DAA Lab

Practical 6

Amber Shukla

A1-B3-42

**Aim:** Program to implement the code for minimum cost and path for a given Multi-Stage Graph.

**Code:**

#include <bits/stdc++.h>

using namespace std;

typedef long long ll;

int main()

{

    ll nodes, stages;

    cout << "Enter No of Stages: ";

    cin >> stages;

    cout << "Enter No of Nodes: ";

    cin >> nodes;

    vector<vector<ll>> arr(nodes, vector<ll>(nodes, INT\_MAX));

    ll connections;

    cout << "Enter No Of Connections: ";

    cin >> connections;

    for (ll i = 0; i < connections; i++)

    {

        printf("\nConnection No %lld From: To: And Cost: ", i + 1);

        int j, k, cost;

        cin >> j >> k >> cost;

        arr[j][k] = cost;

    }

    vector<ll> bcost(nodes, INT\_MAX), prev(nodes, -1), path(nodes, -1);

    bcost[0] = 0;

    for (ll i = 1; i < nodes; i++)

    {

        for (ll j = 0; j < nodes; j++)

        {

            if (arr[j][i] != INT\_MAX && bcost[j] != INT\_MAX)

            {

                if (bcost[j] + arr[j][i] < bcost[i])

                    bcost[i] = bcost[j] + arr[j][i],

                    prev[i] = j;

            }

        }

        // printf("\nMinimum Cost For Node %d is %d\n with Prev as %d", i, bcost[i], prev[i]);

    }

    cout << endl;

    ll curr = nodes - 2;

    path[nodes - stages] = 0;

    path[nodes - 1] = nodes - 1;

    for (ll i = nodes - 2; i > nodes - stages; i--)

        path[i] = prev[i + 1];

    cout << "Shortest Path is: ";

    for (ll i = nodes - stages; i < nodes; i++)

        cout << path[i] << " - ";

    cout << "With Minimum Cost = " << bcost[nodes - 1];

    return 0;

}

**Output:**

