ADA LAB TEST 1 1BM19CS086 MITHIL RAJ

4) Write program to do the following:

a) Print all the nodes reachable from a given starting node in a digraph using BFS method.

b) Check whether a given graph is connected or not using DFS method

A)

```c
#include<stdio.h>
void bfs(int);
int a[10][10],vis[10], n;

void main(){
 int i,j,src;
 printf("\nenter the no of vertices:\t");
 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]);
  }
 }
 printf("\nenter the source node:\t");
 scanf("%d",&src);
 printf("nodes reachable from %d vertex is ",src);
 bfs(src);
 getch();
}

void bfs(int v){

int q[10],f=0,r=0,u,i,j;
vis[v]=1;
q[r]=v;

while(f<=r){
```

```
 u=q[f];
 printf("%d\t",u);
 for(i=1;i<=n;i++){
  if(a[u][i]==1&&vis[i]==0){
  vis[i]=1;
  r=r+1;
  q[r]=i;
  }
 }
 f=f+1;
 }
}
```

OUTPUT:-

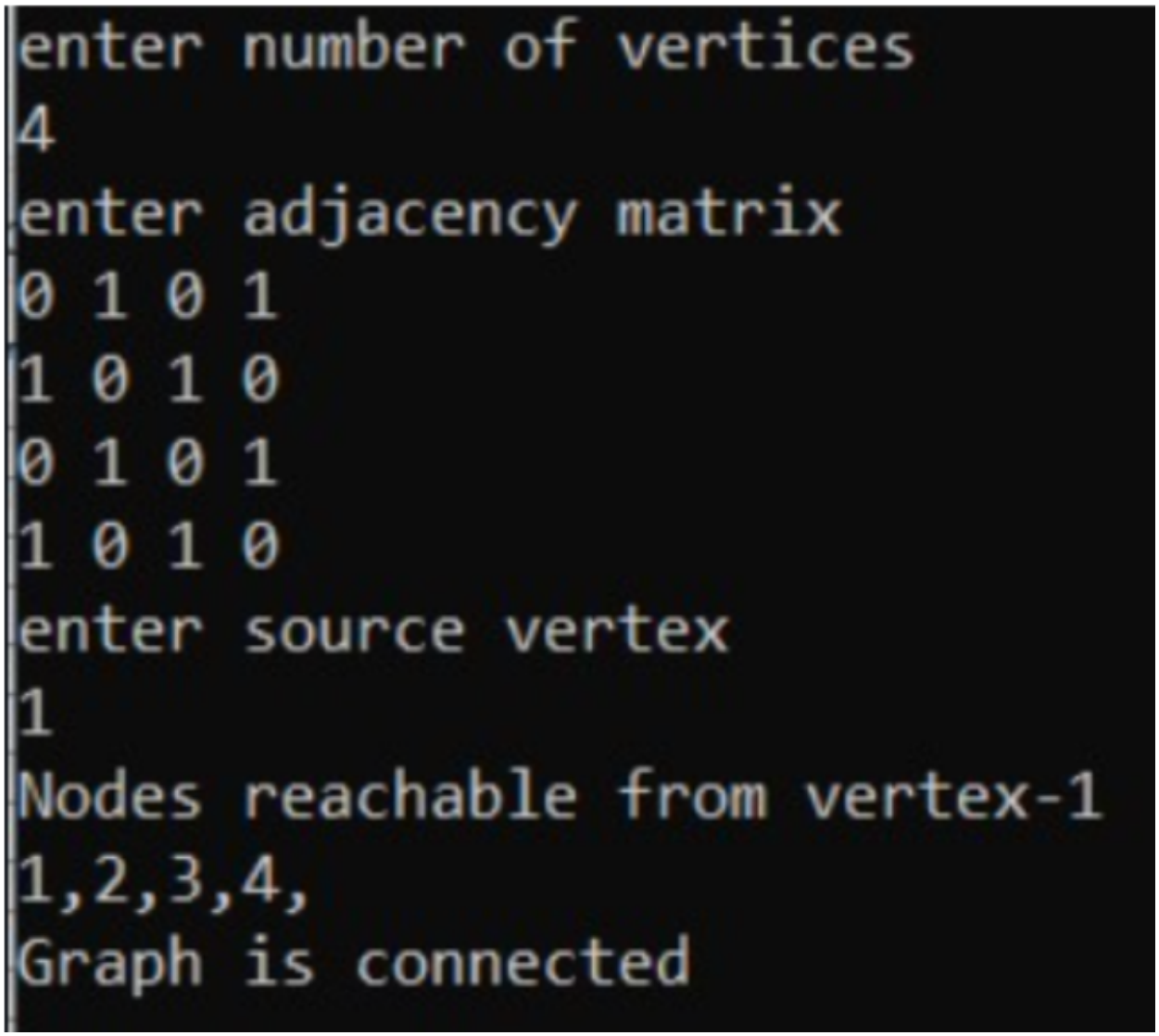
```
enter the no of vertices:       4

enter the adjacency matrix:
0 1 0 1
1 0 1 0
0 1 0 1
1 0 1 0

enter the source node:  1
nodes reachable from 1 vertex is 1      2       4       3

...Program finished with exit code 0
Press ENTER to exit console.
```

B)

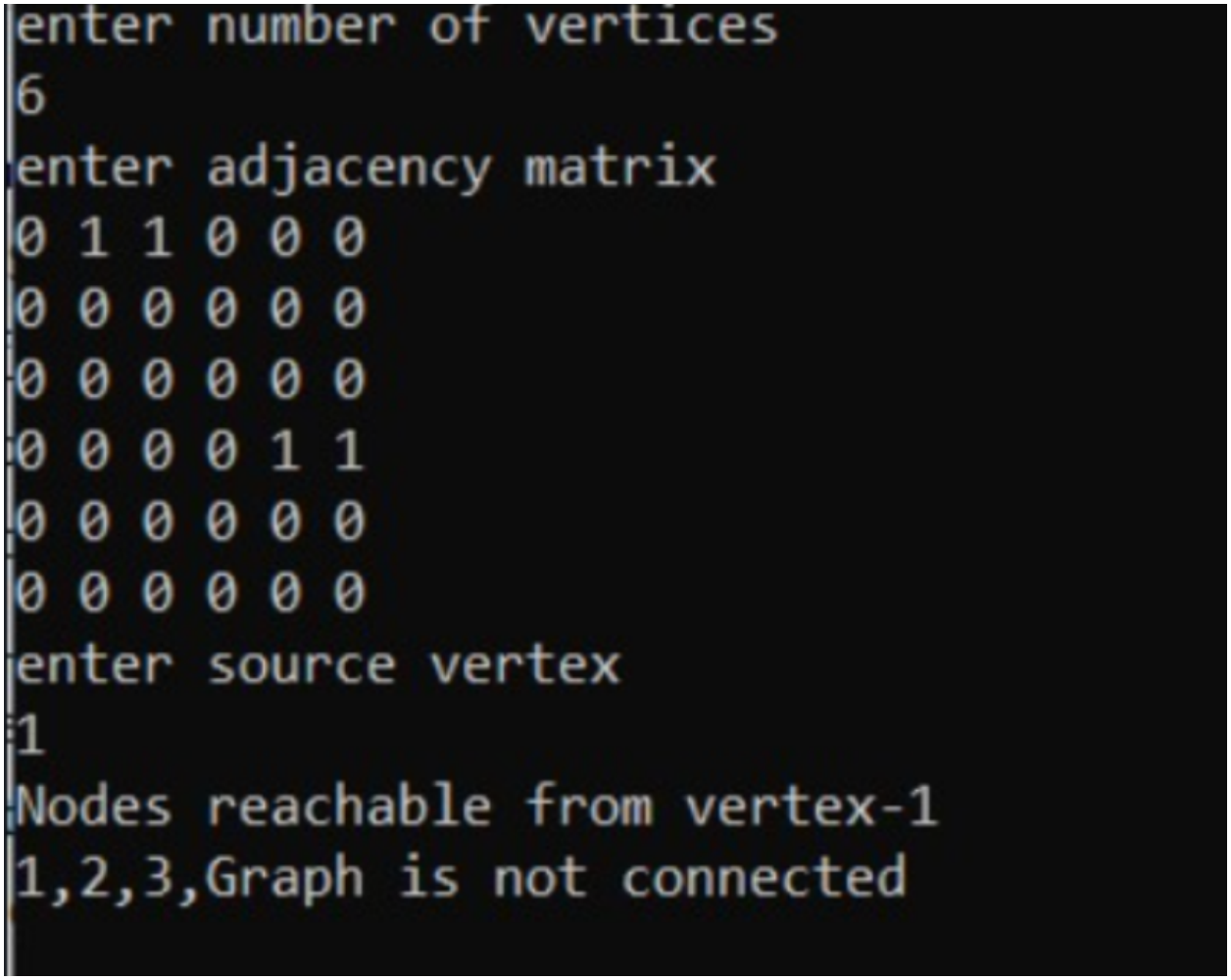
```c
#include<stdio.h>
#include<conio.h>
int a[20][20],vis[20],n;
void dfs(int v)
{
int i;
vis[v]=1;
for(i=1;i<=n;i++)
if(a[v][i] && !vis[i])
{

dfs(i);
}
}
void main()
{
int i,j,count=0;

printf("\n Enter number of vertices:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
vis[i]=0;
for(j=1;j<=n;j++)
a[i][j]=0;
}
printf("\n Enter the adjacency matrix:\n");
for(i=1;i<=n;i++)
```

```
for(j=1;j<=n;j++)
scanf("%d",&a[i][j]);
dfs(1);
printf("\n");
for(i=1;i<=n;i++)
{
if(vis[i])
count++;
}
if(count==n)
printf("\n Graph is connected");
else
printf("\n Graph is not connected");
getch();
}
```

OUTPUT FOR GRAPH CONNECTED :-



OUTPUT FOR GRAPH NOT CONNECTED :-



MODIFICATION PROGRAM

:-BFS, given an undirected graph, print all connected components line by Line.

```c
#include<stdio.h>
int a[20][20],reach[20],n;
void bfs(int v)
{
int i;
reach[v]=1;
for(i=1;i<=n;i++)
if(a[v][i] && !reach[i])
{
printf("\n %d->%d",v,i);
bfs(i);
}
}
void main()
{
int i,j,count=0;
printf("\n Enter number of vertices:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
reach[i]=0;
for(j=1;j<=n;j++)
a[i][j]=0;
}
printf("\n Enter the adjacency matrix:\n");
for(i=1;i<=n;i++)
for(j=1;j<=n;j++)
scanf("%d",&a[i][j]);
```

```
printf("\n Enter the adjacency matrix:\n");
for(i=1;i<=n;i++)
for(j=1;j<=n;j++)
scanf("%d",&a[i][j]);
bfs(1);
printf("\n");
for(i=1;i<=n;i++)
{
if(reach[i])
count++;
}
if(count==n)
printf("\n Graph is connected");
else
printf("\n Graph is not connected");
getch();
}
```

OUTPUT FOR DIRECT AND CONNECTED GRAPH:-

```
Enter number of vertices: 4

Enter the adjacency matrix:
0 1 1 1
0 0 0 1
0 0 0 0
0 0 1 0

1->2
2->4
4->3

Graph is connected

...Program finished with exit code 0
Press ENTER to exit console.
```

OUTPUT FOR GRAPH NOT CONNECTED UNDIRECT GRAPH:-

```
Enter number of vertices:6

Enter the adjacency matrix:
0 1 1 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 1 1
0 0 0 0 0 0
0 0 0 0 0 0

1->2
1->3

Graph is not connected

...Program finished with exit code 0
Press ENTER to exit console.
```

----------X----------