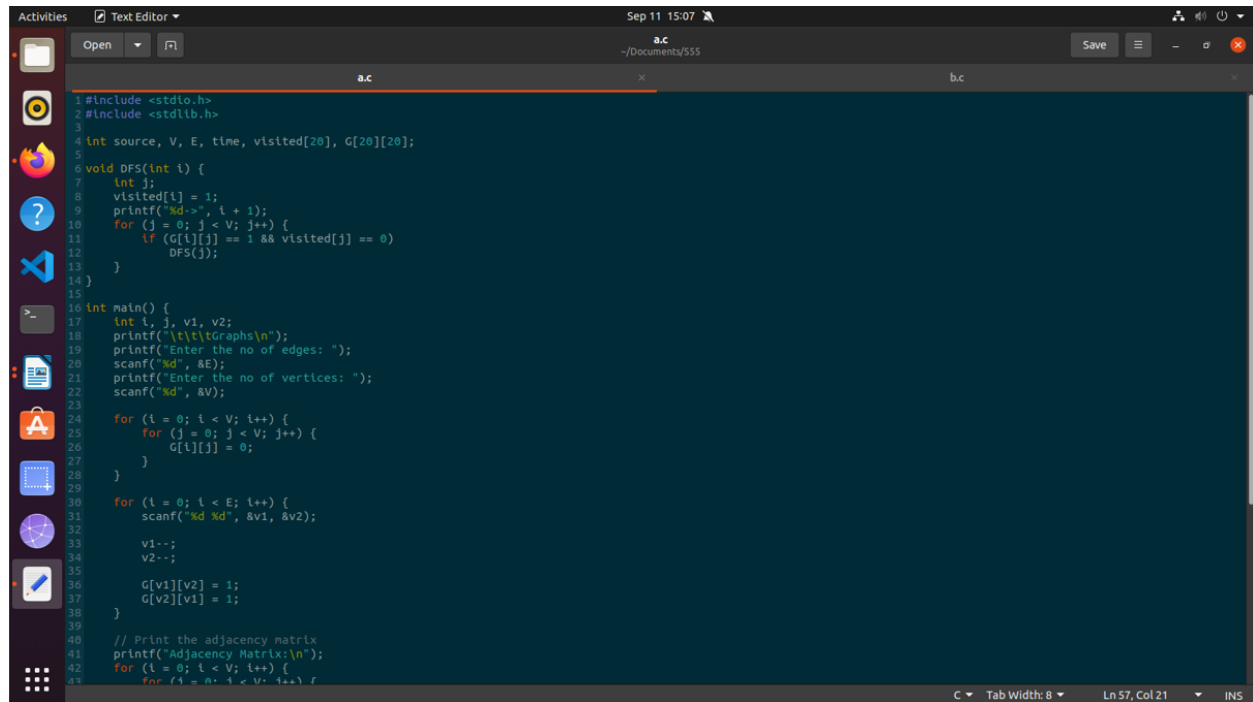
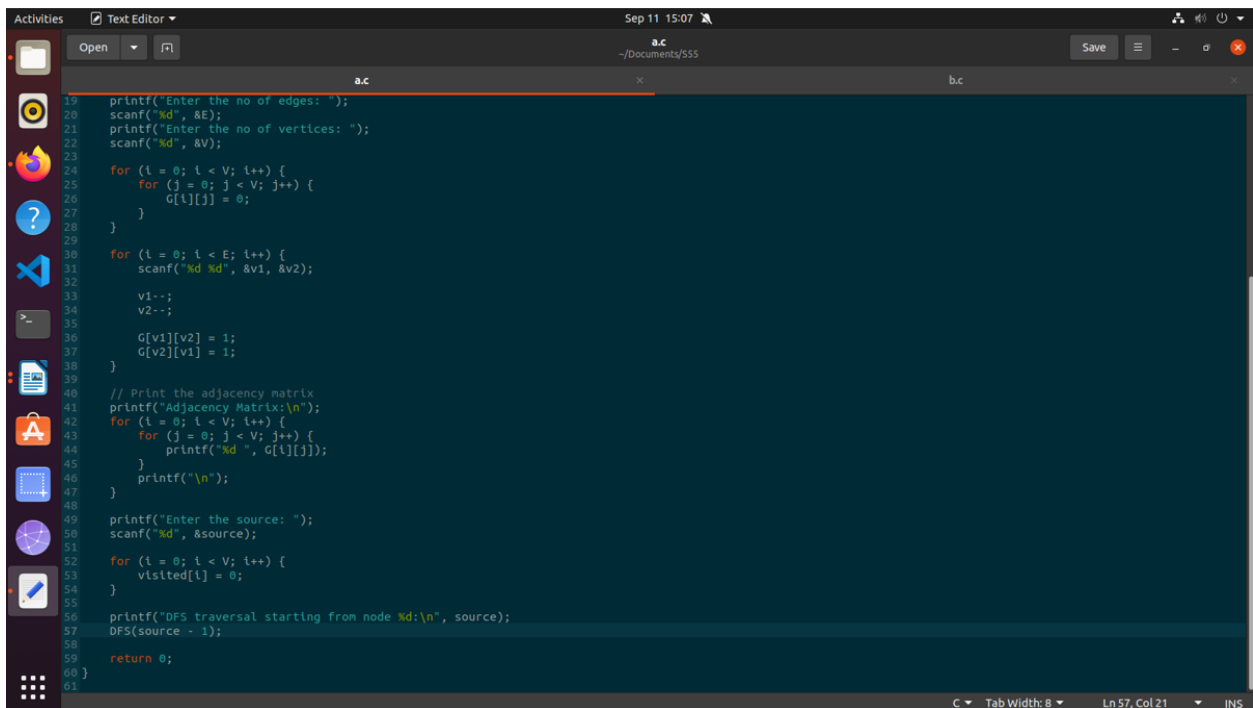


EXP 07
SANSKAR SRIVASTAVA
57
DFS



The screenshot shows a C++ code editor with a dark theme. The code is for a Depth-First Search (DFS) algorithm. It includes standard headers, defines a graph structure, and implements the DFS function. The main function prompts the user for the number of edges and vertices, constructs the adjacency matrix, and prints it.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int source, V, E, time, visited[20], G[20][20];
5
6 void DFS(int i) {
7     int j;
8     visited[i] = 1;
9     printf("%d->", i + 1);
10    for (j = 0; j < V; j++) {
11        if (G[i][j] == 1 && visited[j] == 0)
12            DFS(j);
13    }
14 }
15
16 int main() {
17     int i, j, v1, v2;
18     printf("\t\t\tGraphs\n");
19     printf("Enter the no of edges: ");
20     scanf("%d", &E);
21     printf("Enter the no of vertices: ");
22     scanf("%d", &V);
23
24     for (i = 0; i < V; i++) {
25         for (j = 0; j < V; j++) {
26             G[i][j] = 0;
27         }
28     }
29
30     for (i = 0; i < E; i++) {
31         scanf("%d %d", &v1, &v2);
32
33         v1--;
34         v2--;
35
36         G[v1][v2] = 1;
37         G[v2][v1] = 1;
38     }
39
40     // Print the adjacency matrix
41     printf("Adjacency Matrix:\n");
42     for (i = 0; i < V; i++) {
43         for (j = 0; j < V; j++) {
44             printf("%d ", G[i][j]);
45         }
46         printf("\n");
47     }
48
49     printf("Enter the source: ");
50     scanf("%d", &source);
51
52     for (i = 0; i < V; i++) {
53         visited[i] = 0;
54     }
55
56     printf("DFS traversal starting from node %d:\n", source);
57     DFS(source - 1);
58
59     return 0;
60 }
```



The screenshot shows the same C++ code editor, but with the final part of the code visible. The code includes the DFS function and the main function. The output of the program is shown in the terminal window, displaying the adjacency matrix and the DFS traversal starting from node 1.

```
19 printf("Enter the no of edges: ");
20 scanf("%d", &E);
21 printf("Enter the no of vertices: ");
22 scanf("%d", &V);
23
24 for (i = 0; i < V; i++) {
25     for (j = 0; j < V; j++) {
26         G[i][j] = 0;
27     }
28 }
29
30 for (i = 0; i < E; i++) {
31     scanf("%d %d", &v1, &v2);
32
33     v1--;
34     v2--;
35
36     G[v1][v2] = 1;
37     G[v2][v1] = 1;
38 }
39
40 // Print the adjacency matrix
41 printf("Adjacency Matrix:\n");
42 for (i = 0; i < V; i++) {
43     for (j = 0; j < V; j++) {
44         printf("%d ", G[i][j]);
45     }
46     printf("\n");
47 }
48
49 printf("Enter the source: ");
50 scanf("%d", &source);
51
52 for (i = 0; i < V; i++) {
53     visited[i] = 0;
54 }
55
56 printf("DFS traversal starting from node %d:\n", source);
57 DFS(source - 1);
58
59 return 0;
60 }
```

Output:

```

itl4@22DL407:~$ gedit exp9.c
^C
itl4@22DL407:~$ gcc exp9.c
itl4@22DL407:~$ ./a.out
                                Graphs
Enter the no of edges:8
Enter the no of vertices:9
Enter the edges (format: V1 V2) : 1 2
Enter the edges (format: V1 V2) : 8 3
Enter the edges (format: V1 V2) : 7 5
Enter the edges (format: V1 V2) : 1 4
Enter the edges (format: V1 V2) : 6
8
Enter the edges (format: V1 V2) : 1 6
Enter the edges (format: V1 V2) : 7 2
Enter the edges (format: V1 V2) : 1 0
  0  1  0  1  0  1  0  0  0
  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  1  0
  0  1  0  0  1  0  0  0  0
  0  0  1  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0
Enter the source: 7
itl4@22DL407:~$

```

Bfs:

```
1 #include <stdio.h>
2
3 int a[20][20], q[20], visited[20], n, f = -1, r = -1;
4 void bfs(int v) {
5     int i;
6     for (i = 0; i < n; i++) {
7         if (a[v][i] != 0 && visited[i] == 0) {
8             r = r + 1;
9             q[r] = i;
10            visited[i] = 1;
11            printf("%d ", i);
12        }
13    }
14    f = f + 1;
15    if (f <= r)
16        bfs(q[f]);
17 }
18 int main() {
19     int v, i, j;
20     printf("\nEnter number of vertices");
21     scanf("%d", &n);
22     for (i = 0; i < n; i++) {
23         visited[i] = 0;
24     }
25     printf("\nEnter graph data in matrix form\n");
26     for (i = 0; i < n; i++) {
27         for (j = 0; j < n; j++) {
28             scanf("%d", &a[i][j]);
29         }
30     }
31     printf("\nEnter the starting vertex");
32     scanf("%d", &v);
33     f = r = 0;
34     q[r] = v;
35     visited[v] = 1;
36     printf("%d ", v);
37     bfs(v);
38     if (r != n - 1) {
39         printf("\nBFS not possible\n");
40     }
41     return 0;
42 }
```

Output:

```
itl4@22DL407:~$ gedit exp9.1.c
^C
itl4@22DL407:~$ gcc exp9.1.c
itl4@22DL407:~$ ./a.out

Enter number of vertices5

Enter graph data in matrix form
0 1 0 0 1
1 0 1 1 1
0 1 0 1 0
0 1 1 0 1
1 1 0 1 0

Enter the starting vertex3
3 1 2 4 0
itl4@22DL407:~$ gedit exp9.1.c
```