DEPTH-FIRST SEARCH – WATER JUG PROBLEM
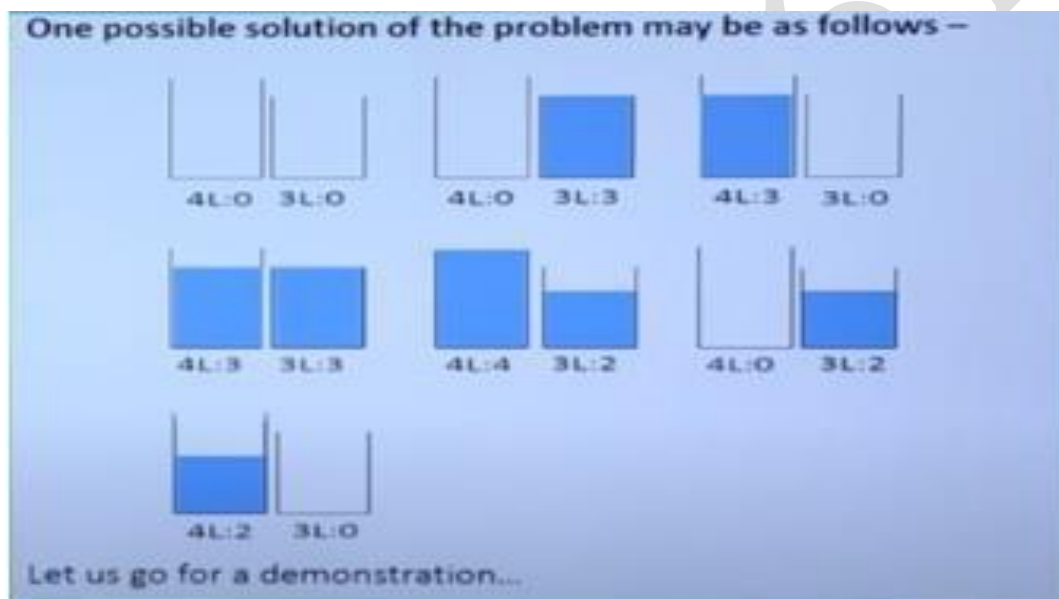
In the **water jug problem in Artificial Intelligence**, we are provided with two jugs: one having the capacity to hold 3 gallons of water and the other has the capacity to hold 4 gallons of water. There is no other measuring equipment available and the jugs also do not have any kind of marking on them. So, the agent's task here is to fill the 4-gallon jug with 2 gallons of water by using only these two jugs and no other material. Initially, both our jugs are empty.

One possible solution of the problem may be as follows –

4L:0 3L:0   4L:0 3L:3   4L:3 3L:0

4L:3 3L:3   4L:4 3L:2   4L:0 3L:2

4L:2 3L:0

Let us go for a demonstration...

```python
            ("fill", 1), ("fill", 2), ("empty", 1), ("empty", 2), ("pour", 1, 2), ("pour", 2, 1)
        ]

        for action in actions:
            if action[0] == "fill":
                if action[1] == 1:
                    next_state = (jug1_cap, j2)
                else:
                    next_state = (j1, jug2_cap)
            elif action[0] == "empty":
                if action[1] == 1:
                    next_state = (0, j2)
                else:
                    next_state = (j1, 0)
            else:  # action[0] == "pour"
                if action[1] == 1:
                    amount = min(j1, jug2_cap - j2)
                    next_state = (j1 - amount, j2 + amount)
                else:
                    amount = min(j2, jug1_cap - j1)
                    next_state = (j1 + amount, j2 - amount)

            if next_state not in visited:
                next_seq = seq + [action]
                result = dfs(next_state[0], next_state[1], next_seq, visited)
                if result:
                    return result

        return None

    # Initialize the DFS
    visited = set()
    return dfs(0, 0, [], visited)

# Example usage
result = water_jug_problem_dfs(4, 3, 2)
print(result)
```

```
[('fill', 1), ('fill', 2), ('empty', 1), ('pour', 2, 1), ('fill', 2), ('pour', 2, 1)]
```