Exp 07

Program:

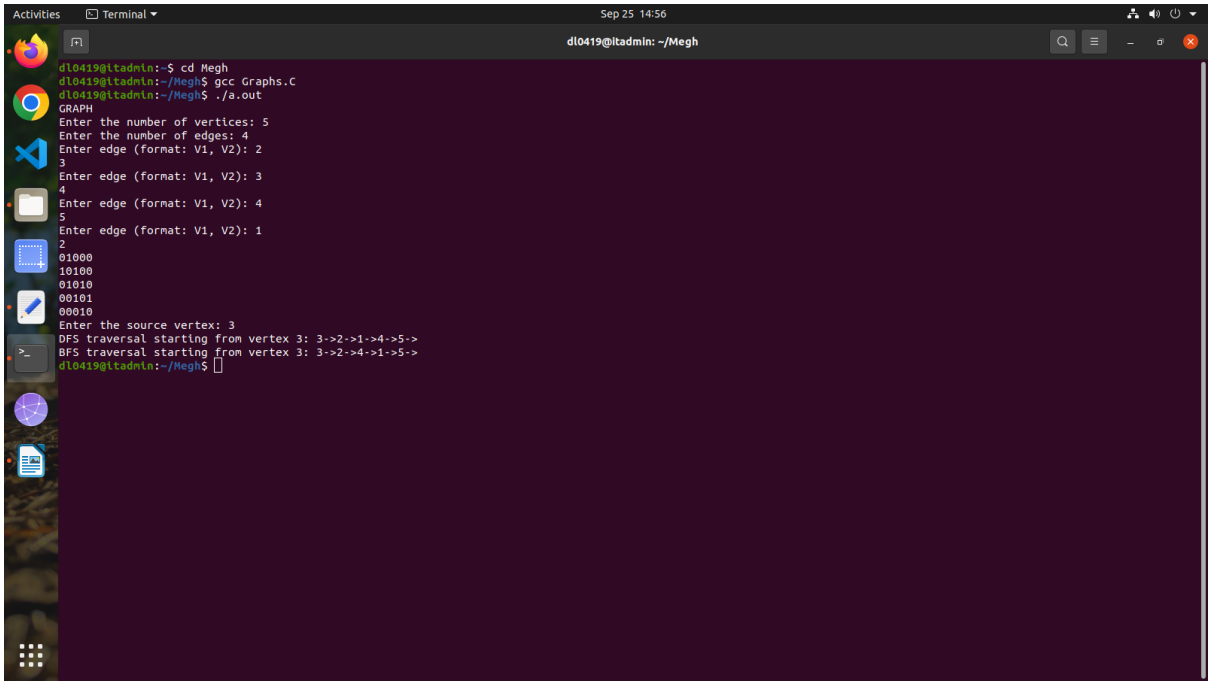
```
#include <stdio.h>
#include <stdlib.h>
#define MAX_VERTICES 20
int V, E;
int G[MAX_VERTICES][MAX_VERTICES];
int visited[MAX_VERTICES];
void DFS(int i);
void bfs(int v);
int main() {
	int i, j, v1, v2, source;
	printf("GRAPH\n");
	printf("Enter the number of vertices: ");
	scanf("%d", &V);
	printf("Enter 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:


Activities
Terminal
Sep 25 14:56
dl0419@itadmin: ~/Megh
dl0419@itadmin:~$ cd Megh
dl0419@itadmin:~/Megh$ gcc Graphs.C
dl0419@itadmin:~/Megh$ ./a.out
GRAPH
Enter the number of vertices: 5
Enter the number of edges: 4
Enter edge (format: V1, V2): 2
3
Enter edge (format: V1, V2): 3
4
Enter edge (format: V1, V2): 4
5
Enter edge (format: V1, V2): 1
2
01000
10100
01010
00101
00010
Enter the source vertex: 3
DFS traversal starting from vertex 3: 3->2->1->4->5->
BFS traversal starting from vertex 3: 3->2->4->1->5->
dl0419@itadmin:~/Megh$