```c
#include<stdio.h>
#include<stdlib.h>
#define MAX 50
void bfs(int graph[MAX][MAX],int visited[MAX],int start,int n);
int main()
{
    int n;
    int graph[MAX][MAX];
    int visited[MAX]={0};
    int start;
    printf("Enter the number of vertices: ");
    scanf("%d",&n);
    printf("Enter the adjacency matrix of the graph:\n");
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            scanf("%d",&graph[i][j]);
        }
    }
    printf("Enter the starting node (0 to %d): ",n-1);
    scanf("%d",&start);
    printf("BFS traversal starting from node %d:\n",start);
    bfs(graph,visited,start,n);
    return 0;
}
void bfs(int graph[MAX][MAX],int visited[MAX],int start,int n)
{
    int queue[MAX],front=0,rear=0;
    queue[rear++]=start;
    visited[start]=1;
    while(front<rear)
    {
        int current=queue[front++];
        printf("%d",current);
        for(int i=0;i<n;i++)
        {
            if(graph[current][i]==1 &&!visited[i])
            {
```

```c
    printf("Enter the starting node (0 to %d): ",n-1);
    scanf("%d",&start);
    printf("BFS traversal starting from node %d:\n",start);
    bfs(graph,visited,start,n);
    return 0;
}
void bfs(int graph[MAX][MAX],int visited[MAX],int start,int n)
{
    int queue[MAX],front=0,rear=0;
    queue[rear++]=start;
    visited[start]=1;
    while(front<rear)
    {
        int current=queue[front++];
        printf("%d",current);
        for(int i=0;i<n;i++)
        {
            if(graph[current][i]==1 &&!visited[i])
            {
                queue[rear++]=i;
                visited[i]=1;
            }
        }
    }
}
```

```
PS C:\Users\STUDENT\Downloads\canteen\style> cd "c:\Users\STUDENT\Down
Enter the number of vertices: 3
Enter the adjacency matrix of the graph:
0 0 1
1 0 0
1 0 1
Enter the starting node (0 to 2): 1
BFS traversal starting from node 1:
102
PS C:\Users\STUDENT\Downloads\canteen\style>
```