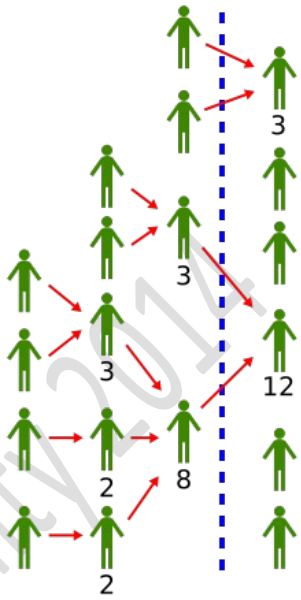
President Election of T Republic



The people of T Republic have decided to elect their president using a democracy voting process.

The voting rules are as below:

- Every voter has one vote.
- A voter can vote for one candidate, or delegate their vote to another voter.
- If the delegated voter does not vote, or a circular delegation is formed, all these votes are invalid.
- A voter who does not vote is not counted.

Three candidates for president are nominated: Steve, Jenny, and Eva.

Create a program that calculates the number of votes for each candidate and the number of votes that are invalid.

Note: There are at most **20 million** eligible voters in T Republic.

Input:
Every voter is assigned a unique integer voter-ID (0 < voter-ID < 20,000,000). The vote data is sorted by voter-ID, one voter per line, in the following format:
<voter-ID> <space> (<delegated voter-ID> or <candidate-name>) \n

Consider the following example set of data:

```
1 3
2 Jenny
3 10
4 Eva
5 Steve
7 7
8 6
10 3
11 5
```

- Voter 1 -> 3 -> 10 -> 3 is a circular delegation. Voter 1's vote is invalid.
- Voter 2's vote is Jenny.
- Voter 3 -> 10 -> 3 is a circular delegation. Voter 3's vote is invalid.
- Voter 4's vote is Eva.
- Voter 5's vote is Steve.
- Voter 6 does not vote.
- Voter 7 -> 7 is a circular delegation. Voter 7's vote is invalid.
- Voter 8 -> 6, and 6 does not vote. Voter 8's vote is invalid.
- Voter 9 does not vote.
- Voter 10 -> 3 -> 10 is a circular delegation. Voter 10's vote is invalid.
- Voter 11 -> 5 -> Steve. Voter 11's vote is Steve.

There are 9 lines of input, so the total number of votes is 9. Your program should read the vote data from the standard input. Every voter only has one vote, so your program will not see a duplicated voter-ID in the input.

Output:
Following the example above, your program should write the results of the vote using standard output in the following order:

```
Steve 2
```

```
Jenny 1
Eva 1
Invalid 5
```