CMSC 4200 - Artificial Intelligence

Spring 2026

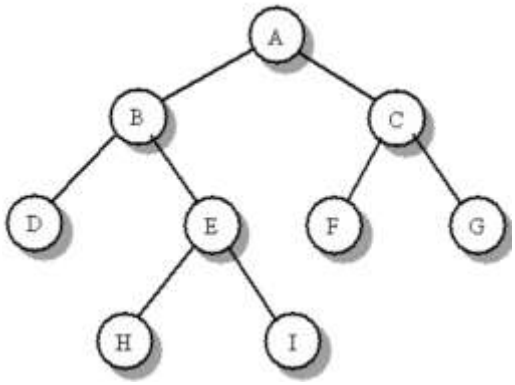
Group Assignment 2

Due – Tuesday, February 24, 2026, at 11:59pm EST

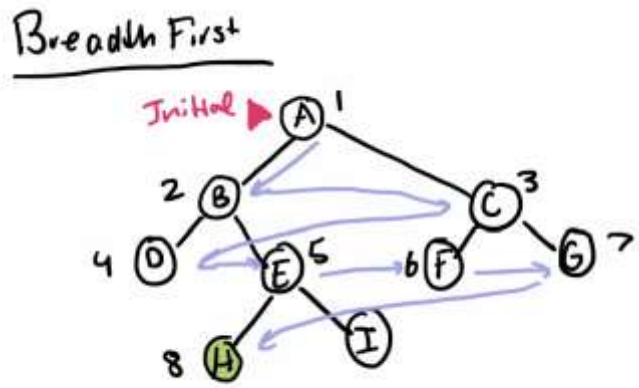
Group #	1
Names:	Margo Bonal
	Luke Ruffing
	Evan Thompson

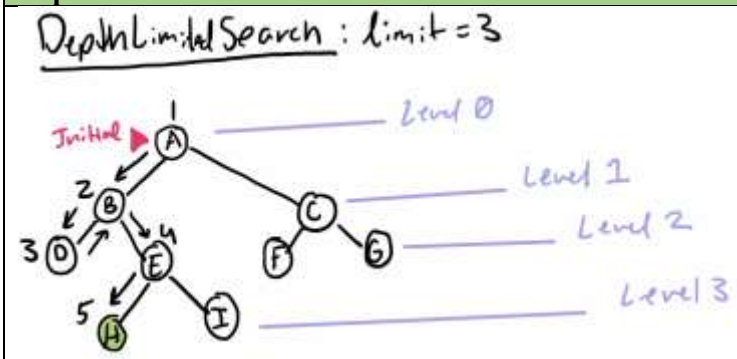
This assignment will be graded out of 20 points. Submit your answer file to the right D2L Dropbox named "Assignment 2 Dropbox" by the due date and time.

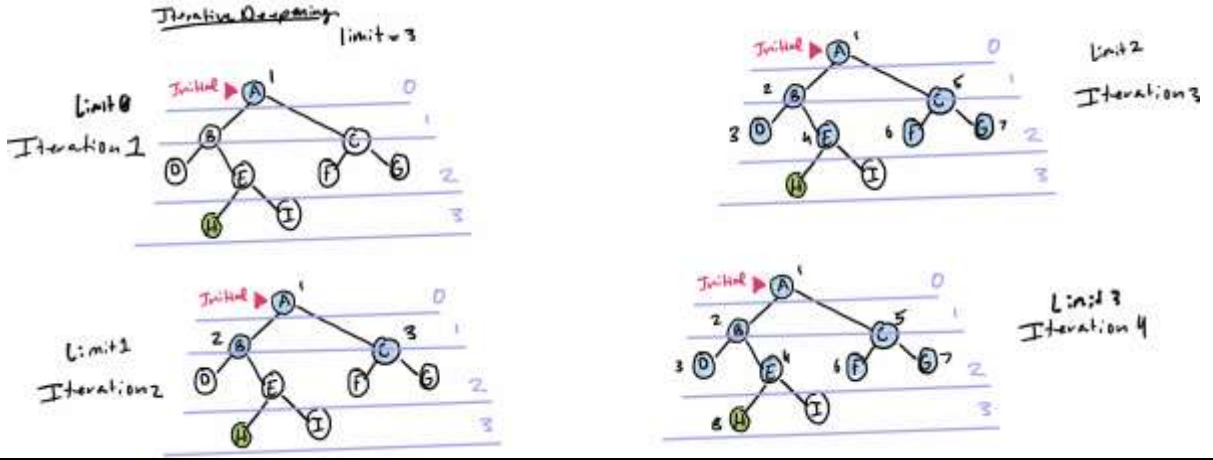
Question 1. (6 points) The state space shown in the figure below has 9 different states where the initial state is ***A***. List the order in which nodes will be visited with breadth-first search, depth-limited search with limit 3, and iterative deepening search when the goal state is ***H***.



Note: Following Diagrams show H = Goal State as green and Discovered nodes as blue.

Breadth First Search:	Explanation:
	The Breadth First Search expands nodes in width before depth. The diagram above shows path taken during the algorithms process. The Solution Path is represented as: A -> B -> C -> D -> E -> F -> G -> H **Completing process at goal state H**

Depth Limited Search:	Explanation:
	The Depth Limited Search expands nodes in depth before width like the Depth First search. However, it differs by putting a depth constraint or limit on how far the algorithm can search for depth. The diagram above shows path taken during the algorithms process. The Solution Path is represented as: A -> B -> D -> E -> H **Completing process at goal state H**

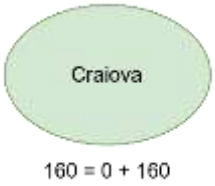
Iterative Deepening Search:


Iterative Deepening Search Explanation:
This search uses the benefits of the depth first and breadth first search but performs searches in iterations, running until the goal state is found. Since the depth of the goal is not known, the number of iterations is also not known. Iteration 1, Limit 0: A Iteration 2, Limit 1: A -> B -> C Iteration 3, Limit 2: A -> B -> D -> E -> C -> F -> G Iteration 4, Limit 3: A -> B -> D -> E -> C -> F -> G -> H

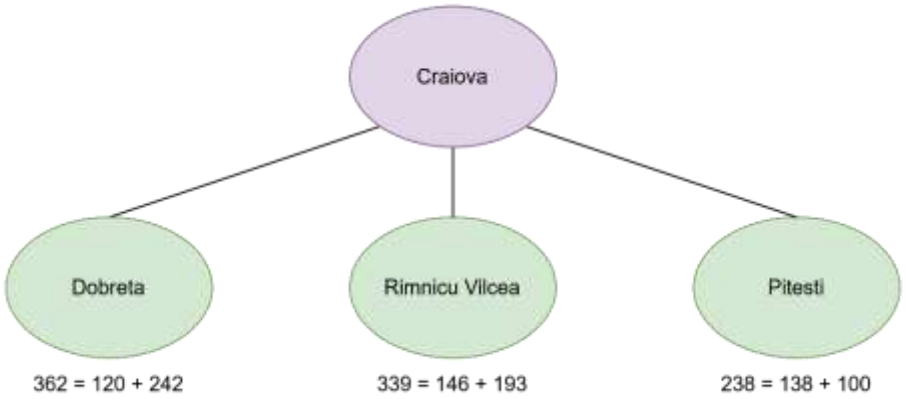
Question 2. (6 points) Trace the operation of A* search applied to the problem of getting to *Bucharest* from *Craiova* using the map of Romania and the straight-line distance heuristic. You need to draw the different stages of the search tree as shown in the slides 51 and 52 of Chapter 3.

Using evaluation function f(n) = g(n) + h(n), where g(n) is the cost from the initial state to node n, and h(n) is the estimated shortest cost from n to the goal state:

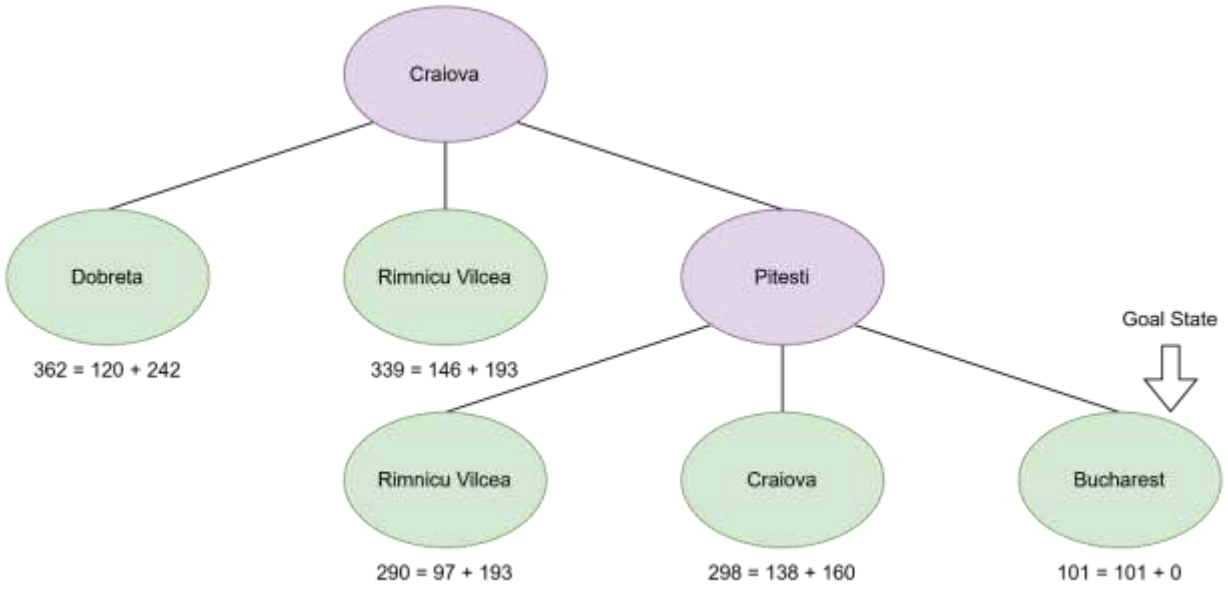
1. Start with Craiova, it is the only node so f(n) = 0 + 160:



2. Expand Craiova, find f(n) for each connected node:

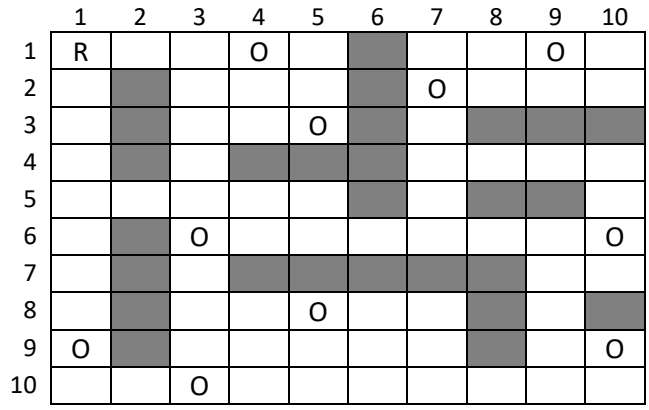


3. Expand Pitesti since it has the lowest f(n) value:



4. We have found the goal state which is Bucharest.

Question 3. (8 points) A room is divided into a 10 X 10 square grid, each grid square indicated by Square(row#, column#). There are 10 identical objects in the room indicated by "O" characters. Also in the room, there is a robot indicated by "R" in Square(1,1). This robot needs to collect all these 10 objects by passing through the squares of these objects and then that robot needs to return to Square(1,1). There are some dark squares representing some obstacles in which the robot cannot pass through them. An example setup is shown below.



The aim is to use a search algorithm to find a cost-optimal path, one with least total cost, that the robot can follow to pass through the 10 objects to pick them up and then it returns to its starting position Square(1,1). You can assume that the robot knows about the locations of all objects and obstacles. The robot can move horizontally and vertically through squares that have no obstacles but not diagonally. The robot can carry all these 10 objects without issue.

A) Give an efficient way to describe different states of this problem.

Each state must include the robots current position and the positions of the objects it has collected so far.

B) Give a complete and efficient problem formulation that can be used to run a search algorithm to find the best solution.

- States: Each state includes robots position and collected objects positions
- Initial state: (1,1) with 0 objects collected
- Actions: Up, Down, Left, Right
- Transition model:
  - Up – Moves the robot one cell up, and marks the cell that it collected an object if object is present
  - Down – Moves the robot one cell down, and marks the cell that it collected an object if object is present
  - Left – Moves the robot one cell left, and marks the cell that it collected an object if object is present
  - Right – Moves the robot one cell right, and marks the cell that it collected an object if object is present
- Goal states: Move through all states with an "O", return to state (1,1)
- Action cost: Each move costs 1

C) What search algorithm you will use to find an optimal path? Why?

A* is the most optimal search algorithm to use for this problem so we will be using it. A* search algorithm is the most optimal search algorithm for this problem because it is a complete and cost-optimal informed search algorithm. And since the robot has knowledge of where the objects and obstacles are an informed search algorithm is much more efficient to use in this case then an uninformed search algorithm.

D) Develop a heuristic function that you can use for an informed search algorithm to find an optimal solution? Provide some explanations for why you selected that heuristic function.

h(n) = Manhattan distance from current cell to closest remaining uncollected object

This heuristic function is admissible because the robot must travel at least that distance to reach the nearest object. Using Manhattan distance won't cause any overestimations so A* will guarantee a cost-optimal solution.