Tutorial 3

1. Discuss the advantages and disadvantages of **breadth-first search** and **depth-first search**.

Answer:

	Breadth-first search	Depth-first search
Advantages	Guarantee a result if it exists Guarantee the shortest path If the goal is at higher level, then the search is fast, hence consumption of memory could be less	Guarantee a result if it exists Usually requires lesser memory as it only stores the nodes in the search path Usually the search time is faster
Disadvantage	Needs more memory to store all branches Large branching factors will lead to longer search time	Cannot guarantee shortest path It might stuck at infinite loop The search path can be very long if the goal is very deep down, so may also consume more memory

2. **Figure 1** shows a puzzle problem that requires rearrangement of the tiles to transform the order from start to goal state. One is only permitted to slide a tile **left**, **right**, **up** or **down** into the blank square.

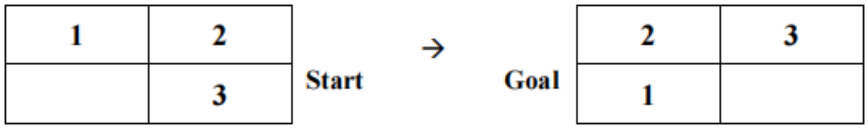


Figure 1: The Puzzle Problem

a) Provide the goal formulation of the puzzle problem above.
b) Formulate the puzzle problem above by specifying the initial state, successor functions, goal test, step cost, and path cost.
c) Perform **breadth-first search** and **depth-first search** on the puzzle problem above. Draw the resulting trees for both.

Answer:

a)

Goal formulation

Goal: to reach the goal state as shown in Figure 1

2	3
1	

Optimal solution: find the shortest path to reach the goal

Abstraction: time, material

b)

Problem formulation

Initial state: Start state

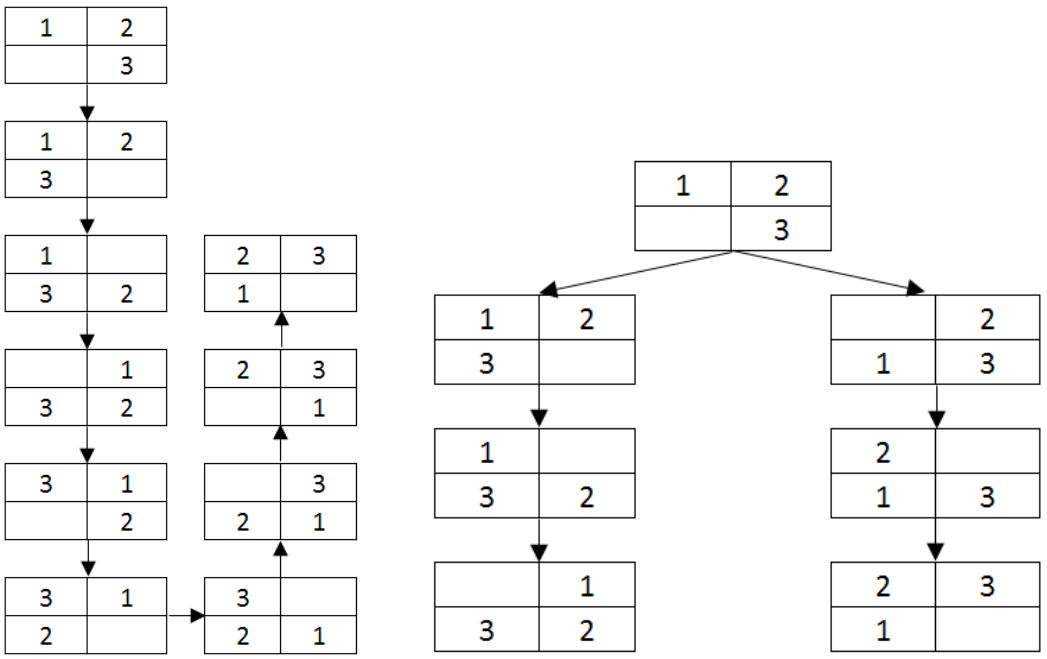
Successor function: Slide the empty tile follows the sequence of left, right, up and down

Goal test – to check if the goal state has been reached, otherwise continue to search

Step cost – the distance between one node to the other, e.g. each step is 1.

Path cost – the total step cost between start state and the goal state along a path

c)



3. **Figure 2** shows a puzzle problem that requires rearrangement of the tiles to transform the order from start to goal state. One is only permitted to slide a tile **down, right, up, or left** into the blank square.

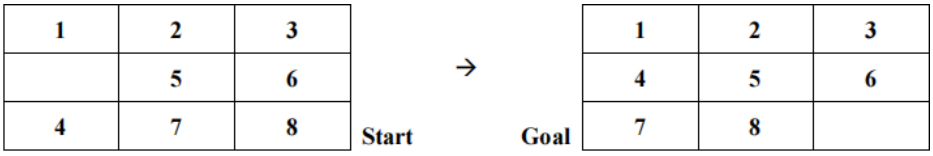


Figure 2: The 8-Puzzle Problem

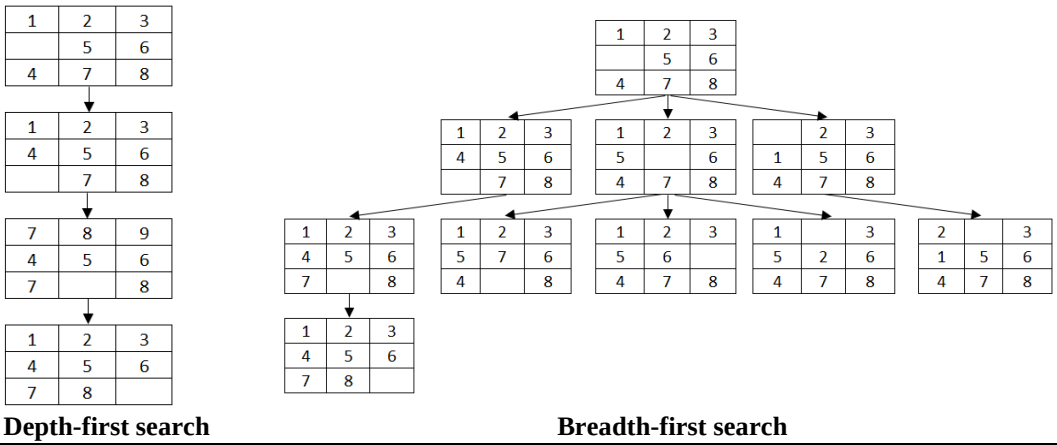
a) Perform **breadth-first search** and **depth-first search** on the 8-puzzle problem above. Draw the resulting trees for both. (Remark: you may stop the search at level 4)

b) Evaluate the efficiency of **breadth-first search** and **depth-first search** in terms of completeness, optimality, time efficiency and space efficiency in solving the problem above.

Answer:

a)

Successor function: Slide the empty tile follows the sequence of down, right, up and left



b)

Efficiency	Breadth-first search	Depth-first search
Completeness	Yes	Yes
Optimality	Yes	Yes
Time-complexity	Consume more time	Consume less time
Space complexity	Consume more memory	Consume less memory

4. In the family river-crossing problem, two parents are with their two children - a son and a daughter - came to a wide river. The only way to get to the other side was to ask a fisherman if he could lend them his boat. However, the boat could carry only two persons. For safety reason, no child should be left alone without the supervision of at least one parent. The family must get to the other side and finally returns the boat to the fisherman, assuming only the fisherman and the two parents know how to row the boat. Based on this family river-crossing problem, answer the following questions.
   a) Suggest a simple representation of the initial state. You must briefly explain the representation.
   b) Describe the goal formulation and problem formulation.
   c) Draw the depth-first search tree to show how all the states are being traversed.

Answer:

a)

Representation:

[2, 2, 1, L, 0, 0, 0] == ([LP LC LF L RP RC RF])

[no. of parents at left side, no. of children at left side, no. of fisherman at left side, position of boat, no. of parents at right side, no. of children at left side, no. of fisherman at right side])

b)

Goal formulation:

Goal- to reach [0 0 0 R 2 2 1] (The family and fisherman at the right hand side)

Optimal solution- least number of step (trip) to reach the goal

Abstraction- no time limit in reaching the goal (eg. Do not care about the time used to reach the other side of the river, which can be caused by unexpected factors such as weather)

Problem Formulation

Initial state - [2, 2, 1, left, 0, 0, 0]

Successor Function - The possible actions available to the agent that will change the state from the current state.

	If boat at left side	If boat at right side
1F	L to R	R to L
1F+1P		
1P+1C		
1P		

Therefore, the successor functions are shown as below:

If boat at left:

[LP LC LF-1 R RP RC RF+1]

[LP-1 LC LF-1 R RP+1 RC RF+1]

[LP-1 LC-1 LF R RP+1 RC+1 RF]

[LP-1 LC LFR RP+1 RC RF]

If boat at right:

[LP LC LF-1 L RP RC RF+1]

[LP-1 LC LF-1 L RP+1 RC RF+1]

[LP-1 LC-1 LF L RP+1 RC+1 RF]

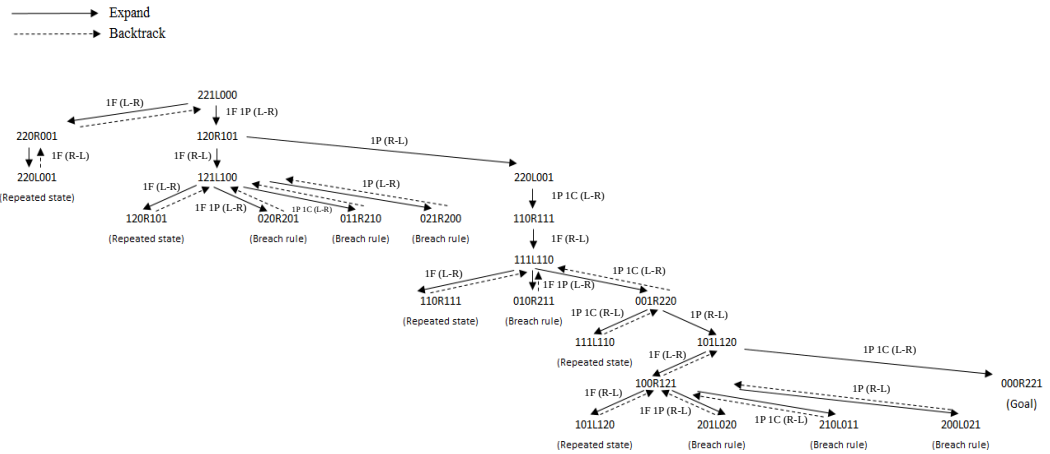
[LP-1 LC LF L RP+1 RC RF]

F: Fisherman; P:Parent; C:Child; L:Left; R:Right

2P and 1C are not feasible as it breach the rule of "no child should be left alone without the supervision of at least one parent"

Goal Test - To check whether the given state is a goal state. (state([0,0,0,R,2,2,1]))
Step Cost - The route distance from one state to another state. (1)
Path Cost - The sum of costs of the individual actions along the path.

c)



5. **Figure 3** below shows a directed graph. Assume that the traversal would start from **Vertex 0** to **Vertex 7**. All vertices to be visited in **ascending order** (i.e. from smaller number to bigger number).

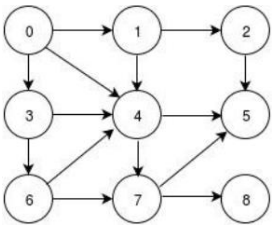
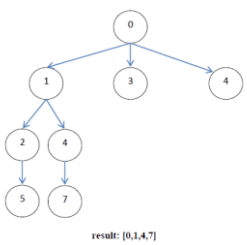


Figure 3: The Directed Graph

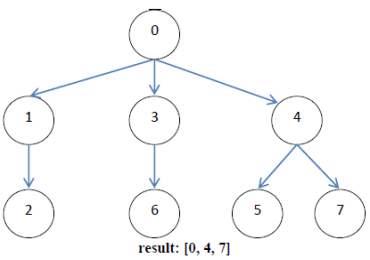
a) Perform a **depth-first search** on the directed graph to traverse from Vertex 0 to Vertex 7. Draw the resulting tree and list the returned path.
b) Perform a **breadth-first search** on the directed graph to traverse from Vertex 0 to Vertex 7. Draw the resulting tree and list the returned path. Avoid repeated state.
c) Evaluate the efficiency of **breadth-first search** and **depth-first search** in terms of completeness, optimality, time efficiency and space efficiency in solving the problem above.

Answer:

a) b)



Depth-first search



Breadth-first search

c)

Efficiency	Breadth-first search	Depth-first search