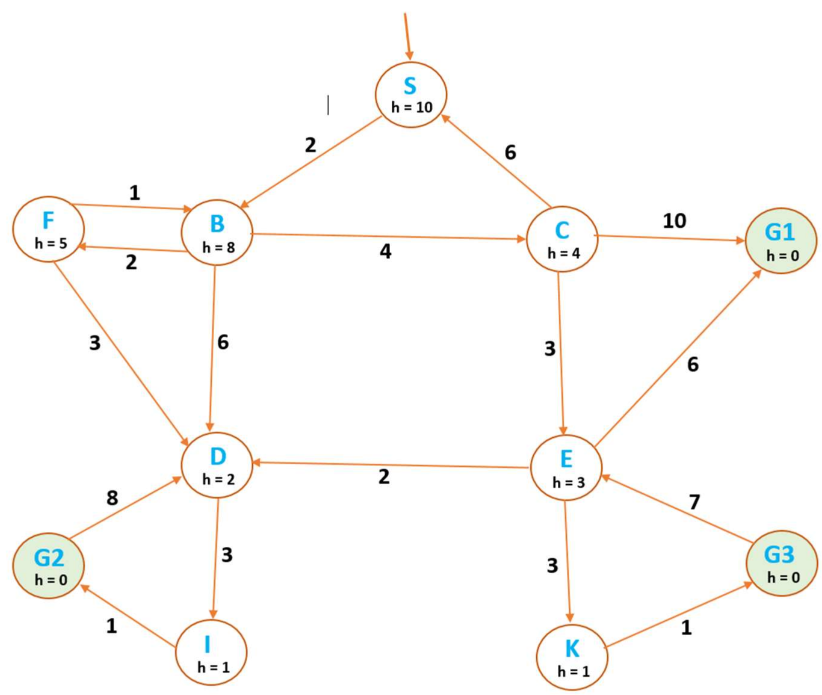
c) Consider an 8-puzzle problem with the following initial state:

2	8	7
3	1	4
5	6	

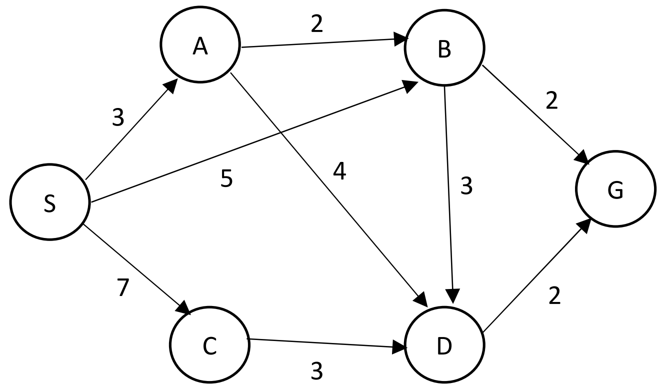
Now using depth first search draw the search tree starting from this initial state up to depth 3 (stop when the first node at depth 3 is generated) **[2]**

d) Consider the state-space graph in the following figure. Find out the solution paths and costs returned by the following search algorithms. **[2+2+2]**

i. UCS
ii. Greedy BFS
iii. A* Search



3. a) Consider the following state space graph:



Suppose you start from state S and you want to reach state G. You are using A* search with the following heuristic function:

$$h(x) = \begin{cases} 0 & \text{if x is a goal state} \\ \text{minimum cost among the outgoing actions from x,} & \text{if x is not a goal state} \end{cases}$$

Is h admissible? Is it consistent? Explain your answer. **[3]**

b) Suppose you have three different heuristic functions h1, h2 and h3 for a problem you are going to solve with A* tree search. The values are given for a state space graph in the following table along with the exact cost to goal.

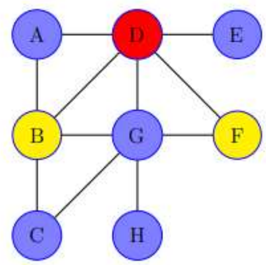
	S	A	B	C	D	G
h1	7	5	6	4	3	0
h2	0	0	0	0	0	0
h3	7	5	7	5	2	0
Exact	7	5	6	4	3	0

Now answer the following: **[3]**

i. Which heuristic will expand the maximum number of nodes but give optimal solution?
ii. Which heuristic will expand the minimum number of nodes and give optimal solution?
iii. Which one will give non-optimal solution?

4. a) Let G be the simple graph shown below. The problem is to find a coloring of V using colors red, blue, and yellow, so that no two adjacent vertices are assigned the same color. Let's define a state as an assignment of colors for all the vertices. Suppose we want to use greedy hill climbing search to solve the problem and reach the goal state (example shown in figure). Now answer the following: **[4]**

   i. What is the size of the state space?
   ii. How many successors does a state have?
   iii. What can be used as a suitable heuristic cost function?
   iv. Will the algorithm always find optimal solution?



b) Give the name of the algorithm that results from each of the following special cases: **[2]**

   i. Local beam search with k = 1.
   ii. Simulated annealing with T = 0 at all times.
   iii. A* Search with h=0 for all nodes
   iv. Uniform cost search with all arc costs equal to 1.

5. Consider the following game tree.

Max Min Terminal node

Show which nodes will be pruned if you use alpha-beta pruning with minimax search algorithm. **[3]**