Problem G
Networking

Input File: G.DAT

Program Source File: G.PAS or G.C or G.CPP or G.JAVA

You are assigned to design network connections between certain points in a wide area. You are given a set of points in the area, and a set of possible routes for the cables that may connect pairs of points. For each possible route between two points, you are given the length of the cable that is needed to connect the points over that route. Note that there may exist many possible routes between two given points. It is assumed that the given possible routes connect (directly or indirectly) each two points in the area.

Your task is to design the network for the area, so that there is a connection (direct or indirect) between every two points (i.e., all the points are interconnected, but not necessarily by a direct cable), and that the total length of the used cable is minimal.

Input: The input file consists of a number of data sets. Each data set defines one required network. The first line of the set contains two integers: the first defines the number P of the given points, and the second the number R of given routes between the points. The following R lines define the given routes between the points, each giving three integer numbers: the first two numbers identify the points, and the third gives the length of the route. The numbers are separated with white spaces. A data set giving only one number P=0 denotes the end of the input. The data sets are separated with an empty line.

The maximal number of points is 50. The maximal length of a given route is 100. The number of possible routes is unlimited. The nodes are identified with integers between 1 and P (inclusive). The routes between two points *i* and *j* may be given as *i j* or as *j i*.

Output: For each data set, print one number on a separate line that gives the total length of the cable used for the entire designed network.

Sample Input:
```
1 0

2 3
1 2 37
2 1 17
1 2 68

3 7
1 2 19
2 3 11
3 1 7
1 3 5
2 3 89
3 1 91
1 2 32

5 7
1 2 5
2 3 7
2 4 8
4 5 11
3 5 10
1 5 6
4 2 12

0
```

Sample Output:
```
0
17
16
26
```