Denis came to Nastya and discovered that she was not happy to see him... There is only one chance that she can become happy. Denis wants to buy all things that Nastya likes so she will certainly agree to talk to him.

The map of the city where they live has a lot of squares, some of which are connected by roads. There is exactly one way between each pair of squares which does not visit any vertex twice. It turns out that the graph of the city is a tree.

Denis is located at vertex 1 at the time 0. He wants to visit every vertex at least once and get back as soon as possible.

Denis can walk one road in 1 time. Unfortunately, the city is so large that it will take a very long time to visit all squares. Therefore, Denis took a desperate step. He pulled out his pocket time machine, which he constructed in his basement. With its help, Denis can change the time to any non-negative time, which is less than the current time.

But the time machine has one feature. If the hero finds himself in the same place and at the same time twice, there will be an explosion of universal proportions and Nastya will stay unhappy. Therefore, Denis asks you to find him a route using a time machine that he will get around all squares and will return to the first and at the same time the maximum time in which he visited any square will be minimal.

Formally, Denis's route can be represented as a sequence of pairs: $\{v_1, t_1\}, \{v_2, t_2\}, \{v_3, t_3\}, \ldots, \{v_k, t_k\}$, where v_i is number of square, and t_i is time in which the boy is now.

The following conditions must be met:

- The route starts on square 1 at time 0, i.e. $v_1 = 1, t_1 = 0$ and ends on the square 1, i.e. $v_k = 1$.
- All transitions are divided into two types:
  1. Being in the square change the time: $\{v_i, t_i\} \to \{v_{i+1}, t_{i+1}\} : v_{i+1} = v_i, 0 \leq t_{i+1} < t_i$.
  2. Walk along one of the roads: $\{v_i, t_i\} \to \{v_{i+1}, t_{i+1}\}$. Herewith, v_i and v_{i+1} are connected by road, and $t_{i+1} = t_i + 1$
- All pairs $\{v_i, t_i\}$ must be different.
- All squares are among $v_1, v_2, \ldots, v_k$.

You need to find a route such that the maximum time in any square will be minimal, that is, the route for which $\max(t_1, t_2, \ldots, t_k)$ will be the minimum possible.

Input

The first line contains a single integer n $(1 \leq n \leq 10^5)$ — the number of squares in the city.

The next $n - 1$ lines contain two integers u and v $(1 \leq v, u \leq n, u \neq v)$ - the numbers of the squares connected by the road.

It is guaranteed that the given graph is a tree.

Output

In the first line output the integer k $(1 \leq k \leq 10^6)$ — the length of the path of Denis.

In the next k lines output pairs v_i, t_i — pairs that describe Denis's route (as in the statement).

All route requirements described in the statements must be met.

It is guaranteed that under given restrictions there is at least one route and an answer whose length does not exceed 10^6. If there are several possible answers, print any.

Example

input

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

output

```
13
1 0
2 1
3 2
3 1
2 2
4 3
4 1
5 2
5 1
4 2
2 3
2 0
1 1
```

She has a mysterious set which consists of tiles (this set can be empty). Each tile has an integer value between 1 and n, and **at most n tiles** in the set have the same value. So the set can contain at most n^2 tiles.

You want to figure out which values are on the tiles. But Yui is shy, she prefers to play a guessing game with you.

Let's call a set consisting of **three** tiles *triplet* if their values are the same. For example, $\{2, 2, 2\}$ is a triplet, but $\{2, 3, 3\}$ is not.

Let's call a set consisting of **three** tiles *straight* if their values are consecutive integers. For example, $\{2, 3, 4\}$ is a straight, but $\{1, 3, 5\}$ is not.

At first, Yui gives you the number of triplet subsets and straight subsets of the initial set respectively. After that, you can insert a tile with an integer value between 1 and n into the set **at most n times**. Every time you insert a tile, you will get the number of triplet subsets and straight subsets of the current set as well.

Note that two tiles with the same value are treated different. In other words, in the set $\{1, 1, 2, 2, 3\}$ you can find 4 subsets $\{1, 2, 3\}$.

Try to guess the number of tiles in the initial set with value i for all integers i from 1 to n.

Input

The first line contains a single integer n ($4 \le n \le 100$).

The second line contains two integers which represent the number of triplet subsets and straight subsets of the initial set respectively.

Output

When you are ready to answer, print a single line of form "$!\ a_1\ a_2\ \ldots\ a_n$" ($0 \le a_i \le n$), where a_i is equal to the number of tiles in the initial set with value i.

Interaction

To insert a tile, print a single line of form "$+\ x$" ($1 \le x \le n$), where x is the value of the tile you insert. Then you should read two integers which represent the number of triplet subsets and straight subsets of the current set respectively.

After printing a line, do not forget to flush the output. Otherwise, you will get `Idleness limit exceeded`. To do this, use:

- `fflush(stdout)` or `cout.flush()` in C++;
- `System.out.flush()` in Java;
- `flush(output)` in Pascal;
- `stdout.flush()` in Python;
- see documentation for other languages.

You will get `Wrong answer` if you insert more than n tiles.

Hacks

To make a hack you should provide a test in such format:

The first line contains a single integer n ($4 \le n \le 100$).

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le n$) — a_i is equal to the number of tiles with value i in the set.

Example

input

```
5
1 6
2 9
5 12
5 24
6 24
```

output

```
+ 1
+ 1
+ 2
+ 5
! 2 1 3 0 2
```

Note

In the first test, the initial set of tiles is $\{1, 1, 2, 3, 3, 3, 5, 5\}$. It has only one triplet subset $\{3, 3, 3\}$ and six straight subsets, all equal to $\{1, 2, 3\}$. After inserting a tile with value 1 the set of tiles will be $\{1, 1, 1, 2, 3, 3, 3, 5, 5\}$ and will have two triplet subsets $\{1, 1, 1\}$, $\{3, 3, 3\}$ and nine straight subsets, all equal to $\{1, 2, 3\}$.