**GREEN UNIVERSITY OF BANGLADESH**

**Date of Submiss n**

10/11/2021

## Course title :

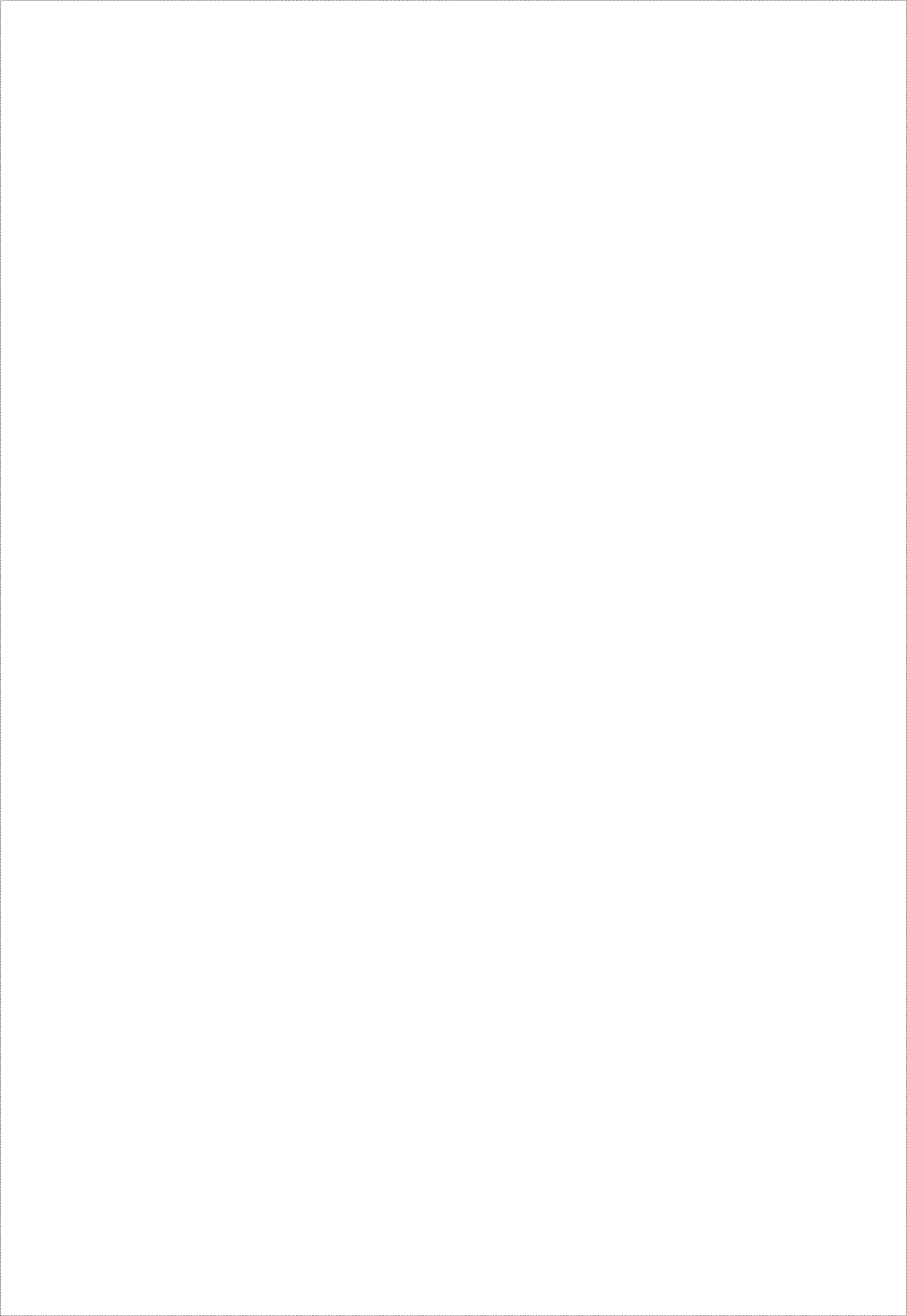
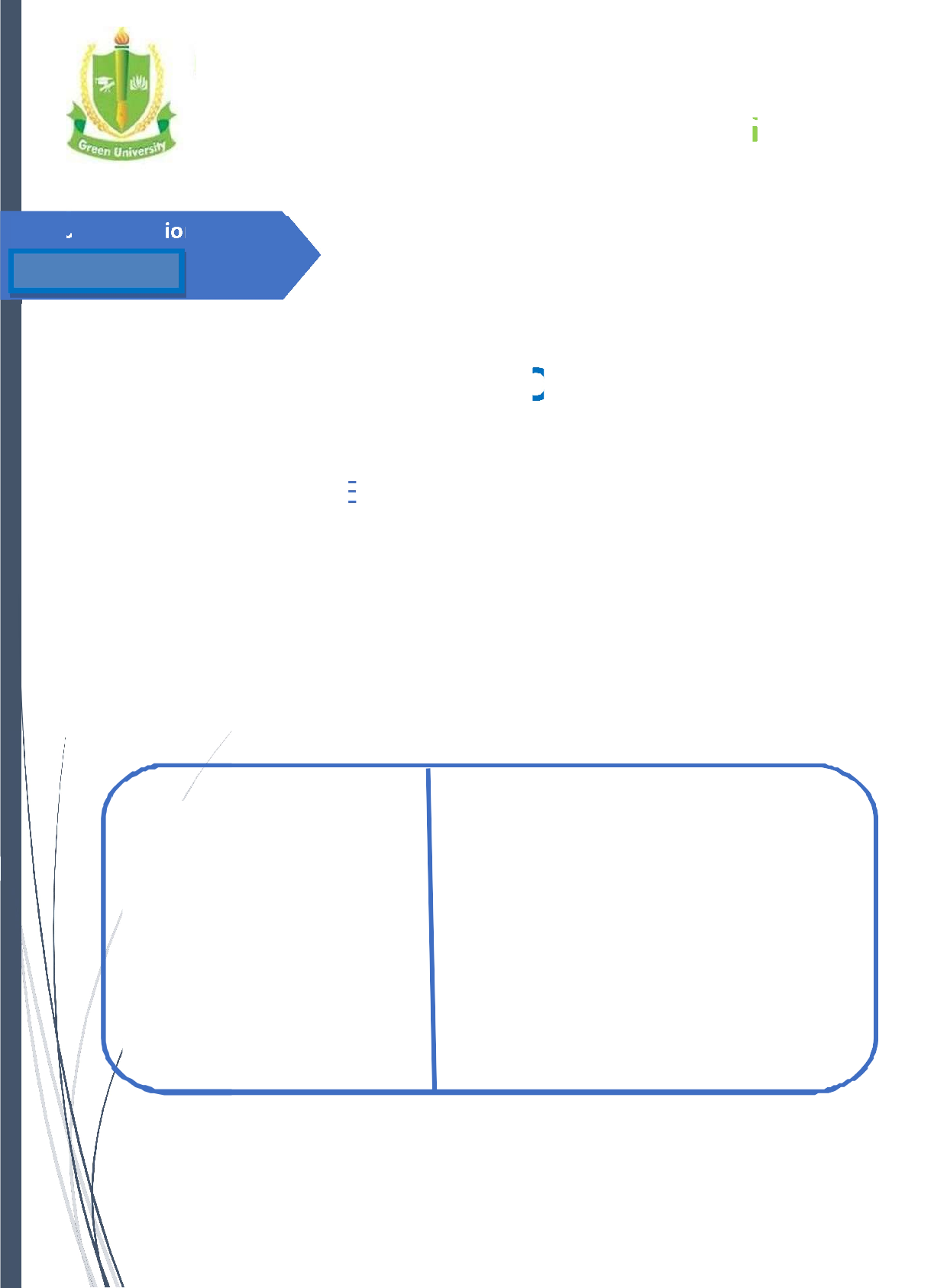
LAB REPORT - 4

## Algorithms Lab

Course Code: CSE - 206

## Date of signature:

#### Submitted by



**Name: Jahid Hasan**

ID: 201015069

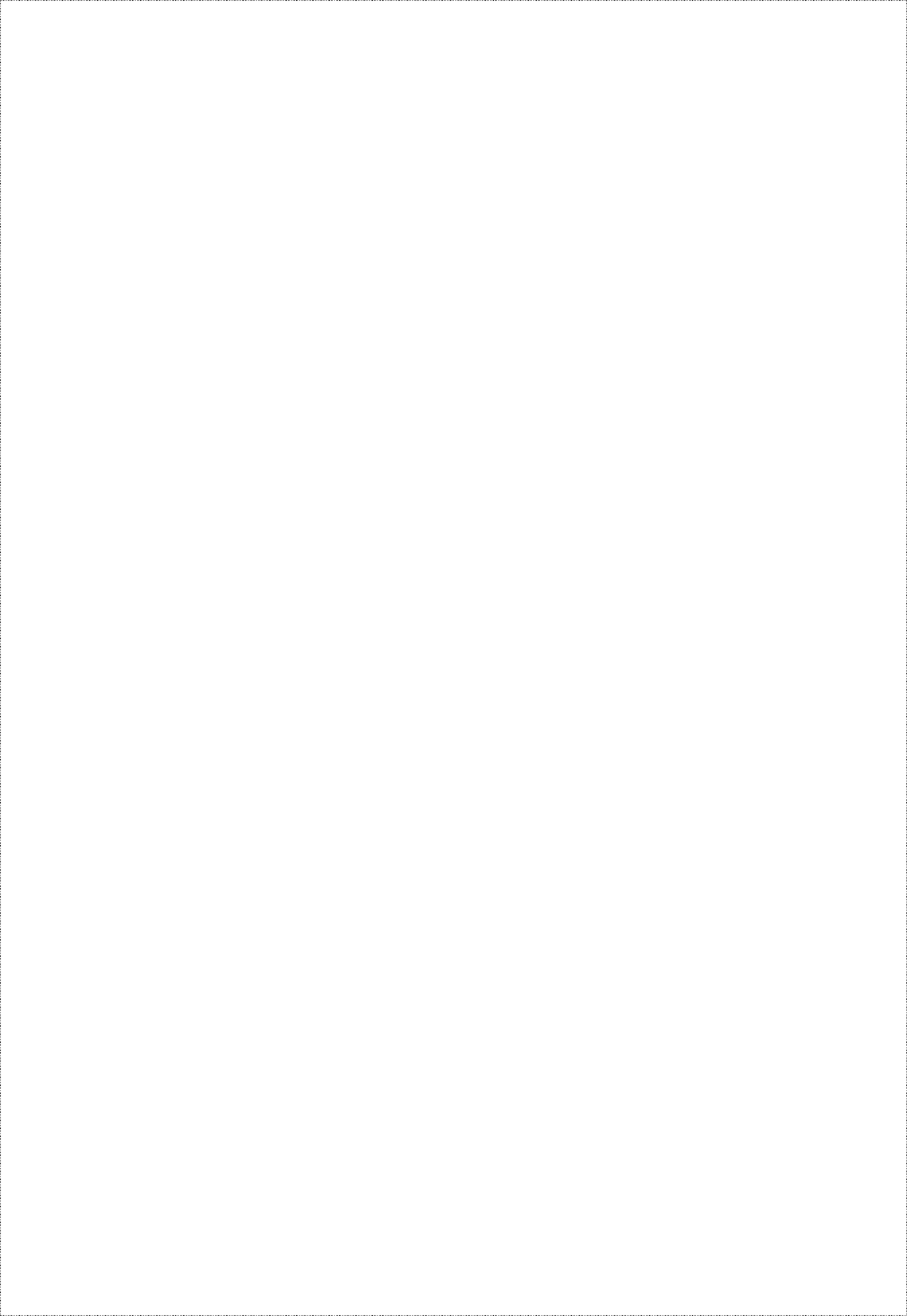
Section: 201-EC

GREEN UNIVERSITY OF BANGLADESH

#### Submitted to

**Tamim Al Mahmud** Assistant Professor Dept. of CSE

GREEN UNIVERSITY OF BANGLADESH



# OBJECTIVE:

The objective of this lab is to Implementation of Depth-First Search (DFS).

* To understand how to represent a graph using adjacency list.
* To understand how Depth-First Search (DFS) works.

# RESOURCE:

* C++

# PROGRAM LOGIC:

**PROCEDURE:**

### Go to DEBUG => Run or press CTRL + F9 to run the program

2

# SOURCE CODE:

**Implementation of Depth-First Search Traversal:**

**//==== Implementation Of DFS ====//**

# include <iostream>

using namespace std;

int n, m, graph[10][10], i, j, k, v, s, visited[10], visit[10], stk[10], top;

int main(){

cout<<"Enter the number of vertices: ";

cin>>n;

cout<<"Enter the number of edges: ";

cin>>m;

for(k=1; k<=m; k++)

{

cin>>i>>j;

graph[i][j]= 1;

}

cout<<"Enter the initial vertex: ";

cin>>v;

cout<<"DFS ORDER: "<<v<<" ";

visited [v] = 1;

s = 1;

while(s<n)

{

for (j=n; j>=1; j--){

if(graph[v][j]!=0 && visited[j]!=1 && visit[j]!=1)

{

visit[j] = 1;

stk [top] = j;

top++;

}

}

v = stk[--top];

cout<<v<<" ";

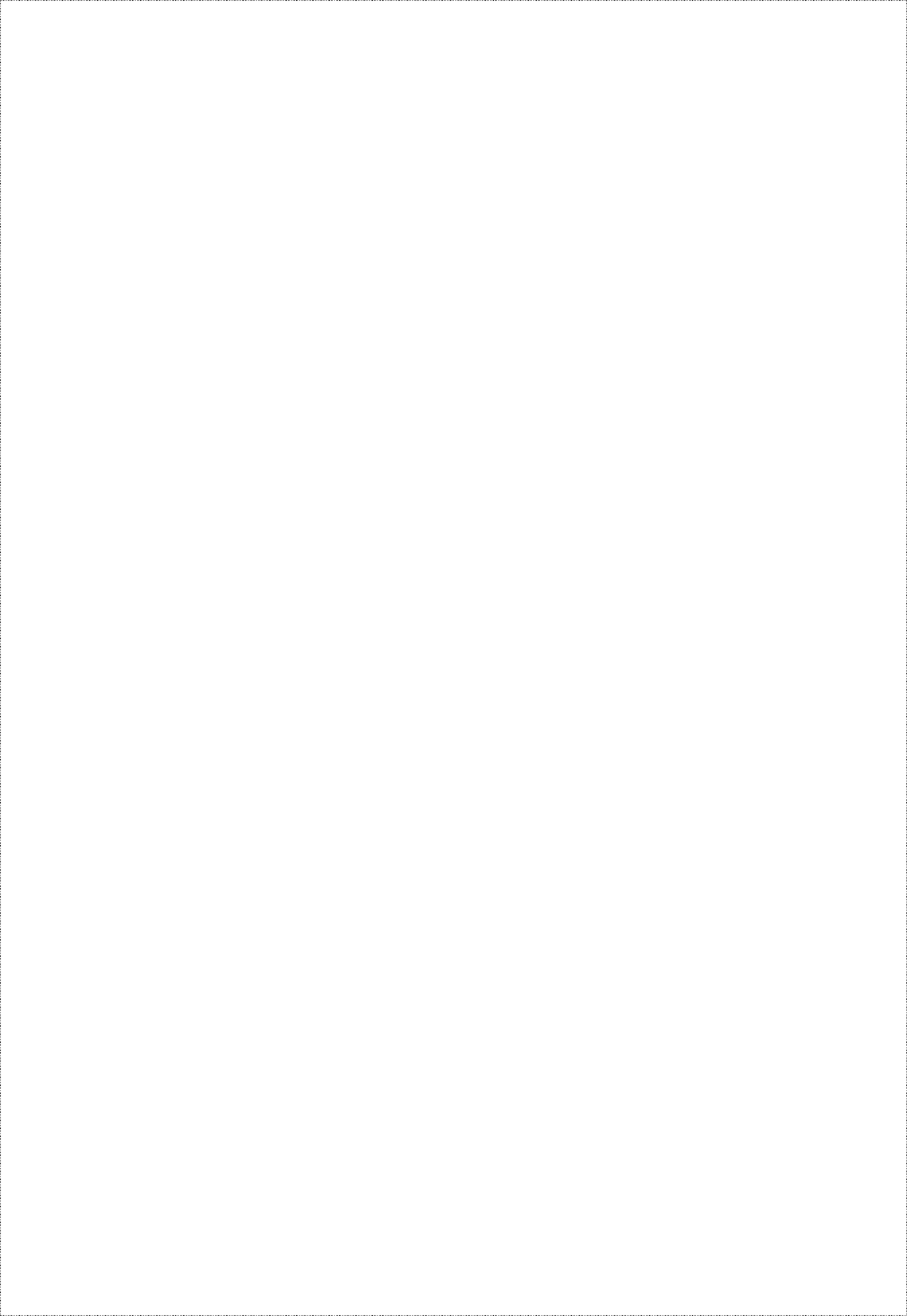
s++;

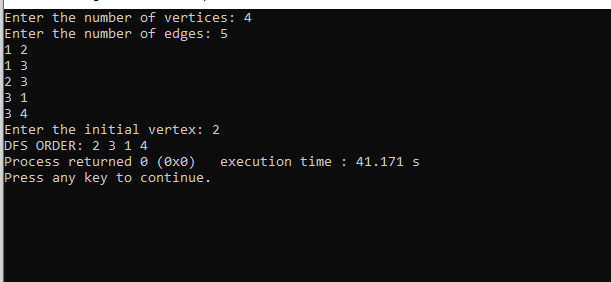
visit[v] = 0;

visited [v] = 1;

}

} 4

**Output:**



5