

## Problem 1. 過半元素

(Time Limit: 2 seconds)

### 問題描述：

一個含  $N$  個整數的序列中，出現頻率超過  $N/2$  的整數稱為『過半元素』。  
寫一個程式求一整數序列是否有過半元素。

### 輸入說明：

一整數序列以空白鍵隔開數字； $0 < N < 11$ ；序列中的數字都是整數；輸入包含多行，一行為一筆測資。

### 輸出說明：

若有過半元素，輸出該數；否則輸出“NO”。

### 範例：

Sample Input:	Sample Output:
23 34 34 5 5 34 34	34
1 -1 1 -1 1 -1 1 -1 1	1
1 2 3 4 5 6 7 8 9 10	NO
5 5 5 5 5 5 5 5 5 5 5	5
543 7873 -9 27 -88	NO

## Problem 2. 未知數

(Time Limit: 1 second)

### 問題描述：

輸入一個等式，式子中放入一個未知數，例如  $2 + x = 15$ ， $x$  即為 13。現在想藉由程式協助解答未知數，也就是藉由簡單的加減乘除運算，求得等式中的未知數。

### 輸入說明：

輸入若干筆測資，每筆測資為一個等式式子，包含 2 個數字 ( $\geq 0$ )、1 個未知數  $x$ 、1 個等號、1 個加減乘除符號（以 “+”、“-”、“\*”、“/” 符號表示），未知數可放在等號後或加減乘除符號左右皆可。每個運算元與運算子都以一個空白間隔，例如  $2 * x = 6$ 。

### 輸出說明：

每筆測資輸出未知數  $x$  的答案數字於一行，無條件捨去輸出到小數一位。

### 範例：

Sample Input:	Sample Output:
$x / 10 = 20$	200.0
$4 / x = 3$	1.3

## Problem 3. 字串切割

(Time Limit: 2 seconds)

### 問題描述：

切割一字串，字串中以三種符號作為文字間的區隔：“:”，“,”，“;”。列印出切割後的結果。

### 輸入說明：

第一行需輸入測資的筆數，再輸入任意字串並以“:”，“,”，“;”三種符號做間隔。

### 輸出說明：

每一筆測資，將以上三種符號與符號間的字串切割，先輸出"Tokens found:\n"，再一行行的輸出切割出來的字串(Token)。

### 範例：

Sample Input:	Sample Output:
2 abc,def:ghijk,lmn word:office;excel,powerpoint;project	Tokens found: abc def ghijk lmn Tokens found: word office excel powerpoint project

## Problem 4. 羅馬數字系統

(Time Limit: 3 seconds)

### 問題描述：

羅馬數字共有 7 個，即 I(1) ， V(5) ， X(10) ， L(50) ， C(100) ， D(500) ， M(1000) 。按照下述的規則可以表示任意正整數。需要注意的是羅馬數字中沒有「0」。

- 重複數次：1 個羅馬數字重複幾次，就表示這個數的幾倍。
- 右加左減：在一個較大的羅馬數字的右邊記上一個較小的羅馬數字，表示大數字加小數字。在一個較大的羅馬數字的左邊記上一個較小的羅馬數字，表示大數字減小數字。但是，左減不能跨越一個位數。比如，99 不可以用 IC 表示，而是用 XCIX 表示（等同於阿拉伯數字每位數字分別表示）。例如：999=900+90+9=CMXCIX。

此外，左減數字必須 1 位，比如 8 寫成 VIII ，而非 IIX。

同理，右加數字不能超過 3 位，比如 14 寫成 XIV ，而非 XIII 。

羅馬數字與十進位數字對照表：

羅馬字	數字	羅馬字	數字	羅馬字	數字	羅馬字	數字
I	1	XI	11	XXI	21	C	100
II	2	XII	12	XXIX	29	CI	101
III	3	XIII	13	XXX	30	CC	200
IV	4	XIV	14	XL	40	D	500
V	5	XV	15	XLVIII	48	DC	600
VI	6	XVI	16	IL	49	CM	900
VII	7	XVII	17	L	50	M	1000
VIII	8	XVIII	18	LX	60		
IX	9	XIX	19	XC	90		
X	10	XX	20	XCVIII	98		

**輸入說明：**

第一列是一個整數表示有幾個測試案例，其後每一列是一個測試案例，亦即每一列是一個十進制阿拉伯數字，數字範圍 1~999。

**輸出說明：**

依序逐列輸出每一個測試案例轉換後的羅馬數字。。

**範例：**

Sample Input:	Sample Output:
2	LXXV
75	XCIX
99	

## Problem 5. Find the Sequence Pattern

(Time Limit: 3 seconds)

### Problem Description

A sequence of pattern is a way of arranging the sequences, which are commonly represented both graphically and in text format to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. For instance, the aligned sequences of DNA , RNA , or protein in bioinformatics are typically represented as rows within a matrix so that identical or similar characters are aligned in successive columns. Sequence alignments are also appeared for non-biological sequences, such as calculating the edit distance cost between strings in a natural language or in financial data.

There are many rules for sequencing a row of data. In this problem, you are required to find the longest common base sequence in the two strands, which can be represented as strings consisting of the letters 'a', 'b', 'c', 'd' and 'e'. For instance, the strands 'abcde' and 'babcd' contain the longest common sequences as 'abcd'.

### Input Format

For each test case, two strings are contained in two lines with at most 100 characters-long for each input strings.

### Output Format

Each test case output the longest common sequences in lexicographical order. If there isn't any common sequence between the two strings, 'No common sequence!' is printed.

### Example

Sample Input:	Sample Output:
abccedda dabcdbdad	abc
bdbdeed cacaccac	No common sequence!