22. Graph Traversal using Depth First Search (DFS)

Aim:

To traverse a graph using DFS.

Algorithm:

1. Start from the source vertex.
2. Mark it visited and print it.
3. Recursively visit all unvisited adjacent vertices.

CODE:

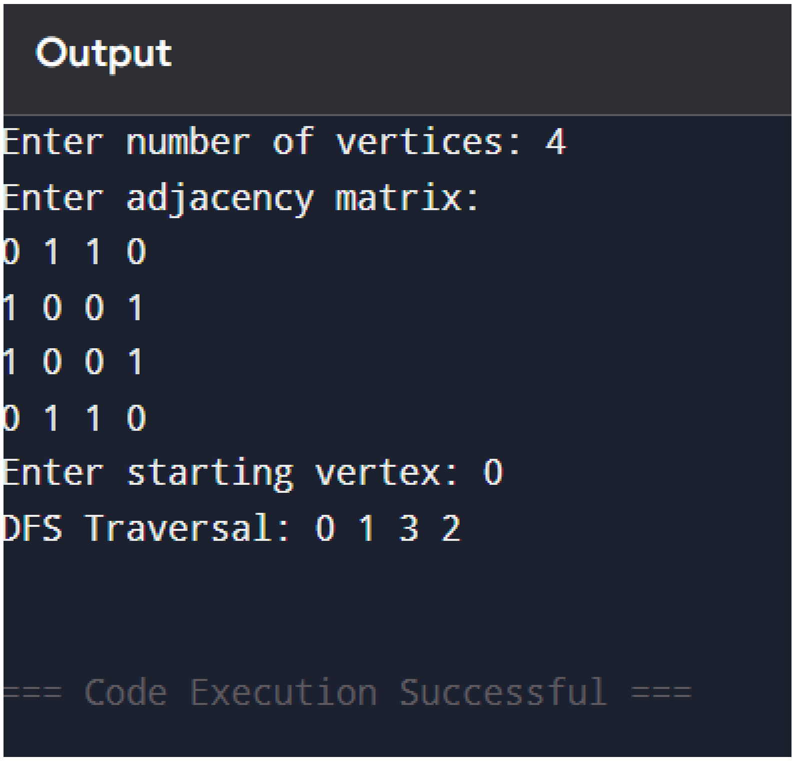
```
#include <stdio.h>
#define MAX 20

int visited[MAX];

void DFS(int adj[MAX][MAX], int n, int v) {
    visited[v] = 1;
    printf("%d ", v);
    for (int i = 0; i < n; i++) {
        if (adj[v][i] && !visited[i]) {
            DFS(adj, n, i);
        }
    }
}

int main() {
    int n, adj[MAX][MAX], start;
    printf("Enter number of vertices: ");
    scanf("%d", &n);
    printf("Enter adjacency matrix:\n");
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            scanf("%d", &adj[i][j]);
    printf("Enter starting vertex: ");
    scanf("%d", &start);

    for (int i = 0; i < n; i++) visited[i] = 0;
    printf("DFS Traversal: ");
    DFS(adj, n, start);
    printf("\n");
    return 0;
}
```



```
Enter number of vertices: 4
Enter adjacency matrix:
0 1 1 0
1 0 0 1
1 0 0 1
0 1 1 0
Enter starting vertex: 0
DFS Traversal: 0 1 3 2


=== Code Execution Successful ===
```

RESULT:
The program successfully executed and displayed the graph traversal using dfs.