12/1/23

LAB - 10

Bellman Ford

```
#include <iostream>
#define MAX 10
using namespace std; {
int src,dest, wt;
} edge;
void bell (int nv, edge e[], int src_graph, int ne){
int u, v, weight, i, j = 0;
int dis[MAX];
for (i=0; i<nv ; i++) {
dis[i] =999; }
dis[src_graph] =0;
for (i=0; i<nv-1 ; i++){
for (j =0; j<ne; j++){
u = e[j].src;
v = e[j].dest;
weight = e[j].wt;
if (dis[u]!=999 && dis[u] + weight < dis[v]){
dis[v] =dis[u] + weight;
} } }
for (j=0; j<ne ; j++) {
u = e[j].src;
v = e[j].dest;
weight = e[j].wt;
if (dis[u] + weight < dis[v]) {
cout << "Neg cycle present";
return;
}
}
```

```
cout << "Vertex" << " Dist from source";
for (i=1; i<=nv; i++) {
cout << "\n" << i << "\t" << dis[i];
} }
```

---

o/p :- Enter no. of vertices : 5

Enter adj matrix :

0	6	5	0	0
6	0	0	7	0
5	0	0	4	3
0	7	4	0	2
0	0	3	2	0

Enter starting node : 1
Dist of 0 = 6
Path = 0 <- 1
Count = 2

Dist of 2 = 11
Path = 2 <- 0 <- 1
Count = 3

Dist of 3 = 7
Path = 3 <- 1
Count = 2

Dist of 4 = 9
Path = 4 <- 3 <- 1
Count = 3

12/1/2023