ANAGHA ACHARYA

1BM19CS224

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. #include<stdio.h>

#include<conio.h>

#include<time.h>

int a[20][20],q[20],visit[20],n,i,j,f=0,r=-1;

void bfs(int v)

{

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

if(a[v][i] && !visit[i])

q[++r]=i;

if(f<=r)

{

visit[q[f]]=1;

bfs(q[f++]);

}

}

void main()

{

int v;

clock\_t start,end;

double time;

printf("Enter the number of vertices:\n");

scanf("%d",&n);

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

{

q[i]=0;

visit[i]=0;

}

printf("Enter the graph data as matrix:\n");

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

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

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

printf("Enter the starting vertex:\n");

scanf("%d",&v);

start=clock();

bfs(v);

end=clock();

time=((double)(end-start))/CLOCKS\_PER\_SEC;

printf("The nodes reachable are:\n");

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

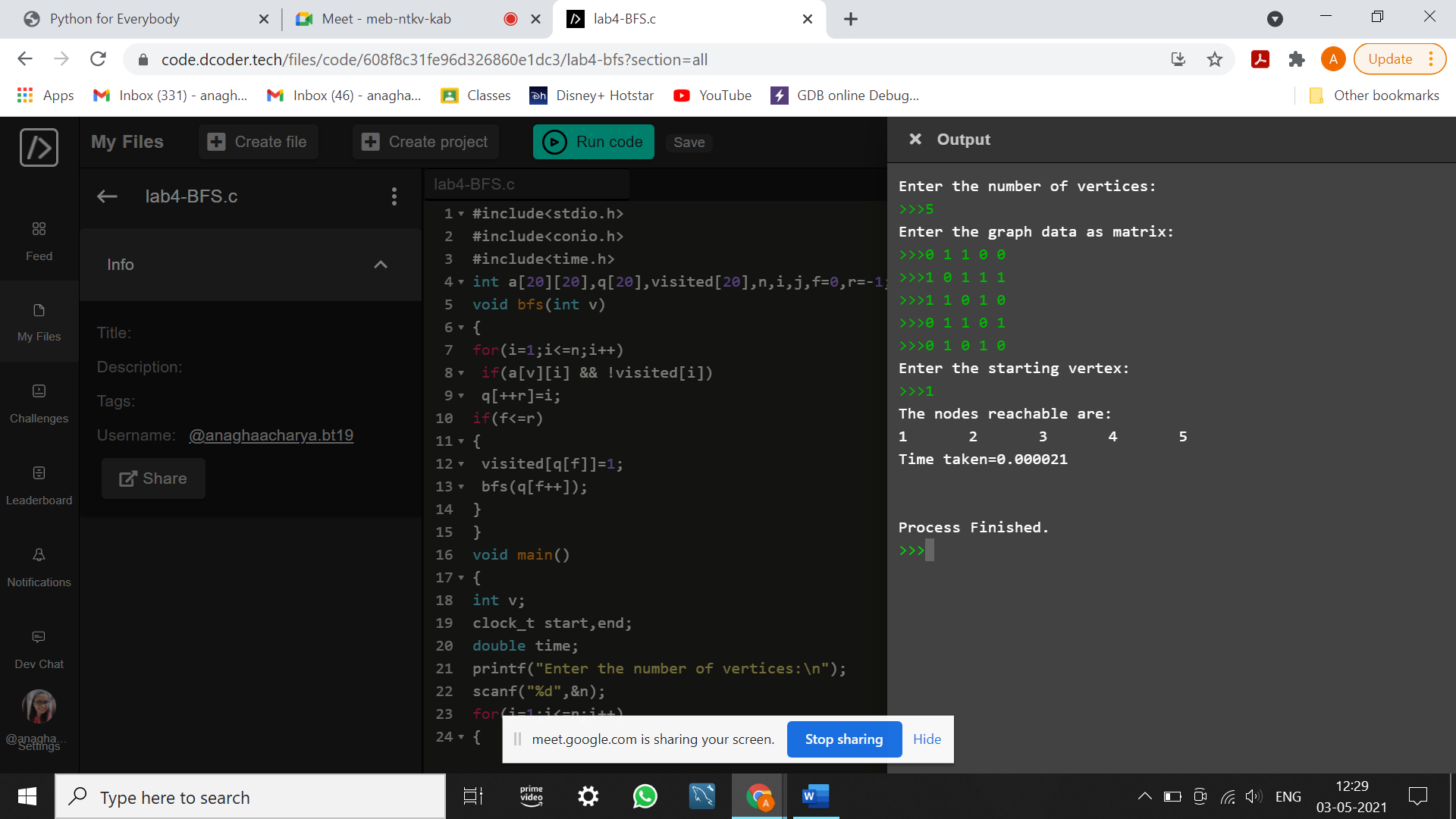
if(visit[i])

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

printf("\nTime taken=%1f\n",time);

}

OUTPUT



b. #include<stdio.h>

#include<conio.h>

#include<time.h>

int a[20][20],reach[20],n;

void dfs(int v){

int i;

reach[v]=1;

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

if(a[v][i] && !reach[i])

{

printf("%d->%d\n",v,i);

dfs(i);

}

}

void main()

{

int i,j,count=0;

clock\_t start,end;

float time;

printf("Enter number of vertices:\n");

scanf("%d",&n);

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

{

reach[i]=0;

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

a[i][j]=0;

}

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

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

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

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

start=clock();

dfs(1);

end=clock();

time=((double)(end-start)/CLOCKS\_PER\_SEC);

printf("\n");

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

{

if(reach[i])

count++;

}

if(count==n)

printf("Graph is connected");

else

printf("Graph is not connected");

printf("\nTime taken=%1f\n",time);

}

OUTPUT:

