Pricey Power

SCANA just wasted **11 billion dollars** on a nuclear facility before the project went under. Now SCANA is in trouble. Can you help SCANA cut costs fast?

SCANA provides electricity to houses all across their expansive power networks. The networks connect houses to one another through power lines. Each month, SCANA pays a maintainence fee for each power line connecting two houses. As long as each house has some power line path to another house on the grid, it is connected to the grid. SCANA cannot afford to keep maintaining every power line, so some will have to be disconnected. Can you write a program to compute the minimum cost per month that SCANA must spend to maintain its power network while keeping every house connected to the grids?

Language Note: The author's Python 3 solution took less than 3 seconds on the hardest test case.

Input Format

The first line contains T, the number of test cases.

Each test case starts with a number N, where N is the number of houses to which SCANA provides power.

The next $N-1$ lines of the test case are formatted as follows:

$x_i \; e \; x_j \; w_{ij} \; x_k \; w_{ik} \; \ldots$

Where x_i is the label of some house, e is the number of power lines connecting x_i to other houses, minus the number of power lines connecting x_i to other houses that have already been accounted for on previous lines (see note), and then each $x_j \; w_{ij}$ pair indicates that a power line connects x_i to x_j with monthly maintainence cost w_{ij}.

Note: *This is a bit confusing. The power lines are given in ascending order of house label, for example consider the following input*

```
1
3
0 2 1 5 2 6
1 1 2 4
2 0
```

The first line means that there is one test case. There are 3 houses in this test case, indicated by the second line.

The third line says that house 0 has two *unaccounted-for* power lines connecting it to other houses. One powerline connects house 0 to house 1 and has a monthly maintainence cost of 5. Another powerline connects house 0 to house 2, and has a monthly maintainence cost of 6.

The fourth line says that house 1 has one *unaccounted-for* power line connecting it to other houses. This powerline connects houses 1 and 2 and has a monthly maintainence cost of 4. There is a power line connecting house 0 and house 1, but we already printed it out on the previous line so we do not print it out again.

The fifth line says that house 2 has no *unaccounted-for* power lines. Remember that house 2 has two power lines connecting to it, but the power line connecting house 0 and house 2 was already accounted for on line three, and the power line connecting house 1 to house 2 was already accounted for on line four, so we have no more power lines to print out. This means that every single power line is printed out only one time per test case.

NOTE: Nowhere does this description specify that all houses are connected to the same grid. A given test case may contain more than one grid. For example:

```
1
4
0 1 1 5
1 0
2 1 3 5
3 0
```

is a valid input. House 0 and house 1 are connected, and house 2 and house 3 are connected, but there are no connections between these two components. This indicates that there are two power grids. If there was a connection between house 0 and house 2, that would imply that there is only one power grid and we would not be able to get rid of the edge between house 0 and house 2.

Constraints

$1 \leq T \leq 15$

$1 \leq N \leq 2^8$

$1 \leq w_{ij} < 2^{32}$

Output Format

T lines, with each line containing an integer indicating the minimum total cost that SCANA must pay per month to maintain their networks.

Sample Input 0

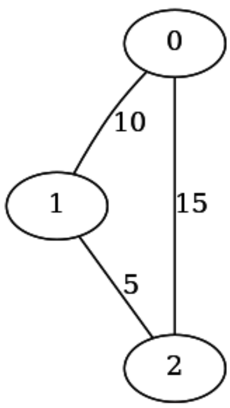
```
2
3
0 2 1 10 2 15
1 1 2 5
0 0
5
0 4 1 1 2 1 3 1 4 1
1 3 2 1 3 1 4 1
2 2 3 1 4 1
3 1 4 1
4 0
```

Sample Output 0

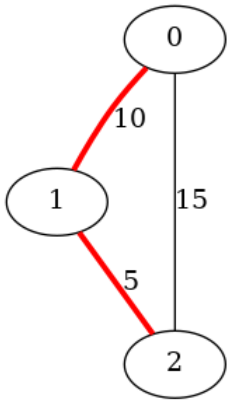
```
15
4
```

Explanation 0

Consider the first of two test cases above. Our power grid looks something like this:

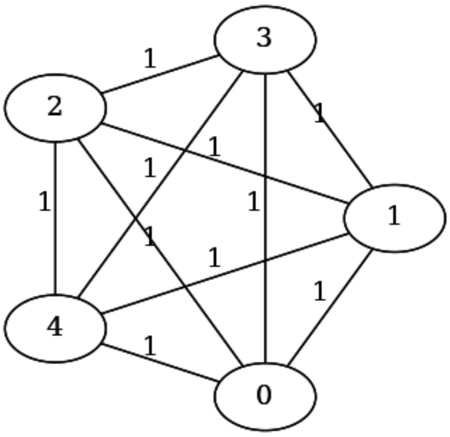


Highlighting the power lines we need to keep to ensure power reaches all houses, we get the following:

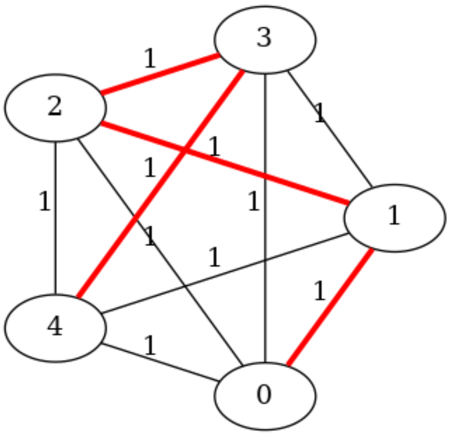


Thus if we add the monthly costs for the lines highlighted above, we get $5 + 10 = 15$, our answer. Note that if we had used any other combination of power lines in this example, the total monthly cost would have been more that 15, so 15 successfully minimizes our monthly cost.

For the second case in this input, our power network looks something like this:



Highlighting the power lines we need to keep to ensure power reaches all houses, we get the following:



Adding up the costs of the highlighted lines, we get $1 + 1 + 1 + 1 = 4$. Note that the highlighting shown here is not the only solution, but no solution produces a monthly cost lower than 4, so 4 is our answer.

Also note that not every house is connected to every other house in this test case. We assume each connected component in an input is part of its own power grid.

Sample Input 1

```
1
6
0 2 1 5 2 5
1 1 2 5
2 0
```

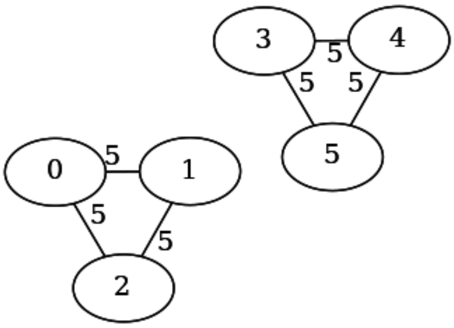
```
3 2 4 5 5 5
4 1 5 5
5 0
```

Sample Output 1

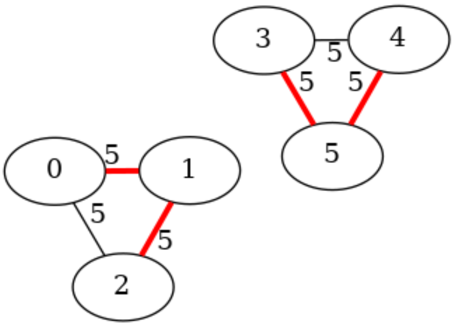
```
20
```

Explanation 1

For this test case, our power network looks something like this:



One set of powerlines that minimizes the monthly cost might look like this:



There are multiple sets of power lines that minimize the monthly cost, but $5+5+5+5=20$ is the lowest possible monthly cost, so our solution is 20.

NOTE: The costs for each power line are very rarely all the same, see the first test case above as an example.