Name: Amber Shukla

Batch: A1-B3-42

DAA Lab

Practical – 2 Fast Learners Problem

Minimum Spanning Forest:

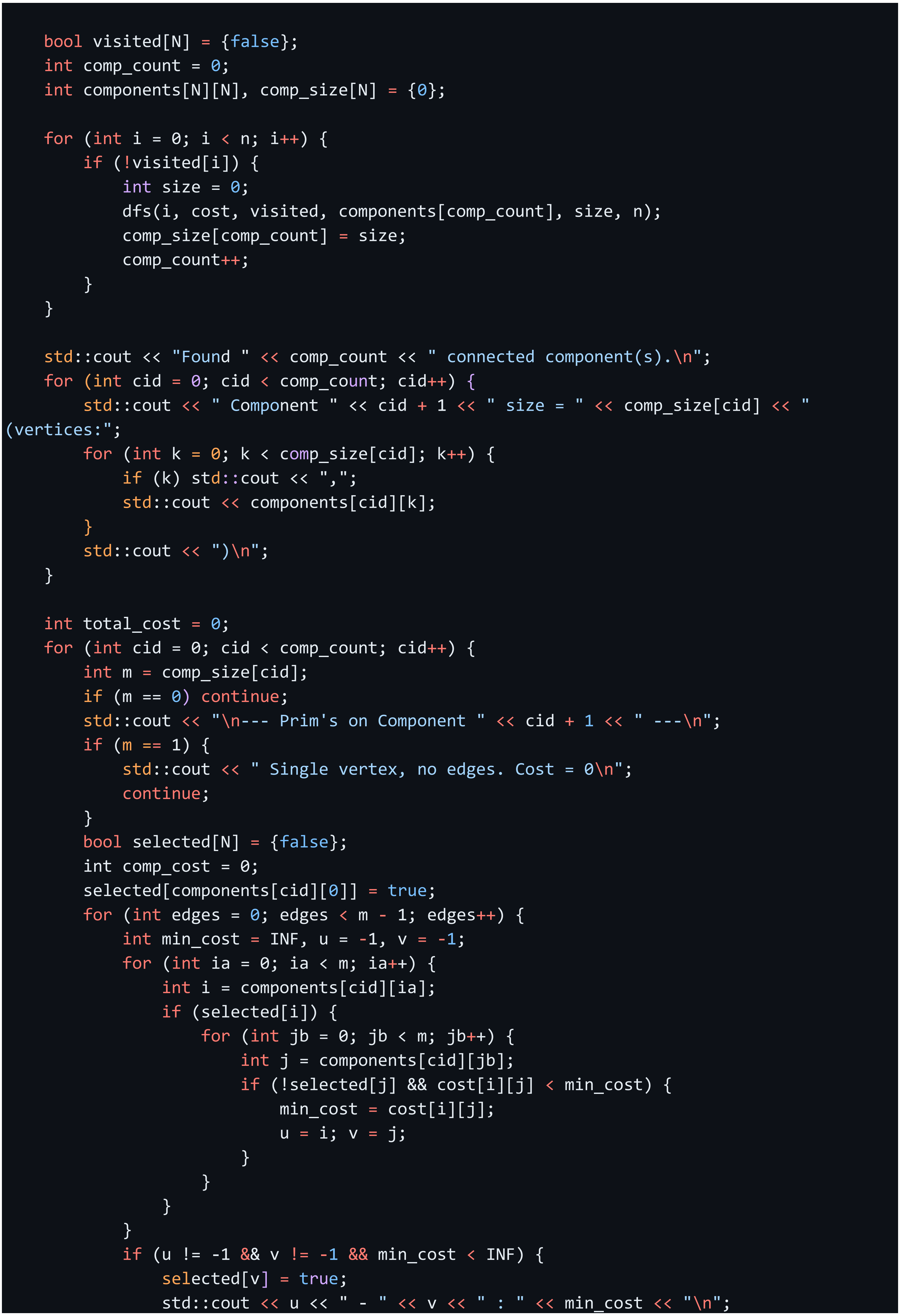
a. Consider a fixed input graph/matrix 20x20 size from a file. The matrix should be such that there are minimum of 3 disconnected components.

b. Find the connected components.

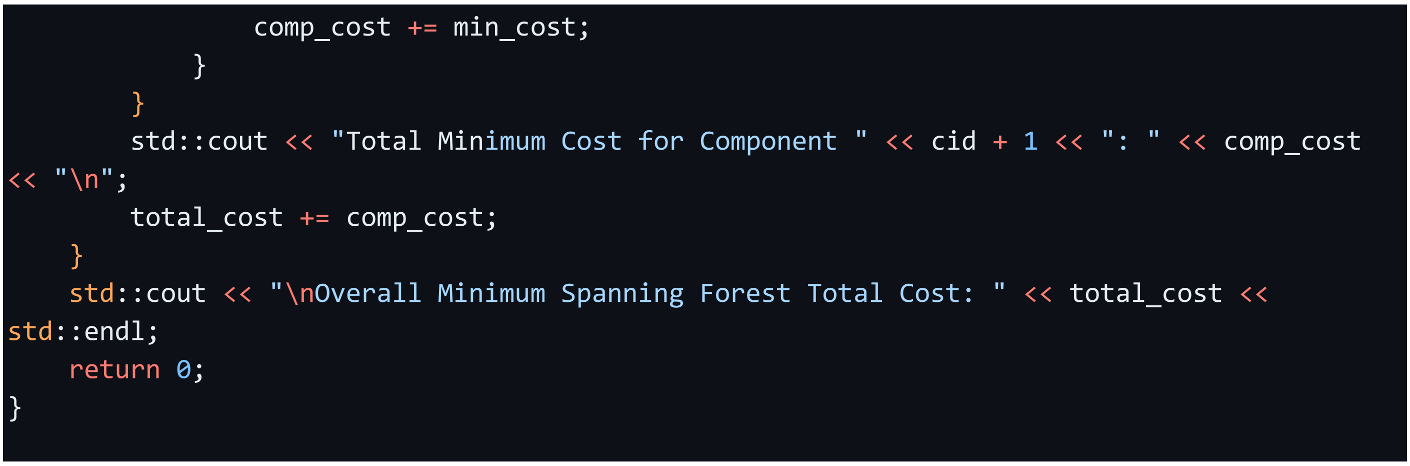
c. Run Prim’s separately on each connected component to get a Minimum Spanning Forest

Code:

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


const int N = 20;
const int INF = 99;


void dfs(int u, int cost[N][N], bool visited[N], int component[], int &size, int n) {
    visited[u] = true;
    component[size++] = u;
    for (int v = 0; v < n; v++) {
        if (!visited[v] && cost[u][v] < INF) {
            dfs(v, cost, visited, component, size, n);
        }
    }
}


int main() {
    int n = N, cost[N][N];
    std::ifstream fin("matrix.txt");
    if (!fin) {
        std::cerr << "Error: cannot open 'matrix.txt'." << std::endl;
        return 1;
    }
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            if (!(fin >> cost[i][j])) {
                std::cerr << "Error: expected 400 numbers in 'matrix.txt'." << std::endl;
                return 1;
            }
    fin.close();
```

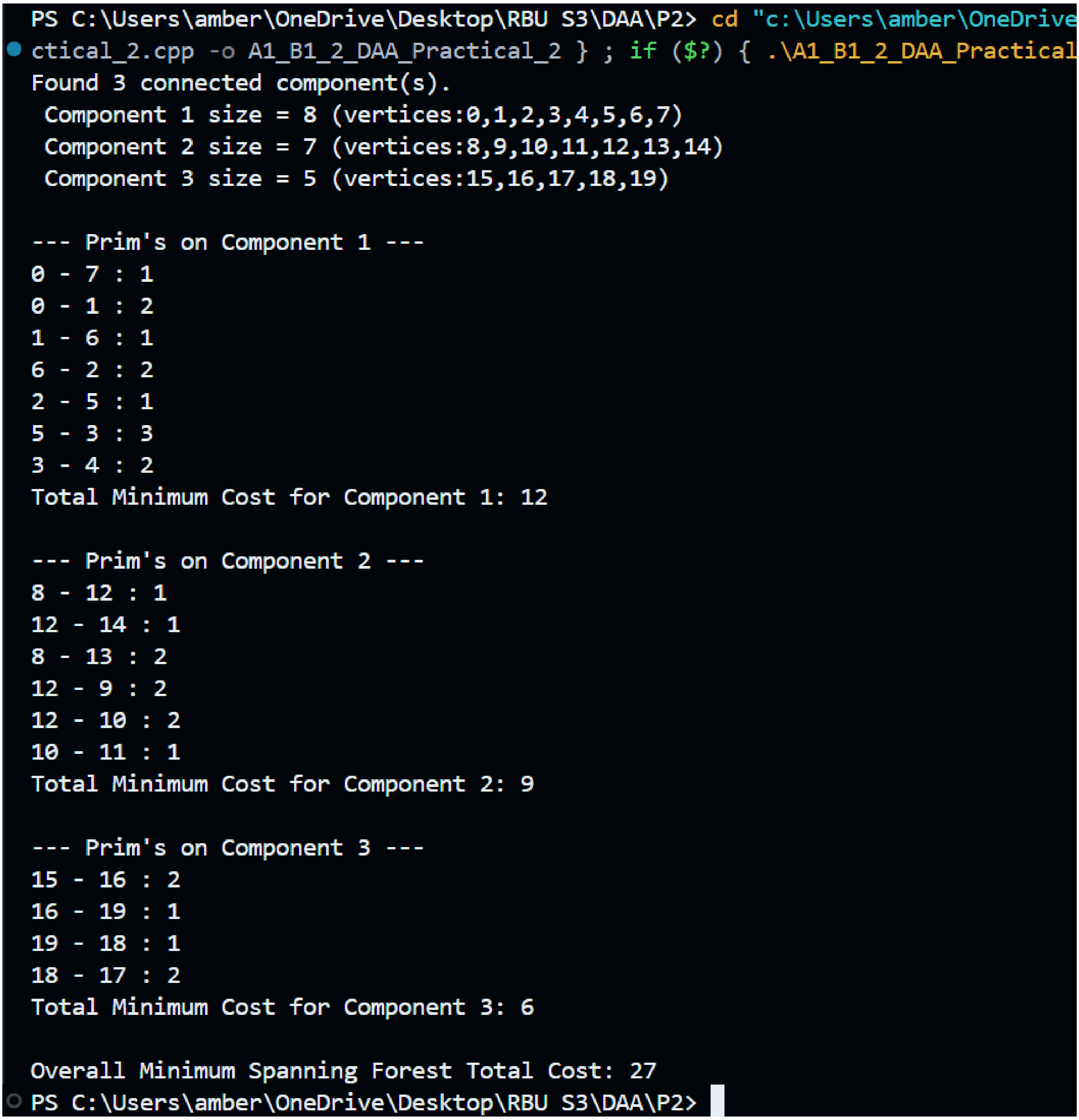


```
    bool visited[N] = {false};
    int comp_count = 0;
    int components[N][N], comp_size[N] = {0};

    for (int i = 0; i < n; i++) {
        if (!visited[i]) {
            int size = 0;
            dfs(i, cost, visited, components[comp_count], size, n);
            comp_size[comp_count] = size;
            comp_count++;
        }
    }

    std::cout << "Found " << comp_count << " connected component(s).\n";
    for (int cid = 0; cid < comp_count; cid++) {
        std::cout << " Component " << cid + 1 << " size = " << comp_size[cid] << "
(vertices:";
        for (int k = 0; k < comp_size[cid]; k++) {
            if (k) std::cout << ",";
            std::cout << components[cid][k];
        }
        std::cout << ")\n";
    }

    int total_cost = 0;
    for (int cid = 0; cid < comp_count; cid++) {
        int m = comp_size[cid];
        if (m == 0) continue;
        std::cout << "\n--- Prim's on Component " << cid + 1 << " ---\n";
        if (m == 1) {
            std::cout << " Single vertex, no edges. Cost = 0\n";
            continue;
        }
        bool selected[N] = {false};
        int comp_cost = 0;
        selected[components[cid][0]] = true;
        for (int edges = 0; edges < m - 1; edges++) {
            int min_cost = INF, u = -1, v = -1;
            for (int ia = 0; ia < m; ia++) {
                int i = components[cid][ia];
                if (selected[i]) {
                    for (int jb = 0; jb < m; jb++) {
                        int j = components[cid][jb];
                        if (!selected[j] && cost[i][j] < min_cost) {
                            min_cost = cost[i][j];
                            u = i; v = j;
                        }
                    }
                }
            }
            if (u != -1 && v != -1 && min_cost < INF) {
                selected[v] = true;
                std::cout << u << " - " << v << " : " << min_cost << "\n";
```



```
                comp_cost += min_cost;
            }
        }
        std::cout << "Total Minimum Cost for Component " << cid + 1 << ": " << comp_cost
<< "\n";
        total_cost += comp_cost;
    }
    std::cout << "\nOverall Minimum Spanning Forest Total Cost: " << total_cost <<
std::endl;
    return 0;
}
```

Output:



```
PS C:\Users\amber\OneDrive\Desktop\RBU S3\DAA\P2> cd "c:\Users\amber\OneDrive
ctical_2.cpp -o A1_B1_2_DAA_Practical_2 } ; if ($?) { .\A1_B1_2_DAA_Practical
Found 3 connected component(s).
 Component 1 size = 8 (vertices:0,1,2,3,4,5,6,7)
 Component 2 size = 7 (vertices:8,9,10,11,12,13,14)
 Component 3 size = 5 (vertices:15,16,17,18,19)

--- Prim's on Component 1 ---
0 - 7 : 1
0 - 1 : 2
1 - 6 : 1
6 - 2 : 2
2 - 5 : 1
5 - 3 : 3
3 - 4 : 2
Total Minimum Cost for Component 1: 12

--- Prim's on Component 2 ---
8 - 12 : 1
12 - 14 : 1
8 - 13 : 2
12 - 9 : 2
12 - 10 : 2
10 - 11 : 1
Total Minimum Cost for Component 2: 9

--- Prim's on Component 3 ---
15 - 16 : 2
16 - 19 : 1
19 - 18 : 1
18 - 17 : 2
Total Minimum Cost for Component 3: 6

Overall Minimum Spanning Forest Total Cost: 27
PS C:\Users\amber\OneDrive\Desktop\RBU S3\DAA\P2>
```