**Task – 3**

Water Jug Problem using Depth-First Search (DFS)

Objective: Measure a target amount of water using two jugs with given capacities.

Algorithm: Depth-First Search (DFS) to explore all possible states.

Functions: - gcd(a,b): Compute the greatest common divisor. - get\_neighbors(state, cap1, cap2): Generate possible moves from current state. - dfs\_water\_jug(cap1, cap2, target): Main DFS function to find solution.

Input: Capacities of two jugs and the target water volume.

Validation: - Target = 0 → Already satisfied. - Target > max(cap1, cap2) → No solution. - Target not multiple of gcd(cap1, cap2) → No solution.

State Representation: Tuple (x, y) = (Jug1 amount, Jug2 amount).

Possible Actions: - Fill Jug1/Jug2 - Empty Jug1/Jug2 - Pour Jug1 -> Jug2 or Jug2 -> Jug1

DFS Exploration: - Start from (0,0), explore neighbors using stack. - Track visited states to avoid cycles. - Maintain parent mapping to reconstruct solution path.

Output: - Print steps from start state to goal. - Show each action applied and resulting state.

Example Runs: - dfs\_water\_jug(4,3,2) - dfs\_water\_jug(5,3,4)

