Artificial Intelligence—Lab

Department of CSE

SSN College of Engineering

Session2: State Space Search—Decantation Problem

26 July 2021

You are given an 8-litre jar full of water and two empty jars of 5- and 3-litre capacity. You have to get exactly 4 litres of water in one of the jars. You can completely empty a jar into another jar with space or completely fill up a jar from another jar.

1. Formulate the problem: Identify states, actions, initial state, goal state(s). Represent the state by a 3-tuple. For example, the intial state state is (8,0,0). (4,1,3) is a goal state (there may be other goal states also).

2. Use a suitable data structure to keep track of the parent of every state. Write a function to print the sequence of states and actions from the initial state to the goal state.

3. Write a function next states(s) that returns a list of successor states of a given state s.

4. Implement Breadth-First-Search algorithm to search the state space graph for a goal state that produces the required sequence of pourings. Use a Queue as frontier that stores the discovered states yet be explored. Use a dictionary for explored that is used to store the explored states.

5. Modify your program to trace the contents of the Queue in your algorithm. How many states are explored by your algorithm?

```
Decantation Problem

Initial State                       : (8, 0, 0)
Total no. of states explored by BFS : 16

No. of Goal States found            : 3

Goal States found                   : [(1, 4, 3), (4, 4, 0), (4, 1,
3)]

Details                             :

explored_states :  [(8, 0, 0), (3, 5, 0), (5, 0, 3), (0, 5, 3), (3, 2,
3), (5, 3, 0), (6, 2, 0), (2, 3, 3), (6, 0, 2), (2, 5, 1), (1, 5, 2),
(7, 0, 1), (1, 4, 3), (7, 1, 0), (4, 4, 0), (4, 1, 3)]
Goal States :  [(1, 4, 3), (4, 4, 0), (4, 1, 3)]

Goal State                          : (1, 4, 3)
```

Path taken by the BFS Algorithm to reach the Goal State:

```
-------------------------------------------------
|      Jug 1   |      Jug 2   |      Jug 3    |
-------------------------------------------------
|        8 L   |        0 L   |        0 L    |
|        3 L   |        5 L   |        0 L    |
|        3 L   |        2 L   |        3 L    |
|        6 L   |        2 L   |        0 L    |
|        6 L   |        0 L   |        2 L    |
|        1 L   |        5 L   |        2 L    |
|        1 L   |        4 L   |        3 L    |
-------------------------------------------------
```

Goal State   : (4, 4, 0)

Path taken by the BFS Algorithm to reach the Goal State:

```
-------------------------------------------------
|      Jug 1   |      Jug 2   |      Jug 3    |
-------------------------------------------------
|        8 L   |        0 L   |        0 L    |
|        3 L   |        5 L   |        0 L    |
|        3 L   |        2 L   |        3 L    |
|        6 L   |        2 L   |        0 L    |
|        6 L   |        0 L   |        2 L    |
|        1 L   |        5 L   |        2 L    |
|        1 L   |        4 L   |        3 L    |
|        4 L   |        4 L   |        0 L    |
-------------------------------------------------
```

Goal State   : (4, 1, 3)

Path taken by the BFS Algorithm to reach the Goal State:

```
-------------------------------------------------
|      Jug 1   |      Jug 2   |      Jug 3    |
-------------------------------------------------
|        8 L   |        0 L   |        0 L    |
|        5 L   |        0 L   |        3 L    |
|        5 L   |        3 L   |        0 L    |
|        2 L   |        3 L   |        3 L    |
|        2 L   |        5 L   |        1 L    |
|        7 L   |        0 L   |        1 L    |
|        7 L   |        1 L   |        0 L    |
|        4 L   |        1 L   |        3 L    |
-------------------------------------------------
```