**Practical No: 8**

**Program:**

#include <iostream>

using namespace std;

class Tree

{

int a[20][20], s, d, w, v, e, visited[20];

public:

void input();

void display();

void minimum();

};

void Tree::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 << "Enter the number of connections: ";

cin >> e;

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

cout << "Enter the end branches of the connection:\n";

cin >> s >> d;

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

cin >> w;

a[s - 1][d - 1] = a[d - 1][s - 1] = w;

}

}

void Tree::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 Tree::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 && min > a[i][j])

{

min = a[i][j];

p = i;

q = j;

}

}

}

}

visited[p] = 1;

visited[q] = 1;

total += min;

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

}

cout << "The minimum total cost of connection for all branches is: " << total << endl;

}

int main()

{

int ch;

Tree t;

do {

cout << "\n1. Input\n2. Display\n3. Minimum\n4. Exit Program\nEnter your choice: ";

cin >> ch;

switch (ch)

{

case 1:

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Input Your Values \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

t.input();

break;

case 2:

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Display the Content \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

t.display();

break;

case 3:

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Minimum \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

t.minimum();

break;

case 4:

cout << "Exiting Program. Thank you!\n";

break;

default:

cout << "Invalid choice. Try again.\n";

}

} while (ch != 4);

return 0;

}

***OUTPUT :***

*1. Input*

*2. Display*

*3. Minimum*

*4. Exit Program*

*Enter your choice: 1*

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Input Your Values \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*Enter the number of branches: 5*

*Enter the number of connections: 6*

*Enter the end branches of the connection:*

*1 2*

*Enter the phone company charges for this connection: 4*

*Enter the end branches of the connection:*

*2 3*

*Enter the phone company charges for this connection: 2*

*Enter the end branches of the connection:*

*1 3*

*Enter the phone company charges for this connection: 3*

*Enter the end branches of the connection:*

*3 4*

*Enter the phone company charges for this connection: 2*

*Enter the end branches of the connection:*

*4 5*

*Enter the phone company charges for this connection: 4*

*Enter the end branches of the connection:*

*3 4*

*Enter the phone company charges for this connection: 6*

*1. Input*

*2. Display*

*3. Minimum*

*4. Exit Program*

*Enter your choice: 2*

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Display the Content \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*Adjacency Matrix:*

*999 4 3 999 999*

*4 999 2 999 999*

*3 2 999 6 999*

*999 999 6 999 4*

*999 999 999 4 999*

*1. Input*

*2. Display*

*3. Minimum*

*4. Exit Program*

*Enter your choice: 3*

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Minimum \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*Minimum cost connection is: 1 -> 3 with charge: 3*

*Minimum cost connection is: 3 -> 2 with charge: 2*

*Minimum cost connection is: 3 -> 4 with charge: 6*

*Minimum cost connection is: 4 -> 5 with charge: 4*

*The minimum total cost of connection for all branches is: 15*

*1. Input*

*2. Display*

*3. Minimum*

*4. Exit Program*

*Enter your choice:4*

*Exiting Program. Thank you!*