```c
#include<stdio.h>
#include<stdlib.h>
void dfs(int vertex,int visit[],int graph[][100],int n)
{
    visit[vertex]=1;
    for(int i=0;i<n;i++)
    {
        if(graph[vertex][i]==1 && !visit[i])
        {
            dfs(i,visit,graph,n);
        }
    }
}
int main()
{
    int n;
    int graph[100][100];
    int visit[100]={0};
    printf("Enter the number of vertices:\n");
    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]);
        }
    }
    dfs(0,visit,graph,n);
    int isConnected=1;
    for(int i=0;i<n;i++)
    {
        if(!visit[i])
        {
            isConnected=0;
            break;
        }
    }
    if(isConnected)
```

```c
    printf("Enter the number of vertices:\n");
    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]);
        }
    }
    dfs(0,visit,graph,n);
    int isConnected=1;
    for(int i=0;i<n;i++)
    {
        if(!visit[i])
        {
            isConnected=0;
            break;
        }
    }
    if(isConnected)
    {
        printf("The graph is connected\n");
    }
    else
    {
        printf("The graph is not connected\n");
    }
    return 0;
}
```

```
PS C:\Users\STUDENT\Downloads\canteen\style> cd "c:\Users\STUDENT\Download
Enter the number of vertices:
3
Enter the adjacency matrix of the graph:
1 1 1
0 0 1
0 0 0
The graph is connected
PS C:\Users\STUDENT\Downloads\canteen\dfs> cd "c:\Users\STUDENT\Downloads\
Enter the number of vertices:
2
Enter the adjacency matrix of the graph:
0 0
1 1
The graph is not connected
PS C:\Users\STUDENT\Downloads\canteen\dfs>
```