

1. C code implementation for BFS

```
Online Compiler
A STAR ALLIANCE MEMBER
HOW Made of Switzerland

main.c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <stdbool.h>
4 #define MAX 100
5 int graph[MAX][MAX], visited[MAX], queue[MAX];
6 int front = 0, rear = 0;
7 void enqueue(int node) {
8     queue[rear++] = node;
9 }
10 int dequeue() {
11     return queue[front++];
12 }
13 bool isEmpty() {
14     return front == rear;
15 }
16 void bfs(int start, int n) {
17     enqueue(start);
18     visited[start] = 1;
19     while (!isEmpty()) {
20         int node = dequeue();
21         printf("%d ", node);
22         for (int i = 0; i < n; i++) {
23             if (graph[node][i] && !visited[i]) {
24                 enqueue(i);
25                 visited[i] = 1;
26             }
27         }
28     }
29 }
30 int main() {
31     int n, start;
32     printf("Enter the number of vertices: ");
33     scanf("%d", &n);
34     printf("Enter the adjacency matrix:\n");
35     for (int i = 0; i < n; i++)
36         for (int j = 0; j < n; j++)
37             scanf("%d", &graph[i][j]);
38     printf("Enter the starting vertex: ");
39     scanf("%d", &start);
40     for (int i = 0; i < n; i++) visited[i] = 0;
41     printf("BFS traversal: ");
42     bfs(start, n);
43     return 0;
44 }
```

```
Output
/tmp/IoaMsslnT9.o
Enter the number of vertices: 5
Enter the adjacency matrix:
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 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
BFS traversal: 0 1 2 3 4
--- Code Execution Successful ---
```

2. C code implementation for DFS

```
main.c
1 #include <stdio.h>
2 #define MAX 100
3 int graph[MAX][MAX]; // Adjacency matrix
4 int visited[MAX];
5 void dfs(int node, int n) {
6     printf("%d ", node);
7     visited[node] = 1;
8     for (int i = 0; i < n; i++) {
9         if (graph[node][i] && !visited[i]) {
10             dfs(i, n);
11         }
12     }
13 }
14 int main() {
15     int n, start;
16     printf("Enter the number of vertices: ");
17     scanf("%d", &n);
18     printf("Enter the adjacency matrix:\n");
19     for (int i = 0; i < n; i++)
20         for (int j = 0; j < n; j++)
21             scanf("%d", &graph[i][j]);
22     printf("Enter the starting vertex: ");
23     scanf("%d", &start);
24     if (start < 0 || start >= n) {
25         printf("Invalid starting vertex\n");
26         return -1;
27     }
28     for (int i = 0; i < n; i++) visited[i] = 0;
29     printf("DFS traversal starting from vertex %d: ", start);
30     dfs(start, n);
31     printf("\n");
32     return 0;
33 }
```

```
Output
/tmp/qevIA3W92n.o
Enter the number of vertices: 3
Enter the adjacency matrix:
0 1 3
0 2 5
0 6 9
Enter the starting vertex: 0
DFS traversal starting from vertex 0: 0 1 2
--- Code Execution Successful ---
```