MODERN OPERATING SYSTEMS AND COMPUTER NETWORK

LAB-2

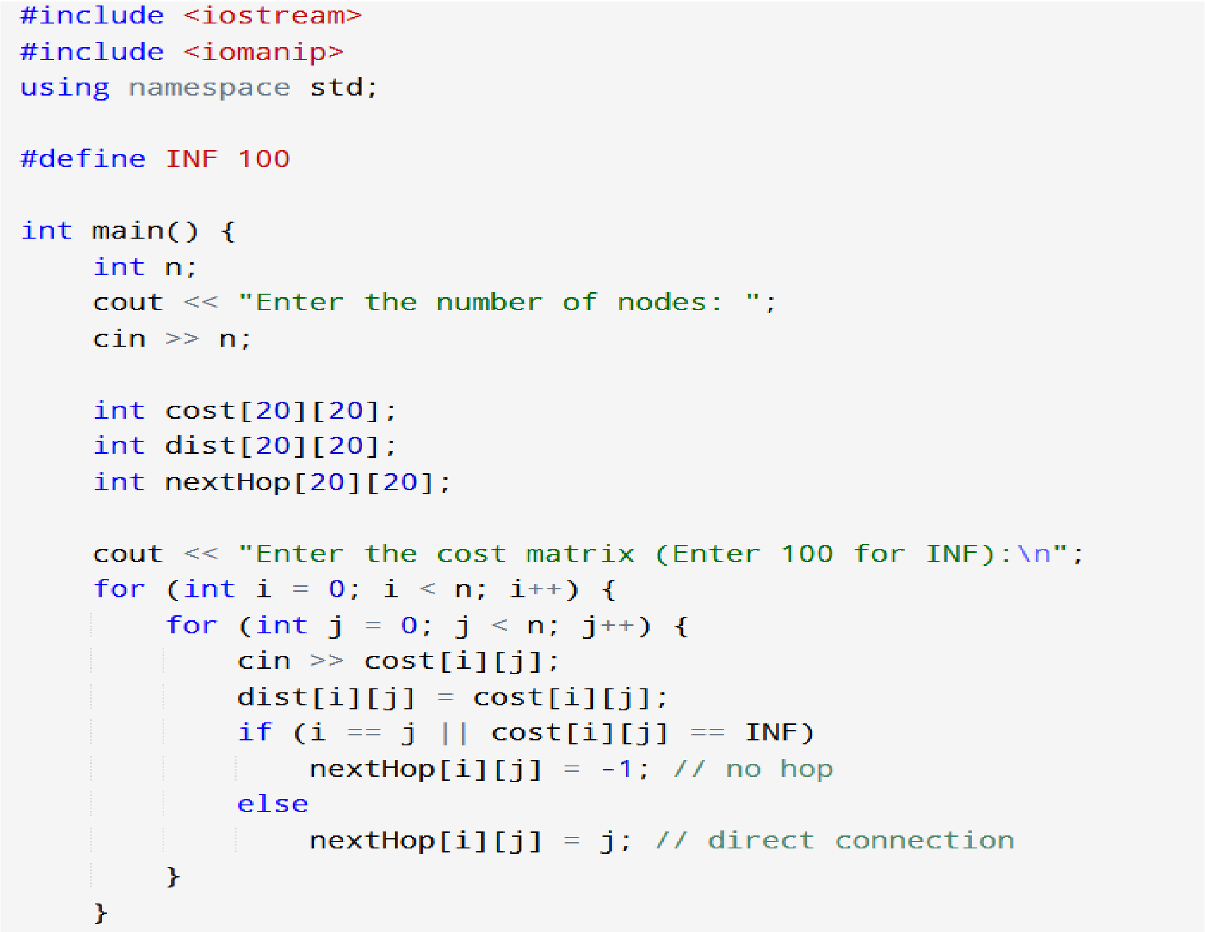
ENROLLMENT NUMBER:2503B05108

Question:

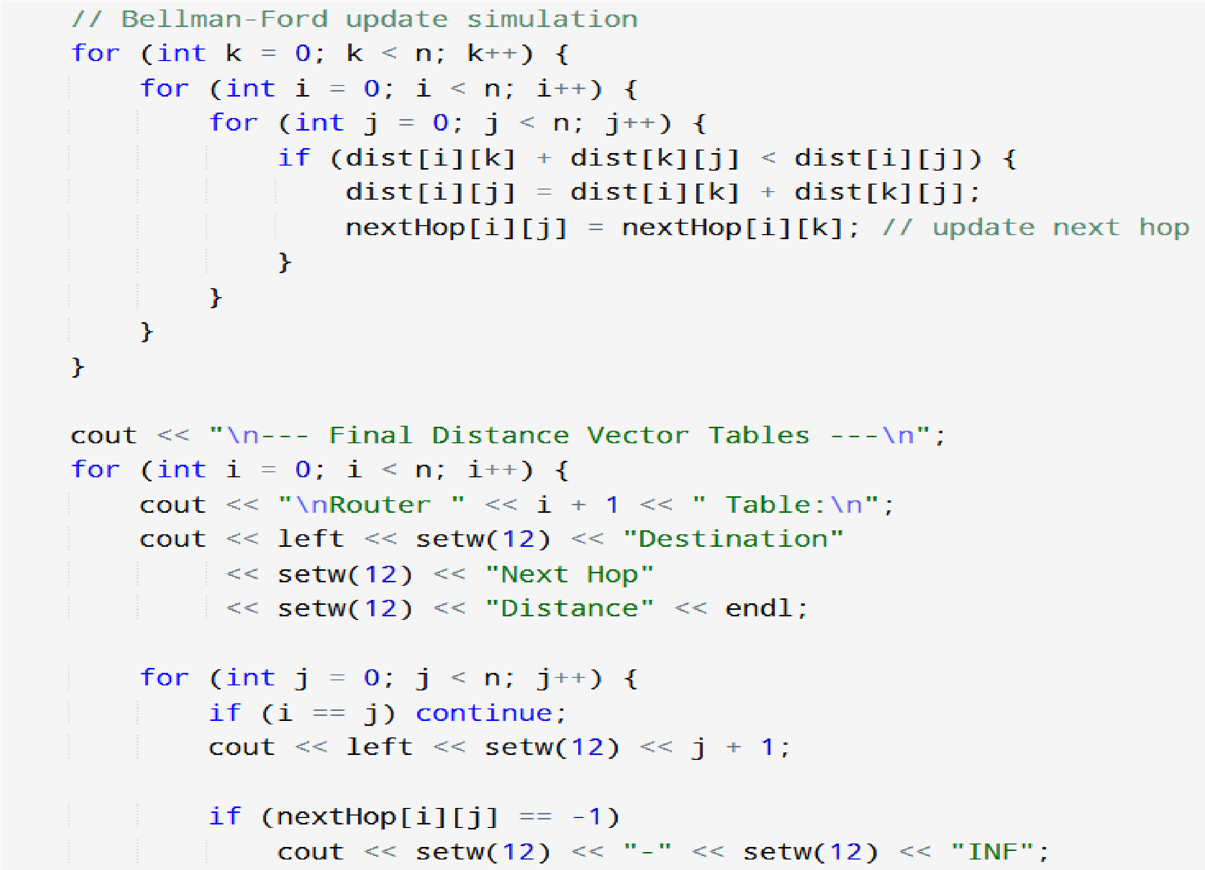
Simulation of **Distance Vector Routing** using the **Bellman-Ford Algorithm** in C++.

```
Enter the number of nodes: 4

Enter the cost matrix (Enter 100 for INF):

0 2 5 1

2 0 3 2

5 3 0 3

1 2 3 0
```

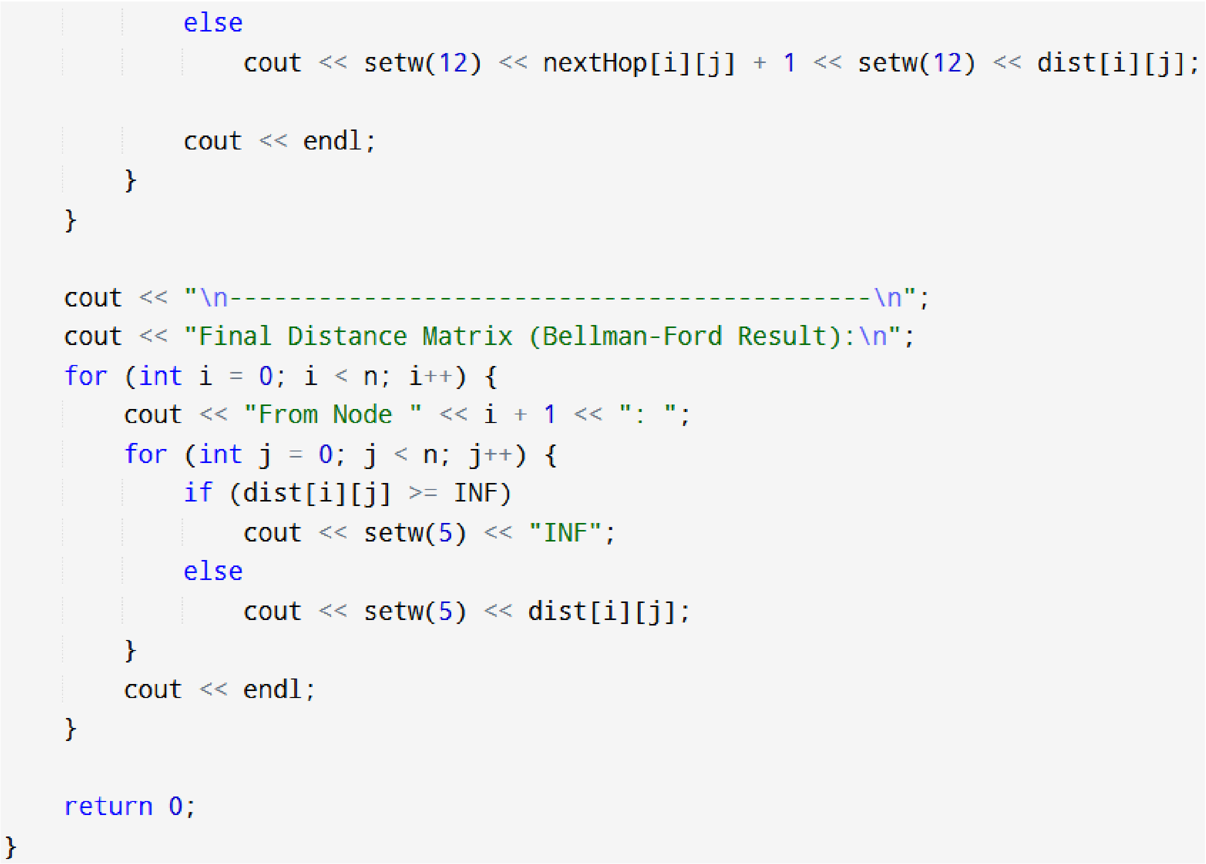
Code:



```cpp
#include <iostream>
#include <iomanip>
using namespace std;

#define INF 100

int main() {
    int n;
    cout << "Enter the number of nodes: ";
    cin >> n;

    int cost[20][20];
    int dist[20][20];
    int nextHop[20][20];

    cout << "Enter the cost matrix (Enter 100 for INF):\n";
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> cost[i][j];
            dist[i][j] = cost[i][j];
            if (i == j || cost[i][j] == INF)
                nextHop[i][j] = -1; // no hop
            else
                nextHop[i][j] = j; // direct connection
        }
    }
```

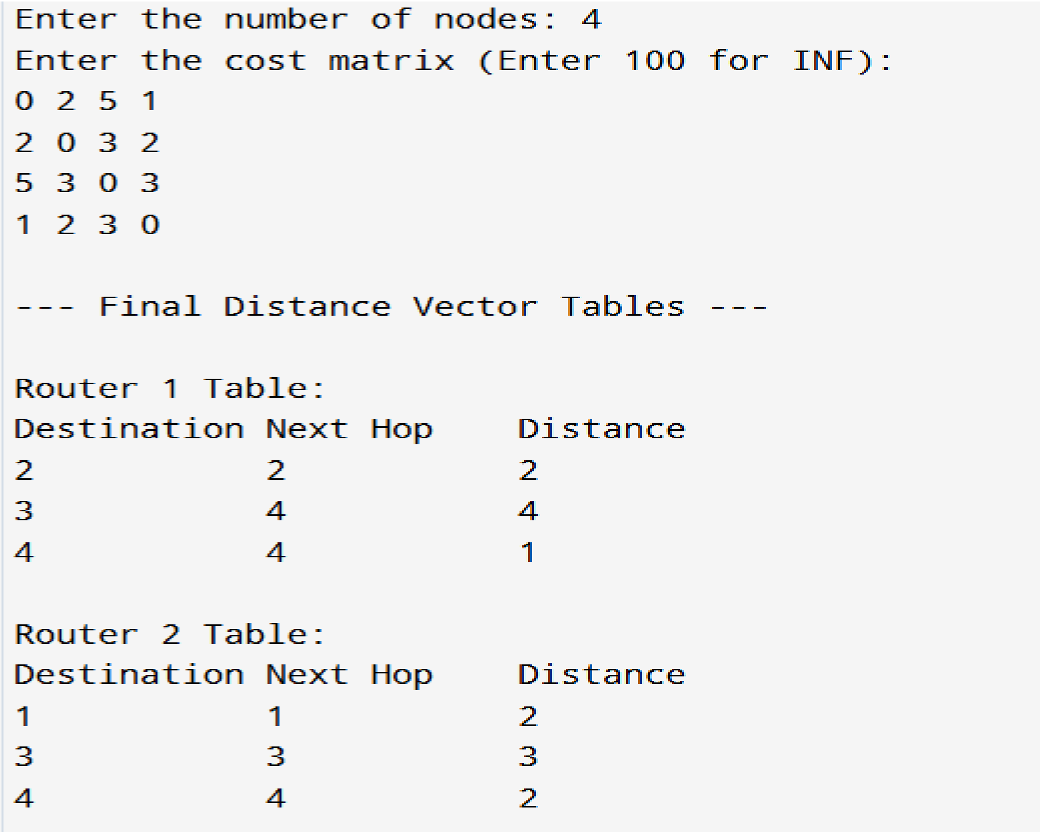


```cpp
    // Bellman-Ford update simulation
    for (int k = 0; k < n; k++) {
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if (dist[i][k] + dist[k][j] < dist[i][j]) {
                    dist[i][j] = dist[i][k] + dist[k][j];
                    nextHop[i][j] = nextHop[i][k]; // update next hop
                }
            }
        }
    }

    cout << "\n--- Final Distance Vector Tables ---\n";
    for (int i = 0; i < n; i++) {
        cout << "\nRouter " << i + 1 << " Table:\n";
        cout << left << setw(12) << "Destination"
             << setw(12) << "Next Hop"
             << setw(12) << "Distance" << endl;

        for (int j = 0; j < n; j++) {
            if (i == j) continue;
            cout << left << setw(12) << j + 1;

            if (nextHop[i][j] == -1)
                cout << setw(12) << "-" << setw(12) << "INF";
```

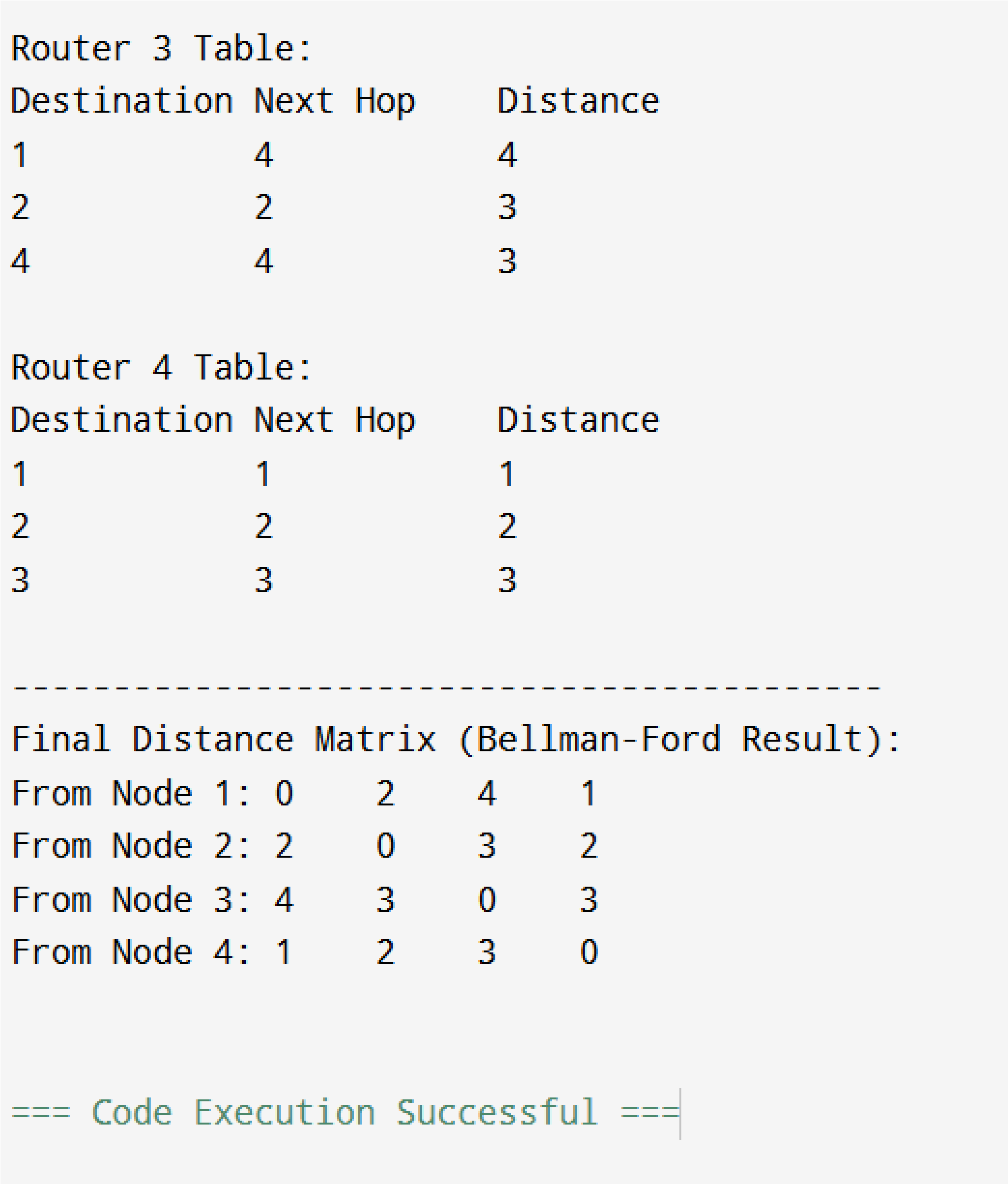


```
            else
                cout << setw(12) << nextHop[i][j] + 1 << setw(12) << dist[i][j];

            cout << endl;
        }
    }

    cout << "\n------------------------------------------\n";
    cout << "Final Distance Matrix (Bellman-Ford Result):\n";
    for (int i = 0; i < n; i++) {
        cout << "From Node " << i + 1 << ": ";
        for (int j = 0; j < n; j++) {
            if (dist[i][j] >= INF)
                cout << setw(5) << "INF";
            else
                cout << setw(5) << dist[i][j];
        }
        cout << endl;
    }

    return 0;
}
```

Output:



```
Enter the number of nodes: 4
Enter the cost matrix (Enter 100 for INF):
0 2 5 1
2 0 3 2
5 3 0 3
1 2 3 0

--- Final Distance Vector Tables ---

Router 1 Table:
Destination Next Hop    Distance
2           2           2
3           4           4
4           4           1

Router 2 Table:
Destination Next Hop    Distance
1           1           2
3           3           3
4           4           2
```



```
Router 3 Table:
Destination Next Hop    Distance
1           4           4
2           2           3
4           4           3

Router 4 Table:
Destination Next Hop    Distance
1           1           1
2           2           2
3           3           3

-------------------------------------------
Final Distance Matrix (Bellman-Ford Result):
From Node 1: 0    2    4    1
From Node 2: 2    0    3    2
From Node 3: 4    3    0    3
From Node 4: 1    2    3    0


=== Code Execution Successful ===
```