 #include <iostream>  
    using namespace std;   
    int n, m, mincost, i, j;  
    struct Edge {  
    int u, v, wt;  
    };  
    int checkCycle (Edge e, int\*);  
    int main() {  
    cout << "Enter the number of vertices in the graph: ";  
    cin >> n;  
    cout << "Enter the number of edges in the graph: ";  
    cin >> m;  
    int path[50+1];  
    struct Edge e[50];  
    struct Edge t;  
    for (i=0; i<m; i++) {  
    cout << "Enter 2 vertices and weight of edge " << i+1 << endl;  
    cout << "First vertex: ";  
    cin >> e[i].u;  
    cout << "Second vertex: ";  
    cin >> e[i].v;  
    cout << "Weight: ";  
    cin >> e[i].wt;  
    cout << endl;  
    }  
    for (i=0; i<=m-1; i++) {  
    for (j=0; j<m-i-1; j++) {  
    if (e[j].wt > e[j+1].wt) {  
    t = e[j];  
    e[j] = e[j+1];  
    e[j+1] = t;  
    }  
    }  
    }  
    for (i=1; i<=n; i++) {  
    path[i] = 0;  
    }  
    i = 0;  
    j = 0;  
    mincost = 0;  
    cout << endl;  
    while ((i!=n-1) && (j!=m)) {  
    cout << "Edge ("  
    << e[j].u << ", " << e[j].v << ") "  
    << "with weight " << e[j].wt << " ";  
    if (checkCycle(e[j], path)) {  
    mincost = mincost + e[j].wt;  
    i++;  
    cout << "is selected";  
    } else {  
    cout << "is discarded";  
    }  
    cout << endl;  
    j++;  
    }  
    cout << "Minimum spanning tree is"<<mincost;  
    if (i!=n-1) {  
    cout << "Minimum spanning tree cannot be formed ";  
    }  
    return 0;  
    }  
    int checkCycle (Edge e, int\* path) {  
    int u = e.u, v = e.v;  
    while (path[u] > 0)  
    u = path[u];  
    while (path[v] > 0)  
    v = path[v];  
    if (u != v) {  
    path[u] = v;  
    return 1;  
    }  
  
    return 0;  
  
    }

OUTPUT:

Enter the number of vertices in the graph: 3

Enter the number of edges in the graph: 2

Enter 2 vertices and weight of edge 1

First vertex: 1

Second vertex: 2

Weight: 5

Enter 2 vertices and weight of edge 2

First vertex: 2

Second vertex: 3

Weight: 17

Edge (1, 2) with weight 5 is selected

Edge (2, 3) with weight 17 is selected

Minimum spanning tree is22