

Experiment No: 3

Name:- Winstar Pereira

Class:- TE COMPS – A

Roll No:- 9569

Title: Use DFS problem solving method for

- a) Water Jug Problem
- b) Missionaries & Cannibals

Postlab:

1. What is the time complexity of the Water Jug problem?

Ans: The time complexity of the water jug problem using the Depth-First Search (DFS) algorithm is $O(m*n)$, where m and n are the capacities of the two water jugs.

The reason for this time complexity is that in the worst-case scenario, the DFS algorithm needs to explore all possible states (combinations of water levels in the two jugs) to find the solution. Since the maximum water level in each jug can go up to m and n respectively, the total number of possible states is mn , which leads to a time complexity of $O(mn)$.

2. Why is DFS not used for solving a water jug problem?

Ans: The Breadth-First Search (BFS) approach is more widely used than the Depth-First Search (DFS) approach for solving the water jug problem because:

a) Optimality:

The water jug problem normally seeks the shortest set of steps to obtain the desired amount of water. BFS ensures the optimal solution, but DFS may not always discover the shortest path.

b) Memory Requirement:

The number of alternative states in the water jug problem can be rather considerable, particularly for bigger jug capacity. BFS employs a queue to keep track of the states to be investigated, whereas DFS uses a stack, which can result in increased memory requirements in the worst-case scenario.

c) Backtracking:

DFS involves a lot of backtracking, as it explores one path as far as possible before backtracking to explore other paths. This can be inefficient for the water jug problem, where the solution may be found on a different branch of the search tree.

d) Completeness:

BFS is guaranteed to find a solution if one exists, while DFS may get stuck in an infinite loop or fail to find a solution, especially when dealing with cyclic graphs (which can occur in the water jug problem).