"Heaven's Light is Our Guide"

Department of Computer Science & Engineering

RAJSHAHI UNIVERSITY OF ENGINEERING & TECHNOLOGY

Lab Report

Course No: CSE 2202

Course Name: Sessional Based on CSE 2201

Submitted to:

Biprodip Pal

Assistant Professor

Department of Computer Science & Engineering

Submitted by:

Md. Al Siam
Department of Computer Science & Engineering
Session: 2016-2017
Section: A
Roll No.: 1603008

Problem:

Checking if it is possible to move back to a node completing a cycle for a given graph.

Solution in C++:

```
#include <bits/stdc++.h>
using namespace std;

int adj[6][6];

int visited[6] = {0};

//int trackara[10], trackindex = 0;

//vector <int, int> tarck;

//stack <int> st;

void makematrix(){
   memset(adj, 0, sizeof (adj));
   adj[0][1] = 1;
   adj[1][2] = 1;
   adj[2][3] = adj[2][4] = adj[2][5] = 1;
   adj[3][0] = 1;
   adj[5][0] = 1;
}

void printmatrix(){
   for(int i = 0; i < 6; i++)
      for(int j = 0; j < 6; j++)
         printf(j == 5 ? "%d\n" : "%d ", adj[i][j]);
}
```

```
void checkfrom(int source){

    stack <int> st;

    st.push(source);

    visited[source] = 1;

    while(!st.empty()){
        int u = st.top();
        st.pop();


        for(int i = 0; i < 6; i++){
            if(adj[u][i] == 1){
                cout << u << "->" << i << "\n";
                visited[i]++;
                if(visited[i] <= 1) st.push(i);
            }
        }
    }
    cout << endl;
}

int main(){
    makematrix();
    printmatrix();

    int n;

    cout << "Enter source: ";
    cin >> n;

    checkfrom(n);
```

```
    cout << "Visit Status:\n";
    for(int i = 0; i < 6; i++)
        printf(i == 5 ? "%d\n" : "%d ", visited[i]);

    if(visited[n] > 1) printf("Possible to get back\n");
    else printf("Not Possible to get back\n");

}
```

Sample Input:

0

Sample Output:

```
0 1 0 0 0 0
0 0 1 0 0 0
0 0 0 1 1 1
1 0 0 0 0 0
0 0 0 0 0 0
1 0 0 0 0 0
Enter source: 0
0->1
1->2
2->3
2->4
2->5
5->0
3->0

Visit Status:
3 1 1 1 1 1
Possible to get back
```