

## Problem 1. 中序轉後序

(Time Limit: 1 seconds)

### 問題描述:

人們平常使用的運算式，是將運算元放在運算子兩旁，例如  $a + b / d$  這樣的式子，這稱為中序 (infix) 表示式；後序是將運算元放在運算子後方，例如  $abd/+$ 。

請寫一個程式幫把數學運算式，中序轉後序以後的答案印出來。

### 輸入說明

第一列有一個正整數  $n$  ( $1 \leq n \leq 10$ ) 代表共有  $n$  組測試案例。

每組測試案例有一個數學運算式 (運算式含空格總長度不超過 50 個字元)。數學運算式中每個運算元及運算子之間均有一個空格。運算元均為正整數，運算子只包含四則運算 (即： $+$ ,  $-$ ,  $*$ ,  $/$ )，運算式為一般正常的運算式 (即不會出現連續兩個數字或符號的情況)，運算法則為先乘除後加減，且括號優先於其他運算。

### 輸出說明:

將該筆測試資料之後序法輸出於一行，數字與符號、符號與符號以及數字與數字之間，以一個空格隔開。

注意：每行輸出最後都會有換行符號。

### 範例

| Sample1 Input:        | Sample1 Output:   |
|-----------------------|-------------------|
| 4                     | 4 5 * 2 +         |
| 4 * 5 + 2             | 2 3 + 3 * 5 - 2 / |
| ((2 + 3) * 3 - 5) / 2 | 1 3 + 2 *         |
| (1 + 3) * 2           | 2 3 * 5 + 9 * 3 / |
| (2 * 3 + 5) * 9 / 3   |                   |

## Problem 2. 提款機程式

(Time Limit: 1 seconds)

### 問題描述:

請撰寫一支程式，模擬提款機的操作，動作包含，輸入帳密、提款、餘款查詢。

假設目前擁有帳戶資料：

| 帳號  | 密碼  | 餘款    |
|-----|-----|-------|
| 123 | 456 | 9000  |
| 456 | 789 | 5000  |
| 789 | 888 | 6000  |
| 336 | 558 | 10000 |
| 775 | 666 | 12000 |
| 566 | 221 | 7000  |

### 輸入說明

第一行輸入兩個字串，以空格隔開，依序代表帳號及密碼 ( 字串不超過 10 個字元 )。

L 表示提款，空格後輸入要提款的金額。

S 表示餘額查詢。

E 表示結束。

### 輸出說明:

如帳密任一項錯誤，請輸出 ERROR，否則顯示 OK。

如提款金額(L)會扣除餘款，顯示 OK。

但提款金額(L)>餘款，會輸出 ERROR，餘額不會改變。

如餘額查詢(S)輸出餘款。

E 表示結束，輸出 BYE。

在登入狀態時，輸入奇怪的指令也顯示 ERROR。

注意：每行輸出最後都會有換行符號。

### 範例

|                |                 |
|----------------|-----------------|
| Sample1 Input: | Sample1 Output: |
|----------------|-----------------|

|  |  |
|--|--|
| 123 456<br>S<br>L 200<br>S<br>123 456<br>E | OK<br>9000<br>OK<br>8800<br>ERROR<br>BYE |
| <b>Sample2 Input:</b>                      | <b>Sample2 Output:</b>                   |
| 456 123<br>999 999<br>566 221<br>E         | ERROR<br>ERROR<br>OK<br>BYE              |
| <b>Sample3 Input:</b>                      | <b>Sample3 Output:</b>                   |
| 775 666<br>L 99999<br>L 200<br>L 200<br>E  | OK<br>ERROR<br>OK<br>OK<br>BYE           |

## Problem 3. 高頻字元

(Time Limit: 1 seconds)

### 問題描述:

R 博士使用電波望遠鏡對準外太空，想接收地球外訊號進行分析，假設這些訊號接收到後會個別轉換成對應的字元，最後輸出一段字串，R 博士想從這字串中找出出現頻率最高的字元進行分析。字串中可能包含大小寫英文字母、數字、空白、標點符號等。

### 輸入說明

第一列有一個正整數  $n$  ( $1 \leq n \leq 10$ ) 代表共有  $n$  組測試案例。

每組測試案例有一字串(字串長度不超過 100)，含各種大小寫與標點符號。

### 輸出說明:

請輸出出現頻率最高的字元。(不會有多個答案)

注意：每行輸出最後都會有換行符號。

### 範例

| Sample1 Input:   | Sample1 Output: |
|--|-----------------|
| 2<br>Eryhds. 3yeeflm, cefdsefe! e<br>Yseghs!!ds! cALsdL PAdsfSS!!! | e<br>!          |

## Problem 4. 閻羅王

(Time Limit: 1 seconds)

### 問題描述:

但是近年來亡者越來越多，而上天堂的人數每一年都有限制，閻羅王會將亡者的生日數字各別相加總和  $N$ ，接著將  $N$  本身以外的所有因數加起來的總和為  $S$ ，接著將  $S$  加上此人的善惡評價指標得到  $G$ ，如果  $N > G$  則立刻下地獄，如果  $N = G$  則立刻上天堂，如果  $N < G$  則具有進入天堂的資格，但是需要等待  $G - N$  年。

假設亡者的生日為 1911/01/01，善惡評價指標為 5，  
 $N = 1 + 9 + 1 + 1 + 0 + 1 + 0 + 1 = 14$ ，14 除了自身以外的因數有 1、2、7，  
 $S = 1 + 2 + 7 = 10$ ， $G = 10 + 5 = 15$ ，因為  $N < G$ ，因此亡者可以上天堂。但是  $G - N = 15 - 14 = 1$ ，因此亡者還需要等待 1 年。

### 輸入說明

第一列有一個正整數  $n$  ( $1 \leq n \leq 10$ ) 代表共有  $n$  組測試案例。

每組測試案例輸入一個 9 碼整數，前 8 碼表示生日，第 9 碼表示善惡評價指標。

e.g. 生日 1992/01/01 則輸入 19920101。

### 輸出說明:

顯示 0 為立即上天堂、顯示 -1 為立即下地獄，如需等待上天堂者，請顯示等待年數。

注意：每行輸出最後都會有換行符號。

### 範例

| Sample1 Input: | Sample1 Output: |
|----------------|-----------------|
| 2              | 1               |
| 198711232      | -1              |
| 196504090      |                 |

## Problem 5. 正方形

(Time Limit: 1 seconds)

### 問題描述:

阿布某天想用身邊的木棍圍起一個正方形土地做為耕種，因為這些木棍的長短都不一樣，所以阿布也不知道這些木棍是否可以剛好圍出一個正方形，請你幫忙阿布算出這些棍子是否可以剛好圍起一個正方形耕地。其中正方形的邊長可以多根木棍。

### 輸入說明

第一列有一個正整數  $n$  ( $1 \leq n \leq 10$ ) 代表共有  $n$  組測試案例。

每組測試案例輸入一列數列，第一個整數為  $m$  ( $4 \leq m \leq 20$ )，代表棍子的數目。接下來的  $m$  個整數分別代表這  $m$  根棍子的長度，每支棍子的長度介於 1 到 1000 之間。

### 輸出說明:

如果這些棍子可以連成一個正方形，輸出 yes。否則輸出 no。

### 範例

| Sample1 Input:   | Sample1 Output: |
|------------------|-----------------|
| 2                | yes             |
| 4 1 1 1 1        | no              |
| 5 10 20 30 40 50 |                 |

## Problem 6. 地雷

(Time Limit: 1 seconds)

### 問題描述:

例如下面  $n \times n$  大小的地圖：

|   |   |   |   |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |

上述代表

炸彈：(1,1), (1,3), (1,4), (2,1), (3,1), (3,2), (3,4), (4,2), (4,3)

非炸彈：(1,2), (2,2), (2,3), (2,4), (3,3), (4,1), (4,4)

請找出非炸彈的格子中周圍八格有幾顆炸彈。

|   |   |   |
|---|---|---|
| 1 | 1 | 0 |
| 0 | x | 1 |
| 1 | 1 | 0 |

上述 x 的值應為 5

若 x 本身就是代表一顆炸彈，則值應為 0

### 輸入說明

第一行輸入一個正整數  $n$  ( $3 \leq n \leq 9$ ) 代表地圖大小。

接下來會有  $n$  行，每行有  $n$  個正整數表示地圖上每格的炸彈數量，中間以空白隔開。

### 輸出說明:

輸出每個格子中周圍八格有幾顆炸彈。若本身就是代表一顆炸彈，則值應為 0。

### 範例

| Sample1 Input: | Sample1 Output: |
|----------------|-----------------|
| 5              | 0 2 3 0 0       |
| 1 0 0 1 1      | 3 0 4 0 4       |
| 0 1 0 1 0      | 0 3 0 4 0       |
| 1 0 1 0 1      | 2 4 5 0 3       |
| 0 0 0 1 0      | 1 0 0 0 2       |
| 0 1 1 1 0      |                 |

## Problem 7. 最後還是想要中序

(Time Limit: 1 seconds)

### 問題描述:

數學的運算式可以分成前序(Preorder)、中序(Inorder)、後序(Postorder)三種。現在請你寫一個程式可以將所有的運算式都轉換成中序。

1.若是輸入的式子本身就是中序運算式，則輸出英文字母 `same`。

2.所有的測試運算式的數字都只有個位數。

3.輸出的中序運算式中的每個運算都必須包含括號，不論括號有無意義都需輸出，除了最前頭以及最尾巴的括號除外，以便於在運算時可以很明確的判斷運算的順序。

### 輸入說明

第一行輸入一個正整數  $n$  ( $1 \leq n \leq 10$ )，表示有  $n$  個測試案例。

每組測試案例為一行不含空白的前序、中序、後序的運算式字串(字串長度  $\leq 100$ )。

### 輸出說明:

請轉換成中序運算式輸出，並請明確地將式子中各運算明確地做括號。

輸入為中序，則輸出 `same`。

### 範例

| Sample1 Input: | Sample1 Output:         |
|----------------|-------------------------|
| 5              | same                    |
| 3*2+1-2        | (5*6)-3                 |
| 56*3-          | (1+2)-3                 |
| -+123          | ((6+((2*9)/3))+(4*5))-8 |
| -++6/*293*458  | same                    |
| (1+6)-(3*5)    |                         |



