50.021 – Artificial Intelligence

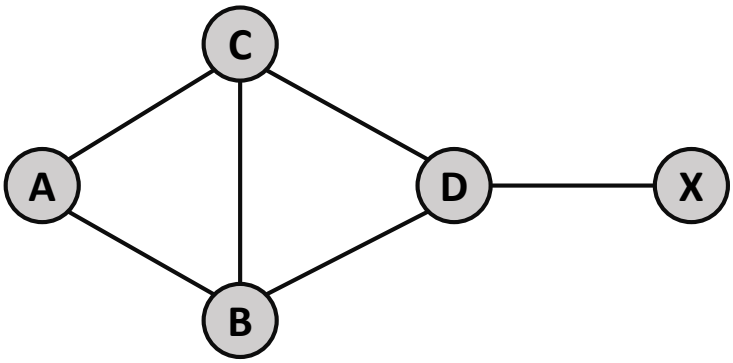
Kwan Hui

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.]

These answers are provided only as a brief guide. There could be more than one way to answer the questions.

1 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.

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

Answer: BFS as a graph search

- A
- AB, AC
- AC, ABD (ABC not added as C is in explored set / frontier)
- ABD
- ABDX (could also be returned without adding to frontier for BFS/DFS like in the pseudo codes)
- Solution: ABDX

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

Answer: DFS as a graph search

- A
- AB, AC
- AB, ACD (ACB not inserted because in explored set / frontier)
- AB, ACDX (ACDB not inserted because in explored / frontier)
- Solution: ACDX

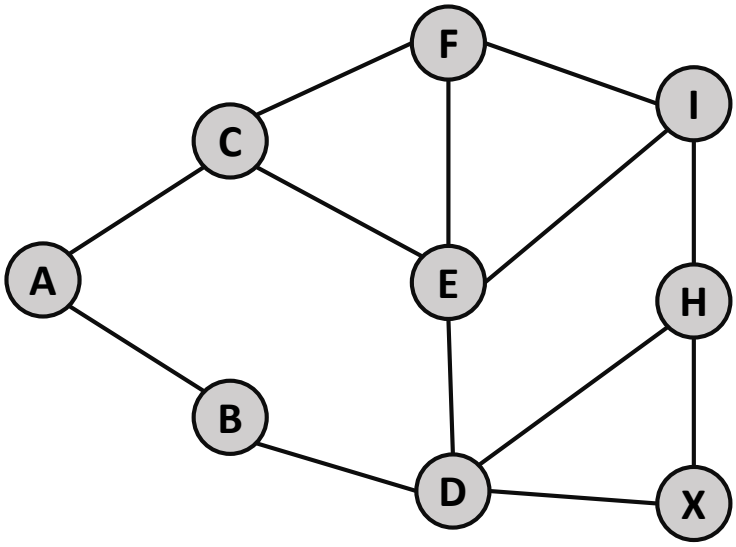
c.) If BFS is run as a tree search (instead of a graph search), what additional nodes will be inserted? List down 3 such nodes.

Answer: Tree search does not keep track of repeated states, so additional nodes can be any with a repeated state such as ABA, ABC, etc.

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.

Answer: Similar to 1c

2 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.

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

Answer: BFS as a graph search

- A
- AB, AC
- AC, ABD (ABC not added as C is in explored set / frontier)
- ABD, ACE, ACF
- ACE, ACF, ABDH, ABDX
- Solution: ABDX

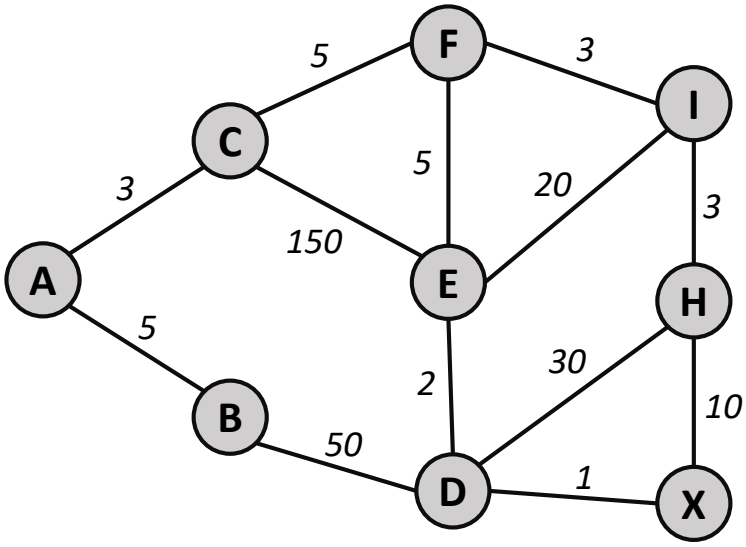
b.) Run DFS as a graph search, and list down the following: (i) the frontier/queue at every step; (iii) the solution (if any).

Answer: DFS as a graph search

- A
- AB, AC
- AB, ACE, ACF
- AB, ACE, ACFI

- AB, ACE, ACFIH
- AB, ACE, ACFIHD, ACFIHX
- Solution: ACFIHX

3 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); and (iii) the solution (if any).

Answer: Uniform Cost Search

- A-0
- AC-3, AB-5
- AB-5, ACF-8, ACE-153
- ACF-8, ABD-55, ACE-153
- ACFI-11, ACFE-13, ABD-55, (ACE-153 is replaced, as ACFE-13 is cheaper)
- ACFE-13, ACFIH-14, ABD-55 (ACFIE-33 is more ex than ACFE-13, so the former is ignored)
- ACFIH-14, ACFED-15 (ACFEI-33 is ignored as I is in the explored set and was removed from the frontier, so higher path cost. ABD-55 is replaced by ACFED-15 as it is cheaper.)

- ACFED-15, ACFIHX-24
- ACFEDX-16
- Nil (ACFEDX-16 is popped out and returned as solution)
- Solution: ACFEDX-16