Assignment 2

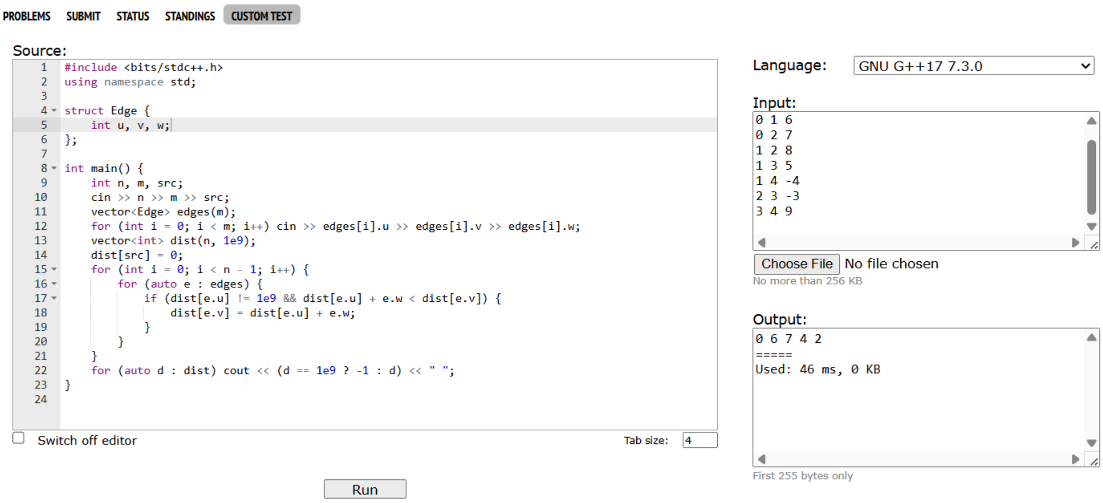
Problem Statement : Write a program to implement Bellman-Ford Algorithm using Dynamic Programming and verify the time complexity .

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

struct Edge {
    int u, v, w;
};

int main() {
    int n, m, src;
    cin >> n >> m >> src;
    vector<Edge> edges(m);
    for (int i = 0; i < m; i++) cin >> edges[i].u >> edges[i].v >> edges[i].w;
    vector<int> dist(n, 1e9);
    dist[src] = 0;
    for (int i = 0; i < n - 1; i++) {
        for (auto e : edges) {
            if (dist[e.u] != 1e9 && dist[e.u] + e.w < dist[e.v]) {
                dist[e.v] = dist[e.u] + e.w;
            }
        }
    }
    for (auto d : dist) cout << (d == 1e9 ? -1 : d) << " ";
}
```

Test Case 1 :



Source:

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

struct Edge {
    int u, v, w;
};

int main() {
    int n, m, src;
    cin >> n >> m >> src;
    vector<Edge> edges(m);
    for (int i = 0; i < m; i++) cin >> edges[i].u >> edges[i].v >> edges[i].w;
    vector<int> dist(n, 1e9);
    dist[src] = 0;
    for (int i = 0; i < n - 1; i++) {
        for (auto e : edges) {
            if (dist[e.u] != 1e9 && dist[e.u] + e.w < dist[e.v]) {
                dist[e.v] = dist[e.u] + e.w;
            }
        }
    }
    for (auto d : dist) cout << (d == 1e9 ? -1 : d) << " ";
}
```

Tab size: 4

Language: GNU G++17 7.3.0

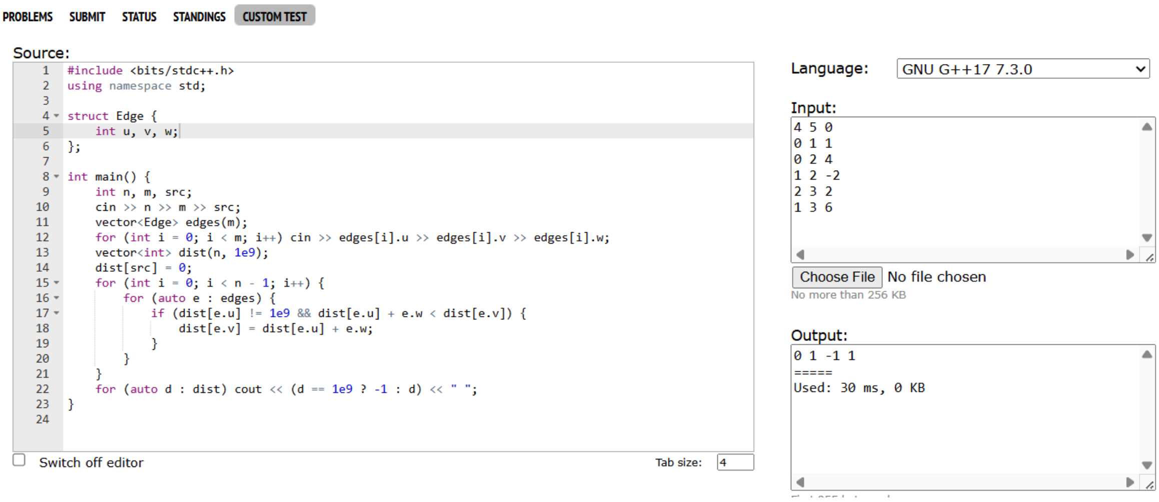
Input:

```
0 1 6
0 2 7
1 2 8
1 3 5
1 4 -4
2 3 -3
3 4 9
```

Output:

```
0 6 7 4 2
=====
Used: 46 ms, 0 KB
```

Test Case 2 :



Source:

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

struct Edge {
    int u, v, w;
};

int main() {
    int n, m, src;
    cin >> n >> m >> src;
    vector<Edge> edges(m);
    for (int i = 0; i < m; i++) cin >> edges[i].u >> edges[i].v >> edges[i].w;
    vector<int> dist(n, 1e9);
    dist[src] = 0;
    for (int i = 0; i < n - 1; i++) {
        for (auto e : edges) {
            if (dist[e.u] != 1e9 && dist[e.u] + e.w < dist[e.v]) {
                dist[e.v] = dist[e.u] + e.w;
            }
        }
    }
    for (auto d : dist) cout << (d == 1e9 ? -1 : d) << " ";
}
```

Tab size: 4

Language: GNU G++17 7.3.0

Input:

```
4 5 0
0 1 1
0 2 4
1 2 -2
2 3 2
1 3 6
```

Output:

```
0 1 -1 1
=====
Used: 30 ms, 0 KB
```