**1.11 DYNAMIC PROGRAMMING**

**AIM**:

To find the number of ways to move the ball out of the grid boundary in exactly N steps from m\*n grid.

**ALGORITHM:**

1. Define a recursive function dfs(m, n, N, i, j):

2. If (i, j) is outside the grid → return 1 (valid way).

3. If N == 0 → return 0 (no moves left, but not outside).

4. Otherwise, recursively try 4 directions: up, down, left, right.

5. Use memoization (cache results) to avoid recomputation.

6. Final answer = dfs(m, n, N, i, j).

**PROGRAM:**

A screenshot of a computer program

AI-generated content may be incorrect.

Input:

m = 2, n = 2, N = 2, i = 0, j = 0

Output:

A screenshot of a computer

AI-generated content may be incorrect.

**RESULT:**

Thus the program is successfully executed, and the output is verified.

**PERFORMANCE ANALYSIS:**

* States = N × m × n
* Transitions = 4 per state
* Time Complexity = O(N × m × n)
* Space Complexity = O(N × m × n) (due to memoization table)