input:

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

/*   ADJACENCY MATRIX    */

int source,V,E,time,visited[20],G[20][20];
void DFS(int i)
{
 int j;
 visited[i]=1;
 printf(" %d->",i+1);
 for(j=0;j<V;j++)
 {
  if(G[i][j]==1 && visited[j]==0)
  {
   DFS(j);
  }
 }
}

int main()
{
 int i,j,v1,v2;
 printf("\t\t\tGraphs\n");
 printf("Enter the no. of Edges : ");
 scanf("%d",&E);
 printf("Enter the no. of vertices : ");
 scanf("%d",&V);
 for(i=0;i<V;i++)
 {
  for(j=0;j<V;j++)
  {
   G[i][j]=0;
  }
 }

 /* Creating Edges:P   */
 for(i=0;i<E;i++)
 {
  printf("Enter the Edges (format: V1 V2) : ");
  scanf("%d%d",&v1,&v2);
  G[v1-1][v2-1]=1;
 }

 for(i=0;i<V;i++)
 {
  for(j=0;j<V;j++)
  {
   printf(" %d\t",G[i][j]);

  }
  printf("\n");
 }
 printf("Enter the Source : ");
 scanf("%d",&source);
 DFS(source-1);
 return 0;
}
```

output:

```
                        Graphs
Enter the no. of Edges : 8
Enter the no. of vertices : 9
Enter the Edges (format: V1 V2) : 1 2
Enter the Edges (format: V1 V2) : 8 3
Enter the Edges (format: V1 V2) : 7 5
Enter the Edges (format: V1 V2) : 1 4
Enter the Edges (format: V1 V2) : 6 8
Enter the Edges (format: V1 V2) : 1 6
Enter the Edges (format: V1 V2) : 7 2
Enter the Edges (format: V1 V2) : 1 0
 0        1        0        1        0        1        0        0        0
 0        0        0        0        0        0        0        0        0
 0        0        0        0        0        0        0        0        0
 0        0        0        0        0        0        0        0        0
 0        0        0        0        0        0        0        0        0
 0        0        0        0        0        0        0        1        0
 0        1        0        0        1        0        0        0        0
 0        0        1        0        0        0        0        0        0
 0        0        0        0        0        0        0        0        0
Enter the Source : 7
7-> 2-> 5->student@dl405-HP-ProDesk-400-G7-Microtower-PC:~$ gedit Exp_7.c
```