

Code for the Inhabitants :

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 const int inf = 1e7;
4 const int n = 1e3;
5 vector<pair<int, int>> g[n];
6 // { node , cost}
7 void dijkstra(int source)
8 {
9     vector<int> distance(n, inf);
10    vector<bool> visited(n, 0);
11    set<pair<int, int>> st;
12    // {cost , node} => kept cost on first value to sort based on lowest cost
13    st.insert({0, source});
14    distance[source] = 0;
15    while (!st.empty())
16    {
17        auto node = *st.begin();
18        // will give the minimum weighted pair {cost , node}
19        int parent_node = node.second;
20        int parent_node_cost = node.first;
21        st.erase(st.begin());
22        if (visited[parent_node])
23        {
24            continue;
25        }
26        visited[parent_node] = 1;
27        // Traverse to the child of v, for Relaxation
28        for (auto child : g[parent_node])
29        {
30            int child_node = child.first;
31            int edge_cost = child.second;
32
33            // Relaxation
34            if ((parent_node_cost + edge_cost) < distance[child_node])
35            {
36                distance[child_node] = (parent_node_cost + edge_cost);
37                st.insert({distance[child_node], child_node});
38            }
39        }
40    }
41    cout << "Node\tDistance from " << source << endl;
42    for (int i = 0; i < n; ++i)
43    {
44        if (distance[i] != inf)
45        {
46            cout << i << "\t" << distance[i] << endl;
47        }
48    }
49 }
50 int main()
51 {
52     int node, edge;
53     cin >> node >> edge;
54     for (int i = 0; i < edge; i++)
55     {
56         int u, v, cost;
57         cin >> u >> v >> cost;
58         g[u].push_back({v, cost});
59         g[v].push_back({u, cost});
60         // u/v indexed node connected with v/u node with cost
61     }
62     dijkstra(0);
63
64     return 0;
65 }
66
```

Minimum costs output for 0 number city's Inhabitants to travel into other cities using C++ :

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

```
8 17
0 1 9
0 4 7
0 3 3
1 4 8
1 5 11
1 7 6
1 3 17
1 2 2
2 7 3
2 6 11
2 5 19
3 4 5
4 6 5
4 5 12
5 7 13
5 6 6
6 7 8
```

City	Distance from 0
0	0
1	9
2	11
3	3
4	7
5	18
6	12
7	14

 SAKIB  AA Test