**CODE:-**

#include <bits/stdc++.h>

using namespace std;

int n, p;

int rd[1100];

int wt[1100];

vector<int> a;

vector<int> b;

vector<int> c;

int ans;

int dfs(int w)

{

    if (cd[w] == 0)

        return w;

    if (wt[w] < ans)

        ans = wt[w];

    return dfs(cd[w]);

}

void solve(int arr[][3])

{

    int i = 0;

    while (i < p)

    {

        int q = arr[i][0], h = arr[i][1],

            t = arr[i][2];

cd[q] = h;

        wt[q] = t;

        rd[h] = q;

        i++;

    }

    a.clear();

    b.clear();

    c.clear();

for (int j = 1; j <= n; ++j)

        if (rd[j] == 0 && cd[j])

        {

            ans = 1000000000;

            int w = dfs(j);

            a.push\_back(j);

            b.push\_back(w);

            c.push\_back(ans);

        }

cout << a.size() << endl;

    for (int j = 0; j < a.size(); ++j)

        cout << a[j] << " " << b[j]

             << " " << c[j] << endl;

}

int main()

{

    n = 9, p = 6;

memset(rd, 0, sizeof(rd));

    memset(cd, 0, sizeof(cd));

    memset(wt, 0, sizeof(wt));

    cout << "Enter the no of edges:-";

    int l;

    cin >> l;

    cout << "Enter the edges and their weights:-"<<endl;

    for (int i = 0; i < l; i++)

    {

        for (int j = 0; j < 3; j++)

        {

            cin >> arr[i][j];

        }

    }

    cout << "Ans matrix and no of answers as follows:-" << endl;

    solve(arr);

return 0;

}

**OUTPUT:-** 