

```
Start here X dfs.c X bfs.c X
1 #include <stdio.h>
2 #define MAX 20
3
4 int pushStack[MAX], popStack[MAX];
5 int top1 = -1, top2 = -1;
6 int visited[MAX];
7
8 void push1(int v) {
9     pushStack[++top1] = v;
10}
11 int pop1() {
12     if (top1 == -1) return -1;
13     return pushStack[top1--];
14}
15 void push2(int v) {
16     popStack[++top2] = v;
17}
18 int pop2() {
19     if (top2 == -1) return -1;
20     return popStack[top2--];
21}
22 int dequeue() {
23     int v;
24     if (top2 == -1) {
25         while (top1 != -1) {
26             push2(pop1());
27         }
28     }
29     return pop2();
30}
31 void BFS_Stacks(int adj[MAX][MAX], int n, int start) {
32     for (int i = 0; i < n; i++)
33         visited[i] = 0;
34     push1(start);
35     visited[start] = 1;
36     while (top1 != -1 || top2 != -1) {
37         int v = dequeue();
38         printf("%d ", v);
39
40         for (int i = 0; i < n; i++) {
41             if (adj[v][i] == 1 && !visited[i]) {
42                 push1(i);
43                 visited[i] = 1;
44             }
45         }
46     }
47 }
48 int main() {
49     int n, adj[MAX][MAX];
50     printf("Enter number of vertices: ");
51     scanf("%d", &n);
52     printf("Enter adjacency matrix:\n");
53     for (int i = 0; i < n; i++)
54         for (int j = 0; j < n; j++)
55             scanf("%d", &adj[i][j]);
56
57     printf("BFS Traversal: ");
58     BFS_Stacks(adj, n, 0);
59
60     return 0;
61 }
62
```

Logs & others

CSE-L4\Desktop\1bm24cs176.bfs.c C/C++ Windows (CR+LF) WINDOWS-1252 Line 41, Col 19, Pos 847 Insert Read/Write default ENG IN 11:40:02 08-12-2025

```
C:\Users\BMSCECSE-L4\Desktop + v X - □ ×
Enter number of vertices: 4
Enter adjacency matrix:
1 0 0 1
0 0 0 0
1 1 1 1
0 1 1 1
BFS Traversal: 0 3 1 2
Process returned 0 (0x0) execution time : 16.769 s
Press any key to continue.

for (int j = 0; j < n; j++)
```