```c
/* Design, Develop and Implement a Program in C for the
following operations on Graph(G) of Cities
a. Create a Graph of N cities using Adjacency Matrix.
b. Print all the nodes reachable from a given starting node in a digraph
using DFS/BFS method */

#include<stdio.h>
#include<stdlib.h>

int a[50][50], n, visited[50];
int q[20], front = -1,rear = -1;
int s[20], top = -1, count=0;

void bfs(int v)
{
   int i, cur;
   visited[v] = 1;
   q[++rear] = v;
   while(front!=rear)
   {
     cur = q[++front];
     for(i=1;i<=n;i++)
     {
       if((a[cur][i]==1)&&(visited[i]==0))
        {
          q[++rear] = i;
          visited[i] = 1;
          printf("%d ", i);
        }
     }
   }
}

void dfs(int v)
{
  int i;
  visited[v]=1;
  s[++top] = v;
  for(i=1;i<=n;i++)
   {
     if(a[v][i] == 1&& visited[i] == 0 )
     {
       printf("%d ", i);
       dfs(i);
     }
   }
}

void main()
{

  int ch, start, i,j;
  printf("\nEnter the number of vertices in graph:  ");
  scanf("%d",&n);
  printf("\nEnter the adjacency matrix:\n");
  for(i=1; i<=n; i++)
  {
    for(j=1;j<=n;j++)
      scanf("%d",&a[i][j]);
  }

  for(i=1;i<=n;i++)
   visited[i]=0;
   printf("\nEnter the starting vertex: ");
   scanf("%d",&start);

   printf("\n1. BFS: Print all nodes reachable from a given starting
node");
   printf("\n2. DFS: Print all nodes reachable from a given starting
node");
   printf("\n3: Exit");
   printf("\nEnter your choice: ");
   scanf("%d", &ch);
   switch(ch)
   {
     case 1: printf("\nNodes reachable from starting vertex %d are: ",
start);
             bfs(start);
             for(i=1;i<=n;i++)
             {
                if(visited[i]==0)
                printf("\nThe vertex that is not reachable is %d" ,i);
             }
             break;


     case 2: printf("\nNodes reachable from starting vertex %d
are:\n",start);
             dfs(start);
             break;
     case 3: exit(0);
     default: printf("\nPlease enter valid choice:");
   }
}
```

Output:

```
Case 1:
Enter the number of vertices in graph:  4
Enter the adjacency matrix:
0          1          0          1
0          0          1          0
0          0          0          1
0          0          0          0


~~~Menu~~~~
```

```
1. BFS: Print all nodes reachable from a given starting node
2. DFS: Print all nodes reachable from a given starting node
3: Exit
Enter your choice: 1
Enter the starting vertex: 1
Nodes reachable from starting vertex 1 are: 2       4          3

Case 2:
Enter the number of vertices in graph:  4
Enter the adjacency matrix:
0          1          0          1
0          0          1          0
0          0          0          1
0          0          0          0
~~~Menu~~~~
1. BFS: Print all nodes reachable from a given starting node
2. DFS: Print all nodes reachable from a given starting node
3: Exit
Enter your choice: 1
Enter the starting vertex: 2
Nodes reachable from starting vertex 2 are:  3      4
The vertex that is not reachable is 1

Case 3:
Enter the number of vertices in graph:  4
Enter the adjacency matrix:
0          1          0          1
0          0          1          0
0          0          0          1
0          0          0          0
~~~Menu~~~~
1. BFS: Print all nodes reachable from a given starting node
2. DFS: Print all nodes reachable from a given starting node
3: Exit
Enter your choice: 2
Enter the starting vertex:     1
Nodes reachable from starting vertex 1 are:  2       3       4

Case 4:
Enter the number of vertices in graph:  4
Enter the adjacency matrix:
0          1          0          1
0          0          1          0
0          0          0          1
0          0          0          0
~~~Menu~~~~
1. BFS: Print all nodes reachable from a given starting node
2. DFS: Print all nodes reachable from a given starting node
3: Exit
Enter your choice: 2
Enter the starting vertex:     2
Nodes reachable from starting vertex 2 are:   3       4
```