**Program 9b**

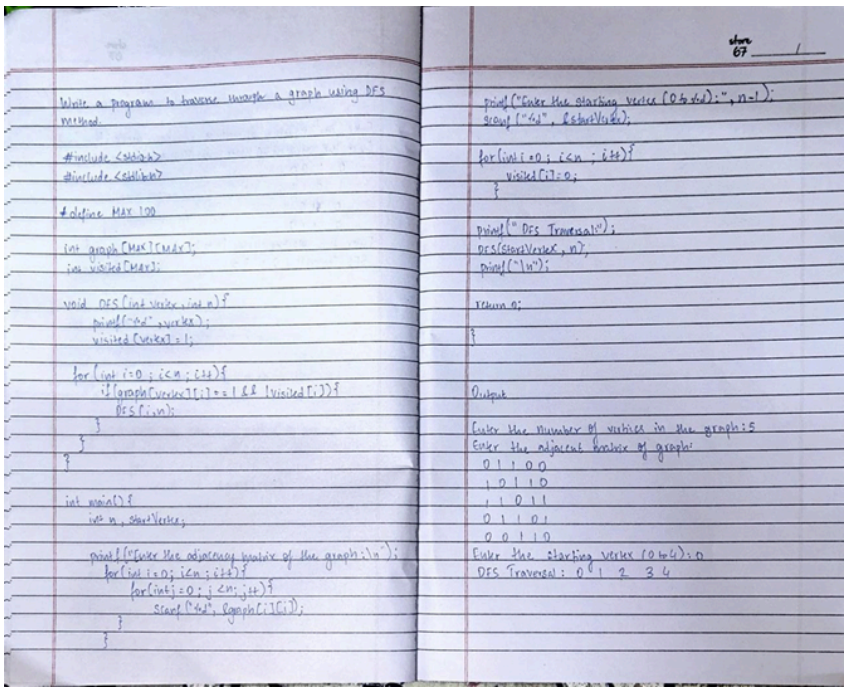
Write a program to traverse through a graph using DFS method.

Code:

```
#include <stdio.h>
#include <stdlib.h>
#define MAX 100
int graph[MAX][MAX];
int visited[MAX];
void DFS(int vertex, int n) {
   printf("%d ", vertex);
   visited[vertex] = 1;

        for (int i = 0; i < n; i++) {
        if (graph[vertex][i] == 1 && !visited[i]) {
        DFS(i, n);
        }
        }
}
int main() {
        int n, startVertex;

   printf("Enter the number of vertices in the graph: ");
```

```
    scanf("%d", &n);
     printf("Enter the adjacency matrix of the graph:\n");
            for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            scanf("%d", &graph[i][j]);
            }
            }
  printf("Enter the starting vertex (0 to %d): ", n - 1);
    scanf("%d", &startVertex);


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

```
Enter the number of vertices in the graph: 5
Enter the adjacency matrix of the graph:
0 1 1 0 0
1 0 1 1 0
1 1 0 1 1
0 1 1 0 1
0 0 1 1 0
Enter the starting vertex (0 to 4): 0
DFS Traversal: 0 1 2 3 4
```



Write a program to traverse through a graph using DFS method.

```
#include <stdio.h>
#include <stdbool.h>

#define MAX 100

int graph[MAX][MAX];
int visited[MAX];

void DFS(int vertex, int n){
    printf("%d ", vertex);
    visited[vertex] = 1;

    for(int i=0; i<n; i++){
        if(graph[vertex][i] == 1 && !visited[i]){
            DFS(i, n);
        }
    }
}

int main(){
    int n, startVertex;

    printf("Enter the adjacency matrix of the graph:\n");
    for(int i=0; i<n; i++){
        for(int j=0; j<n; j++){
            scanf("%d", &graph[i][j]);
        }
    }

    printf("Enter the starting vertex (0 to %d): ", n-1);
    scanf("%d", &startVertex);

    for(int i=0; i<n; i++){
        visited[i] = 0;
    }

    printf("DFS Traversal: ");
    DFS(startVertex, n);
    printf("\n");

    return 0;
}
```

Output

```
Enter the number of vertices in the graph: 5
Enter the adjacent matrix of graph:
0 1 1 0 0
1 0 1 1 0
1 1 0 1 1
0 1 1 0 1
0 0 1 1 0
Enter the starting vertex (0 to 4): 0
DFS Traversal: 0 1 2 3 4
```