EXPERIMENT NO.7
Name: Atharv Uday Wadadekar
Class: SY-IT
Roll No.: 66
Program:

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

#define MAX_VERTICES 20

int V, E;
int G[MAX_VERTICES][MAX_VERTICES];
int visited[MAX_VERTICES];

void DFS(int i);
void bfs(int v);

int main() {
   int i, j, v1, v2, source;

   printf("GRAPH\n");

   printf("Enter the number of vertices: ");
   scanf("%d", &V);

   printf("Enter the number of edges: ");
   scanf("%d", &E);

   for (i = 0; i < V; i++) {
      for (j = 0; j < V; j++) {
         G[i][j] = 0;
      }
   }

   for (i = 0; i < E; i++) {
      printf("Enter edge (format: V1, V2): ");
      scanf("%d %d", &v1, &v2);
      G[v1 - 1][v2 - 1] = 1;
      G[v2 - 1][v1 - 1] = 1;
   }

   for (i = 0; i < V; i++) {
      for (j = 0; j < V; j++) {
         printf("%d", G[i][j]);
      }
      printf("\n");
   }

   printf("Enter the source vertex: ");
   scanf("%d", &source);

   printf("DFS traversal starting from vertex %d: ", source);
```

```
    DFS(source - 1);

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

    printf("\nBFS traversal starting from vertex %d: ", source);
    bfs(source - 1);
    printf("\n");
    return 0;
}

void DFS(int i) {
    int j;
    visited[i] = 1;
    printf("%d->", i + 1);
    for (j = 0; j < V; j++) {
        if (G[i][j] == 1 && visited[j] == 0) {
            DFS(j);
        }
    }
}

void bfs(int v) {
    int q[MAX_VERTICES];
    int r = -1, f = -1;
    q[++r] = v;

    while (f < r) {
        v = q[++f];
        visited[v] = 1;
        printf("%d->", v + 1);

        for (int i = 0; i < V; i++) {
            if (G[v][i] != 0 && visited[i] == 0) {
                q[++r] = i;
                visited[i] = 1;
            }
        }
    }
}
```

Output:

```
GRAPH
Enter the number of vertices: 9
Enter the number of edges: 8
Enter edge (format: V1, V2): 1 2
Enter edge (format: V1, V2): 8 3
Enter edge (format: V1, V2): 7 5
Enter edge (format: V1, V2): 1 4
Enter edge (format: V1, V2): 6 8
Enter edge (format: V1, V2): 1 6
Enter edge (format: V1, V2): 7 2
Enter edge (format: V1, V2): 1 0
010101000
100000100
000000010
100000000
000000100
100000010
010010000
001001000
000000000
Enter the source vertex: 7
DFS traversal starting from vertex 7: 7->2->1->4->6->8->3->5->
BFS traversal starting from vertex 7: 7->2->5->1->4->6->8->3->
```