

```

#include <stdio.h>
#include <limits.h>

#define V 5

void bellmanFord(int graph[V][V], int src) {
    int dist[V];
    int i,u,v,k;

    for (i = 0; i < V; i++)
        dist[i] = INT_MAX;
    dist[src] = 0;

    for (k = 0; k < V - 1; k++)
    {
        for (u = 0; u < V; u++)
        {
            for (v = 0; v < V; v++)
            {
                if (graph[u][v] != 0 && dist[u] != INT_MAX && dist[u] +
graph[u][v] < dist[v])
                {
                    dist[v] = dist[u] + graph[u][v];
                }
            }
        }
    }

    for (u = 0; u < V; u++) {
        for (v = 0; v < V; v++) {
            if (graph[u][v] != 0 && dist[u] != INT_MAX && dist[u] +
graph[u][v] < dist[v])
            {
                printf("Graph contains a negative weight cycle!\n");
                return;
            }
        }
    }

    printf("Vertex\tDistance from Source\n");
    for (i = 0; i < V; i++)
    {
        if (dist[i] == INT_MAX)
            printf("%d\tINF\n", i);
        else
            printf("%d\t%d\n", i, dist[i]);
    }
}

int main() {

    int graph[V][V] = {
        {0, -1, 4, 0, 0},
        {0, 0, 3, 2, 2},
        {0, 0, 0, 0, 0},
    }
}

```

```
        {0, 1, 5, 0, 0},  
        {0, 0, 0, -3, 0}  
};  
  
int source = 0;  
  
bellmanFord(graph, source);  
  
return 0;  
}
```