

Problem D

Dodecaphony


Problem ID: dodecaphony

CPU Time limit: 1 second

Memory limit: 1024 MB

Author: Martin Tran

Source: Calgary Collegiate
Programming Contest 2020

License: 

In the past century, a new style of music composition has emerged. Unlike more traditional methods based on keys and chords, the technique known as *dodecaphony* focuses on using all twelve notes equally. As a quick reminder, the twelve notes, in ascending order are,

$$C, C\sharp, D, D\sharp, E, F, F\sharp, G, G\sharp, A, A\sharp, B$$

The sequence then wraps around so that the next note after B is C and so on. For this problem, we'll ignore equivalent notations that use flats, double sharps, or double flats.

Each successive note above is considered one *semitone* away from the next. Now in our simplified version of dodecaphony, a melody is a permutation of the previous melody by one of three relations.

First, we have *transposition*, where each note has been shifted up by n semitones. A *retrograde* is when the notes have their order reversed. Finally we have *inversion* about the first note of the melody. With inversions, the first note doesn't change, but the rest of the notes are inverted such that the the interval (number of semitones) between that note and the first note is negated.

For example, if F is our first note, and we want to invert an $A\sharp$, which is 5 semitones higher, the inverted note would be a C , which is 5 semitones lower. Note that the first note in an inverted melody is always just the first note of the original melody.

Given two melodies, can you tell what relation the second has to the first?

Input

The first line contains a single integer $2 \leq l \leq 50$, the number of notes in each melody.

The next two lines each contain l space separated notes. It is guaranteed that each note will be one of the twelve listed above.

Output

Output on a single line "Transposition" if the second melody is a transposition of the first, "Retrograde" if the second melody is the first melody reversed, "Inversion" if the second melody is an inversion of the first melody, else "Nonsense" if it is none of the other cases.

If the second melody satisfies more than one relation, output the first valid relation in the order of "Transposition", then "Retrograde", then "Inversion".

Sample Input 1

```
3
C E G
D F# A
```

Sample Output 1

```
Transposition
```

Sample Input 2

```
7
C C G G A A G
C C F F D# D# F
```

Sample Output 2

```
Inversion
```

Sample Input 3

```
7
A B C D E F G
G F E D C B A
```

Sample Output 3

```
Retrograde
```

Sample Input 4

```
5
C D E F G
C C C C C
```

Sample Output 4

Nonsense