**#include<iostream>**

**using namespace std;**

**class Tree {**

**int a[20][20], visited[20], v, e, l, u, w;**

**public:**

**void input() {**

**cout << "Enter the number of branches: ";**

**cin >> v;**

**for (int i = 0; i < v; i++) {**

**visited[i] = 0;**

**for (int j = 0; j < v; j++) {**

**a[i][j] = 999;**

**}**

**}**

**cout << "\nEnter the number of connections: ";**

**cin >> e;**

**for (int i = 0; i < e; i++) {**

**cout << "Enter the end branches of connections: ";**

**cin >> l >> u;**

**cout << "Enter the phone company charges for this connection: ";**

**cin >> w;**

**a[l - 1][u - 1] = a[u - 1][l - 1] = w;**

**}**

**}**

**void display() {**

**cout << "\nAdjacency matrix:\n";**

**for (int i = 0; i < v; i++) {**

**for (int j = 0; j < v; j++) {**

**cout << a[i][j] << " ";**

**}**

**cout << endl;**

**}**

**}**

**void minimum() {**

**int p = 0, q = 0, total = 0, min;**

**visited[0] = 1;**

**for (int count = 0; count < (v - 1); count++) {**

**min = 999;**

**for (int i = 0; i < v; i++) {**

**if (visited[i] == 1) {**

**for (int j = 0; j < v; j++) {**

**if (visited[j] != 1) {**

**if (min > a[i][j]) {**

**min = a[i][j];**

**p = i;**

**q = j;**

**}**

**}**

**}**

**}**

**}**

**visited[p] = 1;**

**visited[q] = 1;**

**total = total + min;**

**cout << "Minimum cost connection is " << (p + 1) << " -> " << (q + 1) << " with charge : " << min << endl;**

**}**

**cout << "The minimum total cost of connections of all branches is: " << total << endl;**

**}**

**};**

**int main() {**

**int ch;**

**Tree t;**

**do {**

**cout << "========== PRIM'S ALGORITHM ==========" << endl;**

**cout << "\n1. INPUT\n2. DISPLAY\n3. MINIMUM\n" << endl;**

**cout << "Enter your choice: ";**

**cin >> ch;**

**switch (ch) {**

**case 1:**

**cout << "\*\*\*\*\*\*\* INPUT YOUR VALUES \*\*\*\*\*\*\*" << endl;**

**t.input();**

**break;**

**case 2:**

**cout << "\*\*\*\*\*\*\* DISPLAY THE CONTENTS \*\*\*\*\*\*\*" << endl;**

**t.display();**

**break;**

**case 3:**

**cout << "\*\*\*\*\*\*\*\*\* MINIMUM \*\*\*\*\*\*\*\*\*\*\*\*" << endl;**

**t.minimum();**

**break;**

**}**

**} while (ch != 4);**

**return 0;**

**}**