Programming Project of INF 421:

Optimal Tree Labeling

Proposed by Hang Zhou (hzhou@lix.polytechnique.fr)

Problem Description

We have a tree $T = (V, E)$, where V is the set of vertices and E is the set of edges. Every vertex $v \in V$ has a *label* $L(v)$, which is a subset of $\{A, B, \ldots, Z\}$.[1] For an edge $e = (u, v)$, we define the weight $w(e)$ as the Hamming distance between $L(u)$ and $L(v)$, that is:

$$w(e) = \Big|\{X \mid X \in L(u) \text{ and } X \notin L(v)\} \cup \{X \mid X \notin L(u) \text{ and } X \in L(v)\}\Big|.$$

For example, for an edge $e = (u, v)$ with $L(u) = \{A, B, C, D, E\}$ and $L(v) = \{B, D, E, F\}$, we have $w(e) = \big|\{A, C, F\}\big| = 3$.

In the *optimal tree labeling* problem, you are given an unrooted tree with the labels of all the leaf vertices. The goal is to find some way to label all the non-leaf vertices, such that the total weight of all the edges in the tree is minimized under this labeling.

For example, in Fig. 1, there is an input tree with the labels of the leaf vertices. Figs. 2 and 3 correspond to two possible labelings (among others) of the non-leaf vertices. The total edge weight under the labeling in Fig. 2 is 6. The total edge weight under the labeling in Fig. 3 is 4. One can verify that the labeling in Fig. 3 achieves the minimum total edge weight on this instance, and is thus optimal.

Programs should be written in either C++ or Java.

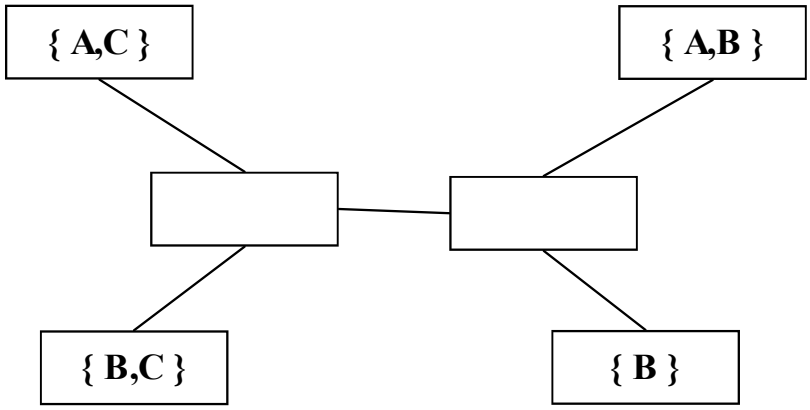

Figure 1: Input tree with labels on the leaf vertices.

[1] $L(v)$ is allowed to be an empty set.

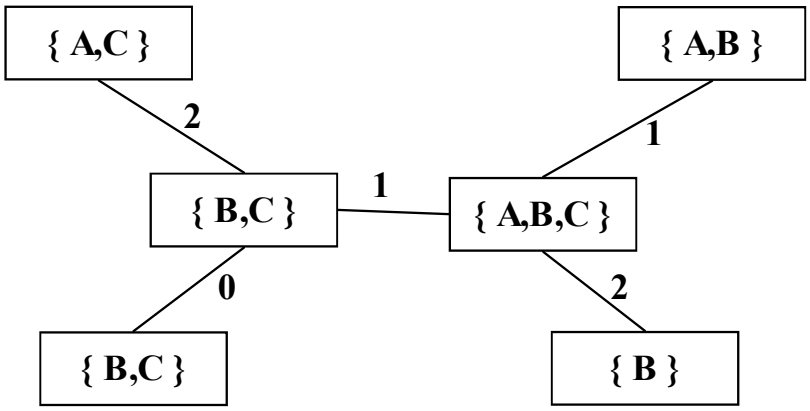

Figure 2: One possible labeling with total edge weight 6.

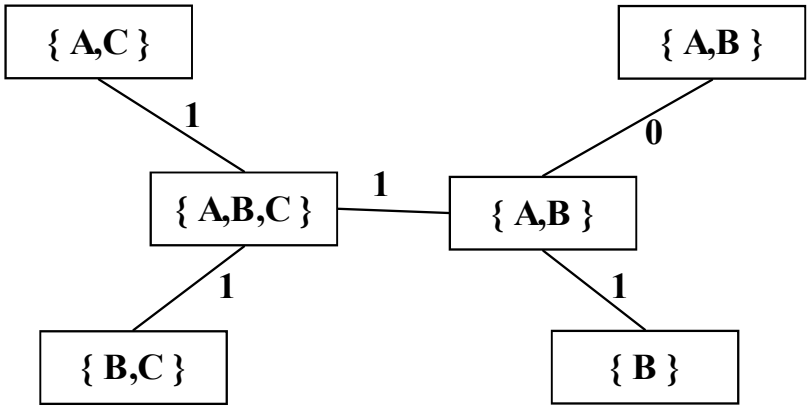

Figure 3: Another possible labeling with total edge weight 4.

Tests

There are 10 independent tests. The input files `labeling.1.in`,...,`labeling.10.in` can be downloaded by clicking here.

Input Format. Your program reads from an input file `labeling.k.in` (where $1 \leq k \leq 10$), which contains the following:

- On the first line, there are two integers N and L, indicating the number of all vertices in the tree and the number of leaf vertices in the tree. We guarantee that $N \leq 50000$. The vertices in the tree are identified by integers from 1 to N.
- On the next $N - 1$ lines, each line contains two integers u and v, representing an edge connnecting vertex u and vertex v in the tree.
- On the next L lines, each line contains an integer u and a string S, representing a leaf

vertex u and its label S. If the label of u is an empty set, then S is denoted by \$ in the input.

Output Format. For each of the 10 tests, your program should output one integer, which is the minimum total edge weight achievable. You should include in your report the output on each test, as well as the running time of your program on that test.

Sample Input 1.

```
6 4
1 3
2 4
3 4
5 3
4 6
1 AC
2 AB
5 BC
6 B
```

Sample Output 1.

```
4
```

Remark. This sample corresponds to the input in Fig. 1 and the optimal solution in Fig. 3.

Sample Input 2.

```
4 3
1 4
2 4
4 3
1 A
2 B
3 $
```

Sample Output 2.

```
2
```

Remark. The optimal solution in this sample can be achieved by labeling the 4-th vertex in the tree with an empty set.