```c
#include <stdio.h>
#define SIZE 20
int queue[SIZE], front = -1, rear = -1;
int visited[SIZE];
void enqueue(int vertex) {
    if (rear == SIZE - 1)
        return;
    if (front == -1)
        front = 0;
    queue[++rear] = vertex;
}
int dequeue() {
    if (front == -1 || front > rear)
        return -1;
    return queue[front++];
}
void BFS(int adjMatrix[SIZE][SIZE], int n, int start) {
    for (int i = 0; i < n; i++)
        visited[i] = 0;
    enqueue(start);
    visited[start] = 1;
    printf("BFS Traversal starting from vertex %d: ", start);
    while (front <= rear) {
        int current = dequeue();
        printf("%d ", current);
        for (int i = 0; i < n; i++) {
            if (adjMatrix[current][i] == 1 && !visited[i]) {
                enqueue(i);
                visited[i] = 1;
            }
        }
    }
    printf("\n");
}
int main() {
    int adjMatrix[SIZE][SIZE], n, start;
    printf("Enter number of vertices: ");
    scanf("%d", &n);
    printf("Enter adjacency matrix:\n");
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            scanf("%d", &adjMatrix[i][j]);
    printf("Enter starting vertex for BFS (0 to %d): ", n - 1);
    scanf("%d", &start);
```

```
    BFS(adjMatrix, n, start);
    return 0;
}
```