50.021 – AI

Kwan Hui, Dorien and Alex

Week 02: Search

[The following notes are compiled from various sources such as textbooks, lecture materials, Web resources and are shared for academic purposes only, intended for use by students registered for a specific course. In the interest of brevity, every source is not cited. The compiler of these notes gratefully acknowledges all such sources.]

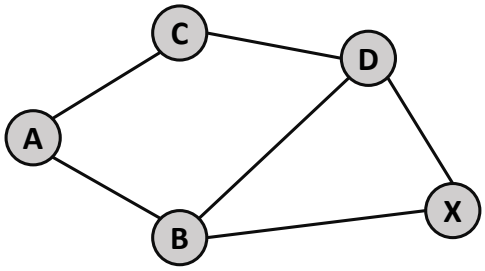
Due: 3rd June, 11:59pm
Submission: via eDimension

1 Problem Formulation

Missionaries and cannibals is a classical formal problem, and is generally stated as follows. Three missionaries and three cannibals are on one side of the river. They all need to cross in a boat that only holds two people at once. There must never be a situation where there is a group of missionaries in one place who are outnumbered by cannibals.

TASK: Formalise the missionaries and cannibals problem in terms of its state space, initial state, goal test, actions and path cost.

2 Tree Search VS Graph Search



Example: For the above graph, A is the initial state and X is the goal state. Assuming that we insert nodes in terms of lowest alphabatical order first. To perform Breadth-First Search (BFS) using graph search on the above graph, we have the following steps (and their frontier and explored set):

1. Frontier: A (Explored: A)
2. Frontier: AB, AC (Explored: A, B, C)
   - *Note: AB is inserted first, followed by AC*
3. Frontier: AC, ABD, ABX (Explored: A, B, C, D, X)

In this case, the solution is the path ABX.

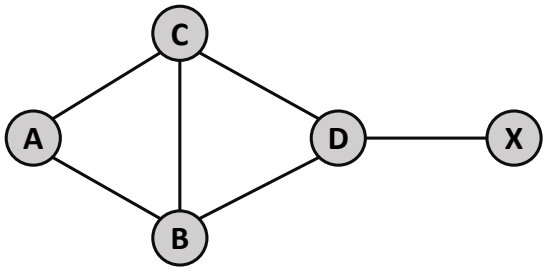
Similarly for Depth-First Search (DFS) using graph search on the same problem, we have:

1. Frontier: A (Explored: A)
2. Frontier: AB, AC (Explored: A, B, C)
3. Frontier: AB, ACD (Explored: A, B, C, D)
4. Frontier: AB, ACDX (Explored: A, B, C, D)
   - *Note: There is also ACDB, which we do not insert as we are using graph search*

In this case, the solution is the path ACDX.

TASK: Answer the following questions:
a.) Name the main difference between tree search and graph search.
b.) What is the difference between nodes and states in terms of a search problem?
c.) Does the explored set keep track of nodes or states? Why is it so?

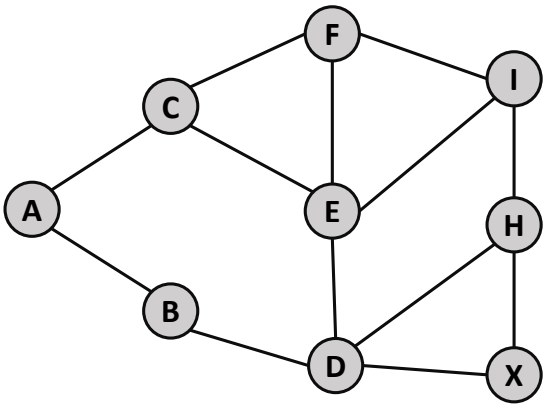
3 Breadth-First Search (BFS) and Depth-First Search (DFS)



For the above graph, A is the initial state and X is the goal state. Assuming that we insert nodes in terms of lowest alphabatical order first (as in 2).

TASK: Answer the following questions:
a.) Run BFS as a graph search, and list down the following: (i) the frontier/queue at every step; (ii) the explored set at every step; and (iii) the solution (if any).
b.) Run DFS as a graph search, and list down the following: (i) the frontier/queue at every step; (ii) the explored set at every step; and (iii) the solution (if any).
c.) If BFS is run as a tree search (instead of a graph search), what additional nodes will be inserted? List down 3 such nodes.
d.) If DFS is run as a tree search (instead of a graph search), what additional nodes will be inserted? List down 3 such nodes.

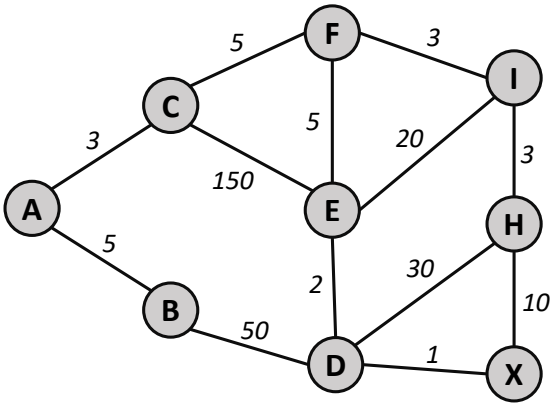
4 More BFS/DFS



For the above graph, A is the initial state and X is the goal state. Assuming that we insert nodes in terms of lowest alphabatical order first (as in 2).

TASK: Answer the following questions:
a.) Run BFS as a graph search, and list down the following: (i) the frontier/queue at every step; (ii) the explored set at every step; and (iii) the solution (if any).
b.) Run DFS as a graph search, and list down the following: (i) the frontier/queue at every step; (ii) the explored set at every step; and (iii) the solution (if any).

5 Uniform Cost Search (UCS)



For the above graph, A is the initial state and X is the goal state. The path

cost is written on the edges of the graph.

TASK: Run UCS as a graph search, and list down the following: (i) the frontier/queue at every step (including the path cost of each node); (ii) the explored set at every step; and (iii) the solution (if any).