
tasks

task	formule	lubenica
source file	formule.pas formule.c formule.cpp	lubenica.pas lubenica.c lubenica.cpp
input data	stdin	
output data	stdout	
time limit (Intel Celeron 2.66Ghz)	1 sec	3 sec
memory limit (heap)	32 MB	
memory limit (stack)	8 MB	16 MB
points	100	100
	200	

formule

Consider mathematical formulas defined as follows:

- a **constant** is a lowercase letter of the English alphabet ('a'-'z')
- a **variable** is the digit '0' or '1'
- a **function** is an uppercase letter of the English alphabet ('A'-'Z')
- a **formula** is
 - a constant
 - or**
 - a variable
 - or**
 - a function(formula,formula)

For example 'a', '0', 'F(a,F(a,a))' and 'G(F(0,1),G(a,1))' are formulas, while 'A', 'a(9)' and 'F(G(),a)' are not.

Formulas that **don't contain** any variables are called **basic formulas**.

You are given two formulas. Write a program that will **substitute** every occurrence (in both formulas) of some variable with some **basic formula** so that, after substitution, the two formulas become **identical**.

If a variable appears more than once in one or both formulas then **each occurrence** has to be substituted with **the same** basic formula.

input data

Two lines of input contain two formulas. Each will contain at most 100 characters.

output data

The first line of output should contain the substitution for the variable '0', and the second line should contain the substitution for the variable '1'.

In both lines the format should be '**variable=basic formula**'. Even if a variable doesn't appear in formulas at all, you should still output a substitution for it.

Note: the test data will be such that a solution, although not necessarily unique, always exists.

examples

input

0
a

output

0=a
1=b

input

F(0,G(a,1))
F(c,G(a,F(0,0)))

output

0=c
1=F(c,c)

input

F(G(1,a),G(1,a))
F(0,0)

output

0=G(b,a)
1=b

lubenica

The traffic network in a country consists of N cities (labeled with integers from 1 to N) and $N-1$ roads connecting the cities. There is a **unique** path between **each pair of different cities**, and we know the exact length of each road.

Write a program that will, **for each of the K given pairs of cities**, find the length of the **shortest** and the length of the **longest** road on the path between the two cities.

input data

The first line of input contains an integer N , $2 \leq N \leq 100\,000$.

Each of the following $N-1$ lines contains three integers A , B and C meaning that there is a road of length C between city A and city B . The length of each road will be a positive integer less than or equal to $1\,000\,000$.

The next line contains an integer K , $1 \leq K \leq 100\,000$.

Each of the following K lines contains two different integers D and E – the labels of the two cities constituting one query.

output data

Each of the K lines of output should contain two integers – the lengths from the task description for the corresponding pair of the cities.

examples

input

```
5
2 3 100
4 3 200
1 5 150
1 3 50
3
2 4
3 5
1 2
```

output

```
100 200
50 150
50 100
```

input

```
7
3 6 4
1 7 1
1 3 2
1 2 6
2 5 4
2 4 4
5
6 4
7 6
1 2
1 3
3 5
```

output

```
2 6
1 4
6 6
2 2
2 6
```

input

```
9
1 2 2
2 3 1
3 4 5
2 7 4
1 5 3
5 6 1
5 9 2
1 8 3
5
6 9
7 8
9 4
1 2
7 3
```

output

```
1 2
2 4
1 5
2 2
1 4
```