```
#include <stdio.h>
#define MAX 10

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

// Function to insert element into queue
void enqueue(int vertex) {
    if (rear == MAX - 1)
        return;
    if (front == -1)
        front = 0;
    queue[++rear] = vertex;
}

// Function to delete element from queue
int dequeue() {
    if (front == -1 || front > rear)
        return -1;
    return queue[front++];
}

// Breadth First Search Function
void BFS(int adj[MAX][MAX], int n, int start) {
    int i, vertex;
    enqueue(start);
    visited[start] = 1;
```

```
    printf("BFS Traversal: ");
    while ((vertex = dequeue()) != -1) {
        printf("%d ", vertex);
        for (i = 0; i < n; i++) {
            if (adj[vertex][i] == 1 && visited[i] == 0) {
                enqueue(i);
                visited[i] = 1;
            }
        }
    }
}

int main() {
    int n, i, j, start;
    int adj[MAX][MAX];

    printf("Enter number of vertices: ");
    scanf("%d", &n);

    printf("Enter adjacency matrix:\n");
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
            scanf("%d", &adj[i][j]);

    printf("Enter starting vertex (0 to %d): ", n - 1);
    scanf("%d", &start);

    for (i = 0; i < n; i++)
```

```
    visited[i] = 0;


  BFS(adj, n, start);

  return 0;
}
```