EXERCISE-21

AIM: To write a C program to implement Breadth-First Search (BFS) traversal of a graph using an adjacency matrix.

Algorithm:

1.Start.

2. Represent the graph using an adjacency matrix.

3. Initialize a visited array to mark visited nodes.

4. Use a queue to keep track of nodes to visit.

5. Enqueue the starting vertex, mark it as visited.

6. While the queue is not empty:

- Dequeue a vertex and print it.
- Enqueue all its adjacent unvisited vertices and mark them as visited.

7. End.

Program Code:

```
#include <stdio.h>

#include <stdlib.h>

#define MAX 20

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

int visited[MAX];

void enqueue(int vertex) {

   if (rear == MAX - 1)

      printf("Queue Overflow\n");
```

```
    else {
        if (front == -1)
            front = 0;
        queue[++rear] = vertex;
    }
}
int dequeue() {
    if (front == -1 || front > rear)
        return -1;
    return queue[front++];
}
void bfs(int adj[MAX][MAX], int n, int start) {
    int i, vertex;
    enqueue(start);
    visited[start] = 1;
    printf("BFS Traversal: ");
    while (front <= rear) {
        vertex = dequeue();
        printf("%d ", vertex);
        for (i = 0; i < n; i++) {
            if (adj[vertex][i] == 1 && !visited[i]) {
                enqueue(i);
                visited[i] = 1;
```

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

Input and Output:

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

Result:

The program successfully performs a Breadth-First Search (BFS) traversal of the given graph using an adjacency matrix.