```c
#include <stdio.h>
#include <process.h>
#include <stdlib.h>

void dfs(int);
int a[10][10],vis[10],n;
int main()
{
 int i,j,src,choice;
 int count=0;
 while(1)
 {
  printf("\n1.Connected and Disconnected Graph\n2.Exit\n");
  printf("Enter the Choice :");
  scanf("%d",&choice);
  switch(choice)
 {
  case 1: printf("Enter the No: of Vertices :");
          scanf("%d",&n);
          printf("Enter 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++)
         vis[i]=0;

         printf("Enter the Source Vertex :");
         scanf("%d",&src);
         printf("\nThe Path from Vertex %d is : \n",src);
         dfs(src);


        for(i=1;i<=n;i++)
       {
```

```
        for(i=1;i<=n;i++)
       {
        if(vis[i])
        count++;
       }
       if(count==n)
       {
        printf("\nGraph is connected");
       }
       else
       {
        printf("\nGraph is not connected");
       }
       break;

  case 2: exit(0);
          break;
  default: exit(0);
 }
}
return 0;
}


void dfs(int v)
{
 int i;
 vis[v]=1;

 for(i=1;i<=n;i++)
 {
  if(a[v][i]==1 && vis[i]==0)
  {
   printf("\n%d to %d\n",v,i);
   dfs(i);
  }
 }
}
```

C:\WINDOWS\SYSTEM32\cmd.exe

```
1.Connected and Disconnected Graph
2.Exit
Enter the Choice :1
Enter the No: of Vertices :5
Enter the Adjacency Matrix :
0 0 1 1 1
0 0 0 1 1
1 0 0 1 0
1 1 1 0 0
1 1 0 0 0
Enter the Source Vertex :1

The Path from Vertex 1 is :

1 to 3

3 to 4

4 to 2

2 to 5

Graph is connected
1.Connected and Disconnected Graph
2.Exit
Enter the Choice :1
Enter the No: of Vertices :4
Enter the Adjacency Matrix :
0 1 0 0
1 0 0 0
0 0 0 1
0 0 1 0
Enter the Source Vertex :2

The Path from Vertex 2 is :

2 to 1

Graph is not connected
1.Connected and Disconnected Graph
2.Exit
```

C:\WINDOWS\SYSTEM32\cmd.exe

```
0 0 1 1 1
0 0 0 1 1
1 0 0 1 0
1 1 1 0 0
1 1 0 0 0
Enter the Source Vertex :1

The Path from Vertex 1 is :

1 to 3

3 to 4

4 to 2

2 to 5

Graph is connected
1.Connected and Disconnected Graph
2.Exit
Enter the Choice :1
Enter the No: of Vertices :4
Enter the Adjacency Matrix :
0 1 0 0
1 0 0 0
0 0 0 1
0 0 1 0
Enter the Source Vertex :2

The Path from Vertex 2 is :

2 to 1

Graph is not connected
1.Connected and Disconnected Graph
2.Exit
Enter the Choice :2


------------------
(program exited with code: 0)
```