**ADA LAB-4**

* **DFS**
  + **PROGRAM**

#include<stdio.h>

#include<conio.h>

int a[1][10];

void dfs(int n, int cost[10][10], int u, int s[])

{

int v;

s[u]=1;

for(v=0;v<n;v++)

{

if((cost[u][v]==1)&&(s[v]==0))

dfs(n,cost,v,s);

}

}

void main()

{

int n,i,j,cost[10][10],s[10],con,flag;

printf("Enter the number of nodes\n");

scanf("%d", &n);

printf("Enter the adjacency matrix\n");

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

scanf("%d", &cost[i][j]);

}

con=0;

for(j=0;j<n;j++)

{

for(i=0;i<n;i++)

s[i]=0;

dfs(n,cost,j,s);

flag=0;

for(i=0;i<n;i++)

{

if(s[i]==0)

flag=1;

}

if(flag==0)

con=1;

}

if(con==1)

printf("Graph is connected\n");

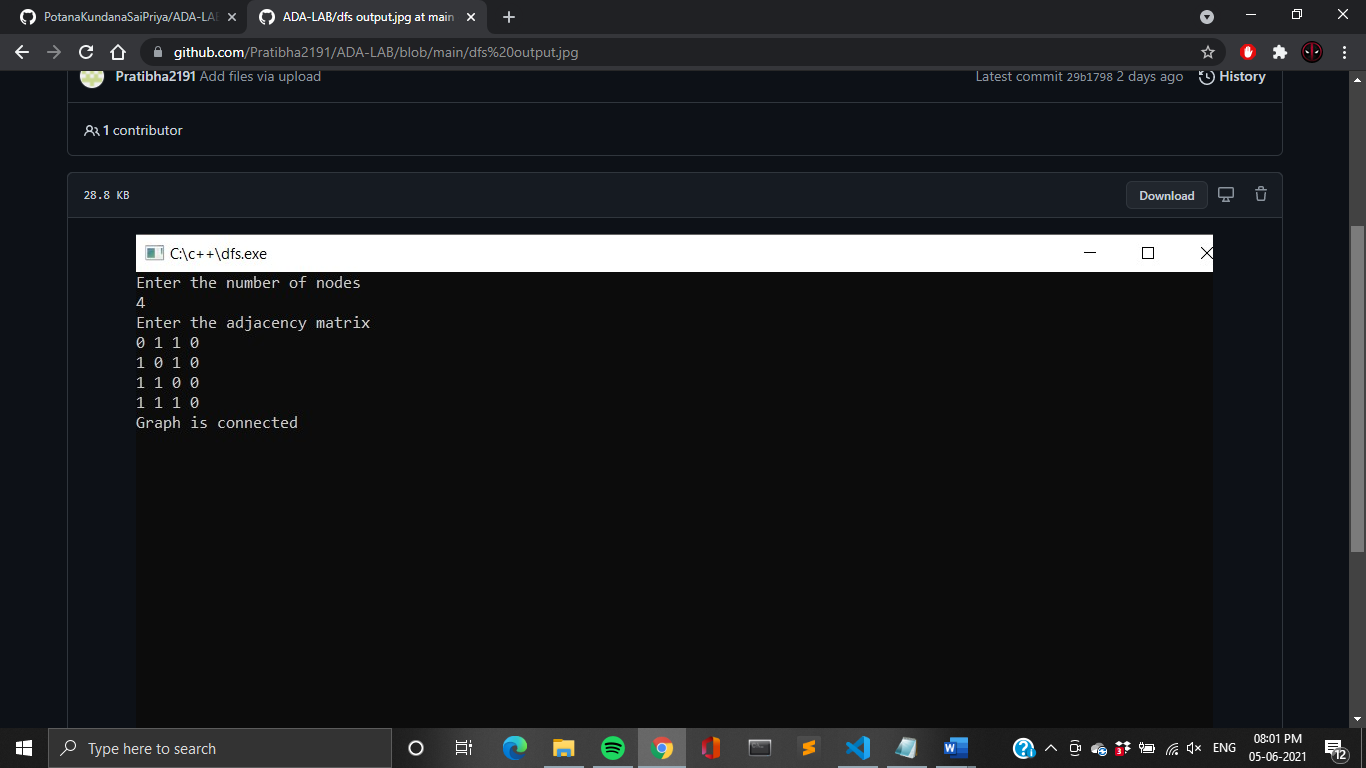
else

printf("Graph is not connected\n");

getch();

}

* + **OUTPUT**



* **BFS**
  + **Program**

#include<stdio.h>

#include<conio.h>

void insertq(int q[],int node, int \*f, int \*r)

{

if((\*f==-1) && (\*r==-1))

{

(\*f)++, (\*r)++, q[\*f]=node;

}

else{

(\*r)++, q[\*r]=node;

}

}

int deleteq(int q[],int \*f,int \*r)

{

int temp;

temp=q[\*f];

if(\*f == \*r) \*f=\*r=-1;

else (\*f)++;

return temp;

}

void bfs(int n, int adj[][20],int src, int visited[])

{

int q[30], f=-1,r=-1,v,i;

insertq(q,src,&f,&r);

while((f <=r ) && (f != -1))

{

v=deleteq(q,&f,&r);

if(visited[v]!=1)

{

visited[v]=1;

printf("%d\t",v);

}

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

if((adj[v][i]==1) && (visited[i] !=1))

insertq(q,i,&f,&r);

}

}

void main()

{

int n,i,j,adj[20][20],src,visited[20];

printf("enter number of vertices\n");

scanf("%d",&n);

printf("Enter adjacency matrix\n");

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

{

visited[i]=0;

for(j=1;j<=n;j++)

scanf("%d",&adj[i][j]);

}

printf("enter starting vertex\n");

scanf("%d",&src);

printf("The nodes reachable from src are\n");

bfs(n,adj,src,visited);

getch();

}

* + **Output**

