

22. Write a C program to Graph traversal using Breadth First Search

```
#include <stdio.h>

#define MAX 20

int queue[MAX];
int front = -1, rear = -1;
int visited[MAX];
int adj[MAX][MAX];
int n;

void enqueue(int v) {
    if (rear == MAX - 1)
        printf("Queue Overflow\n");
    else {
        if (front == -1)
            front = 0;
        rear++;
        queue[rear] = v;
    }
}

int dequeue() {
    int v;
    if (front == -1 || front > rear)
        return -1;
    v = queue[front];
    front++;
    return v;
}

void bfs(int start) {
    int i, current;
    for (i = 0; i < n; i++)
        visited[i] = 0;
```

```

enqueue(start);
visited[start] = 1;
printf("BFS Traversal: ");
while (front <= rear) {
    current = dequeue();
    printf("%d ", current);
    for (i = 0; i < n; i++) {
        if (adj[current][i] == 1 && visited[i] == 0) {
            enqueue(i);
            visited[i] = 1;
        }
    }
}
printf("\n");
}

int main() {
    int i, j, start;
    printf("Enter the number of vertices: ");
    scanf("%d", &n);
    printf("Enter the adjacency matrix:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            scanf("%d", &adj[i][j]);
        }
    }
    printf("Enter the starting vertex (0 to %d): ", n - 1);
    scanf("%d", &start);
    bfs(start);
    return 0;
}

```

main.c

Share

Run

```
1 #include <stdio.h>
2 #define MAX 20
3 int queue[MAX];
4 int front = -1, rear = -1;
5 int visited[MAX];
6 int adj[MAX][MAX];
7 int n;
8
9 void enqueue(int v) {
10     if (rear == MAX - 1)
11         printf("Queue Overflow\n");
12     else {
13         if (front == -1)
14             front = 0;
15         rear++;
16         queue[rear] = v;
17     }
18 }
19
20 int dequeue() {
21     int v;
22     if (front == -1 || front > rear)
23         return -1;
24     v = queue[front];
25     front++;
26     return v;
27 }
28
```

Output

Enter the number of vertices: 5
Enter the adjacency matrix:
1 0 2 1 4
1 5 6 2 3
4 5 2 3 8
7 8 5 6 4
2 3 5 4 6
Enter the starting vertex (0 to 4): 3
BFS Traversal: 3

=== Code Execution Successful ===