

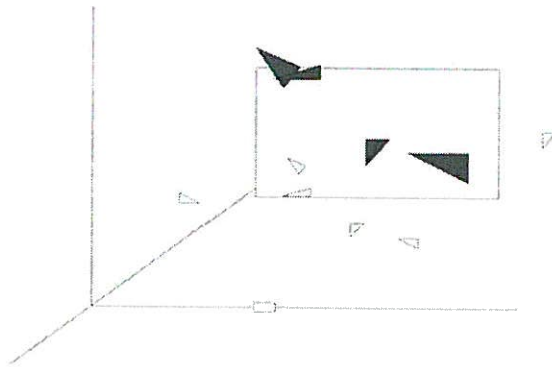
## Problem A

### 蝙蝠的影子

Input file: pa.txt

#### 問題敘述

在一個露天的電影院中，前方放置了長方形銀幕，在銀幕前方一定距離放了一部放映機。當時正值夏天，空地上許多蝙蝠在空中飛舞，飛舞的蝙蝠，經過燈光照射，在銀幕顯現出不同區塊的陰影，由於前後位置的關係，幾隻蝙蝠可能也只產生一塊陰影。給定長方形銀幕四個角落的三維座標、放映機位置的三維座標、分散空間各處蝙蝠（以三角形表示）的三角形頂點座標，請寫一個程式來計算這一群蝙蝠在銀幕上產生陰影區塊的個數。



#### 輸入格式

輸入檔包含數組測試資料。輸入檔的第一列為一整數，表示資料共有幾組，每組資料以只含有單一字元 ”/” 的列隔開，最後一組資料之後以含有單一字元 ”.” 的列作為結束。每一組測試資料的第一列包含四個三維座標  $(x, y, z)$ ，表示銀幕四個角落的三維座標。第二列則包含一個三維座標表示放映機的位置。第三列則有一正整數  $n$  代表蝙蝠個數。接下來的  $n$  列，每列有三個三維座標代表每隻蝙蝠的三角形頂點座標。

#### 輸出格式

針對每一組測試資料，輸出測試資料組數及陰影區塊個數。

輸入範例

```
1
20 0 15 20 20 15 20 20 0 20 0 0
0 10 0
6
10 6 7 10 7 8 10 6 9
5 8 3.5 5 9 3.5 5 9 4
6 16 2 6 17 2 6 17 4
22 0 4 22 1 5 22 2 4
6 12 2 6 11 2 6 12 1
8 10 4 8 10 3 8 9 2
.
```

輸出範例

```
Case 1: Shadow blocks = 3
```

## Problem B

### 資料探勘

Input file: pb.txt

#### 問題敘述

資料探勘是一個目前應用很廣泛的領域，其中一種應用是從文件（documents）中發掘出經常同時出現的關鍵字（keywords）。現在假設我們有  $n$  份文件以及  $m$  個關鍵字，我們可以用一個  $n \times m$  的整數矩陣  $A$  來表示每個關鍵字出現在每份文件中的次數。換句話說， $A(i,j)$  表示第  $j$  個關鍵字在第  $i$  份文件中出現的次數。如果第  $j$  個關鍵字與第  $k$  個關鍵字同時出現在某份文件中，則其在該份文件中「共同出現」（co-occurrence）的次數為其個別出現次數的最小值。例如假設第  $j$  個關鍵字出現 5 次，而第  $k$  個關鍵字出現 3 次，則其「共同出現」於該份文件的次數則定義為 3。兩個關鍵字的「相關性」可以定義為其在不同文件中共同出現的次數之總和。以數學式子來表示，第  $j$  個關鍵字與第  $k$  個關鍵字的相關性  $R(j,k)$  可以下列式子表示：

$$R(j,k) = \sum_{i=1}^n \min(A(i,j), A(i,k))$$

本題是針對給定的關鍵字出現在文件中的次數，找出相關性最高的一對關鍵字（兩個字當然要不一樣）並輸出其相關性的值。

#### 輸入格式

輸入檔包含數組測試資料。輸入檔的第一列為一整數，表示資料共有幾組，每組資料以只含有單一字元 “/” 的列隔開，最後一組資料之後以含有單一字元 “.” 的列作為結束。每組資料第一列為兩個正整數  $n$  以及  $m$ ，其中第一個整數  $n$  代表文件的數目，而第二個整數  $m$  代表關鍵字的數目。皆下來是  $n$  列非負整數，每一列有  $m$  個非負整數，其中第  $i$  列的第  $j$  個整數代表第  $j$  個關鍵字在第  $i$  份文件中出現的次數。

#### 輸出格式

每一列輸出資料為一整數，代表在每一組測試資料中，相關性最高但不相同的一對關鍵字，其相關性的值。

輸入範例

2  
5 3  
2 1 4  
2 3 2  
4 3 1  
5 4 4  
1 2 2  
/  
3 4  
0 3 4 6  
3 7 3 9  
4 8 2 1  
.

輸出範例

11  
11

第一組測試資料分析

關鍵字 1 與 2 之相關性為  $\min(2,1)+\min(2,3)+\min(4,3)+\min(5,4)+\min(1,2)=11$

關鍵字 1 與 3 之相關性為  $\min(2,4)+\min(2,2)+\min(4,1)+\min(5,4)+\min(1,2)=10$

關鍵字 2 與 3 之相關性為  $\min(1,4)+\min(3,2)+\min(3,1)+\min(4,4)+\min(2,2)=10$

故最大值為 **11**

## Problem C

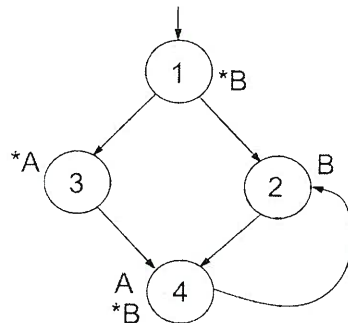
### 特殊旅遊卷

Input file: pc.txt

#### 題目敘述

有一家飛龍旅行社要企畫一個旅遊計畫。該計畫以某個城市為起點旅行到某個終點城市。在起點與終點之間有許多的城市。現在飛龍旅行社透過關係拿到了一批來歷不明的旅遊卷。旅遊卷的種類從 A 到 Z (最多 26 種)。每一種旅遊卷可以使用在某些城市的景點或飯店。不過，這一批旅遊卷有一些奇怪的地方。在使用前必須先蓋印屬於該旅遊卷特有的有效章才能使用。很不幸的，飛龍旅行社並沒有這些章可以事先蓋印在所有旅遊卷上。

經過飛龍旅行社查訪的結果，在某些城市裡，有合夥的旅行社可以透過管道替某些種類的旅遊卷蓋有效章。假設，從起點到終點，飛龍旅行社設計了許多條熱門的路徑，好讓客戶可以使用這批旅遊卷來自由。如圖一，從城市 1 到城市 4，可以經過城市 2 或城市 3。他們提供了這許多條路徑讓客戶利用這批旅遊卷



圖一

來進行自由。在圖一，假設在某個城市可以為某旅遊卷 X 來蓋章，我們在城市的旁邊標示 \*X。若旅遊卷 X 可以在該城市中使用，我們在城市的旁邊標示 X。如圖一，城市 3 可以為旅遊卷 A 來蓋有效章。在城市 4 當中，可以使用旅遊卷 A，也可以為旅遊卷 B 來蓋章。

現在，飛龍旅行社想知道的是，如果他們設計了上述的自由行路徑圖，是否每張旅遊券在使用前都能被蓋好有效章。以圖一為例，旅遊卷 B 沒有問題，因為若客戶走 1-2-4 的路徑，還是 1-3-4-2-4 的路徑，或任何路徑，旅遊卷 B 都可以在使用前，蓋好有效章。旅遊卷 A 就有問題。因為客戶若選擇走 1-3-4，那麼使用旅遊卷 A 沒有問題。可是若客戶選擇走 1-2-4，他會發現在城市 4 當中，他不能使用旅遊卷 A，因為沿路上不能蓋有效章。

請寫一程式，輸入為上述的路徑企畫，然後回答有哪些旅遊卷是有問題的。

#### 注意

每種旅遊卷有可能可以在一個以上的城市中使用。你不需要考慮旅遊卷是否在某

城市中用掉與否的問題。你只需要考慮該旅遊卷若要在某城市中使用時，是否能在到達該城市的所有路徑上蓋到有效章。

#### 輸入格式

輸入檔包含數組測試資料。輸入檔的第一列為一整數，表示資料共有幾組，每組資料以只含有單一字元 ”/” 的列隔開，最後一組資料之後以含有單一字元 ”.” 的列作為結束。每組資料第一列為一個正整數  $m \leq 100$ ，代表城市的個數且城市的編號從 1 到  $m$ 。下一列有兩個整數，第一個整數是起點城市的編號，第二個整數是終點城市的編號。接下來有  $m$  列，每一列則是每個城市的旅行卷蓋章或使用的狀況。這部分的資料先以城市編號起頭，然後是相關旅遊卷的資訊。例如 2 A \*B @ 代表在城市 2 可以使用旅遊卷 A，也可以為旅遊卷 B 蓋章。中間以空白隔開。讀到小老鼠符號(@)時，則表示該城市的資料結束。在此  $m$  列資料之後的第一列有一個整數  $e$ ，代表城市與城市連結的個數。緊接著是  $e$  列的城市與城市連結資料，每一筆資料包含出發地與目的地。

#### 輸出格式

針對每一組測試資料，請於一行之內輸出所有有問題的旅遊卷種類。若多於一種，則以空白隔開。

#### 輸入範例

```
1
4
1 4
1 *B @
2 B @
3 *A @
4 A *B @
5
1 2
1 3
2 4
3 4
4 2
.
```

#### 輸出範例

```
A
```

## Problem D

### 換字母遊戲

*Input file: pd.txt*

#### 問題敘述

我們在教孩童學習英語單字時，常常使用換字母的方式，使孩童能夠一個單字接著一個單字記憶下來，而有舉一反三之效果。舉例而言，下列四個單字 buy、but、out、our 相鄰之間只有一個位置的字母不同，老師就可逐字教孩童認字，學習起來，也格外有趣了。

爲了此目標，我們已在文字檔 diction.txt 中，存有一給孩童用的英文字典，依字典次序排列，共存有 994 個常用的英文單字，如下所示：

```
a
ability
able
:
your
yourself
zoo
```

本題是要你撰寫一程式，開始時輸入兩個長度一樣的字串 A、B，接著請你到此特定字典中尋找換字母遊戲之最短解答的總單字數量即可。舉例而言，如果 A='buy'，而 B='our'，那麼我們可在此字典中找到另外兩個單字 but、out，而使得依序下來這四個單字 buy、but、out、our 相鄰兩字之間只有一個位置的字母不同。以此例而言，這是一組最短的解答，也就是總單字數量(=4)最少的解答。

#### 輸入格式

共有五組測試資料，每組有一行，輸入以空白隔開、長度一樣的二個不同的字串，此處字串的長度介於 1 到 14 之間。

#### 輸出格式

共有五行輸出總單字數量的答案，分別對應五組測試資料的答案。如果某組測試資料輸入的兩個字串中有一個字串不在字典 diction.txt 中，那麼你的程式必須在所對應的那一行輸出"-1"。如果某組測試資料輸入的兩個字串無法使用其它字串換字母成功，而且這兩個字串彼此之間超過一個以上位置的字母不同，那麼你的程式必須在所對應的那一行輸出"-1"。如果某組測試資料輸入的兩個字串無法使用其它字串換字母成功，但這兩個字串彼此之間只有一個位置的字母不同，那麼你的程式必須在所對應的那一行輸出"2"。如果非上述之情形，則輸出最短解答的總單字數量。

輸入範例

buy our  
zoo can  
possible possibly  
a i  
programming probability

輸出範例

4  
-1  
2  
2  
-1



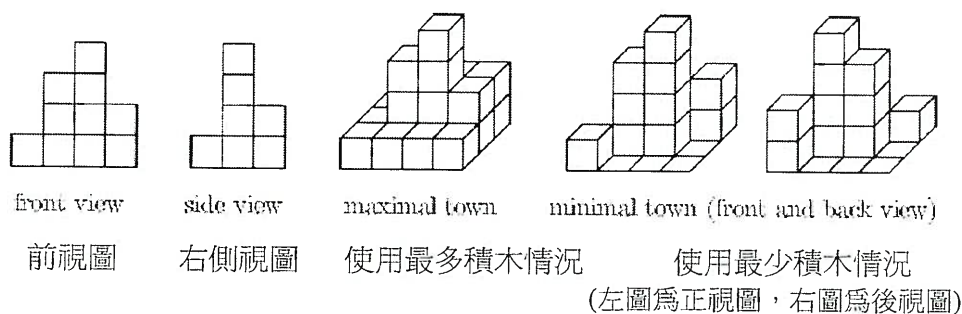
## Problem E

### 積木問題

Input file: pe.txt

#### 問題說明

樂高建築公司是一個有名的建築公司，許多建築師都已能夠進入該公司工作為榮。如今樂高建築公司正在招考新建築師，其中一項考題即是要測試建築師們的3D空間建築規劃能力。其考試方法如下：給定一張長方形的桌子（寬為  $K$  單位，長為  $L$  單位）及一個社區建築物的前視圖和右側視圖，每位建築師必須在有限的時間內用積木（積木的長、寬、高各為一單位長）蓋出符合該圖的最小社區（即利用最少積木）和最大社區（即利用最多積木）。以下便是一個前視圖和右側視圖的範例：



請你寫一個程式，根據樂高建築公司所給的前視圖和右側視圖，計算出蓋出最小社區及最大社區所需的積木個數。

#### 輸入檔案格式

輸入資料檔的第一行有一個整數  $n$  代表共有  $n$  組測試資料。每一組測試資料的第一行包含了兩個正整數  $K, L$  (分別代表桌子的寬和長)， $K$  和  $L$  均不超過 5,000。接下來的  $K$  行中，每行包含了一個整數，分別代表前視圖中，由左到右的積木堆疊高度(堆疊的高度相當於積木的個數，即積木的體積為  $1 \times 1 \times 1$ )。同樣的，接下來的  $L$  行中，每行也包含了一個整數，分別代表右側視圖中由左到右的積木堆疊高度。任一堆疊積木的高度將不會超過 5,000，而且所需的積木總數不會超過 1,000,000,000。

#### 輸出格式

每一組測試資料的輸出結果只有一行：測試資料組數及兩個數字，分別代表最小社區及最大社區所需的積木個數，中間以一個空白隔開，若是不可能建造出符合前視圖和右側視圖的社區，則列印出 "No solution"。

### 輸入檔案範例

2 代表共有兩組測試資料  
4 3 第一組測試資料：桌子的寬為 4 單位，長為 3 單位  
1 前試圖  
3 前試圖  
4 前試圖  
2 前試圖  
1 右側視圖  
4 右側視圖  
2 右側視圖  
2 2 第二組測試資料：桌子的寬為 2 單位，長為 2 單位  
4 前試圖  
1 前試圖  
1 右側視圖  
3 右側視圖

### 輸出範例

Case 1: 10 21  
Case 2: No solution

## Problem F

### 求最大值

*Input file: pf.txt*

#### 問題敘述

假設有一軟體公司接了若干個計畫，每個計畫皆有結案的日期及所需的工作天數，另外每一個計畫如果能在期限內完工結案，有關廠商將提供額外獎金。為簡化問題，假設結案日期為一正整數且每一計畫一但開始執行就必須連續執行直到結案為止，中途也不能執行其他計畫。試寫一程式排出最佳的執行計畫順序，使能獲得最多的獎金。注意，未必所有的計畫均能如期的結案而獲得獎金。

#### 輸入格式

輸入檔中有三組測試資料，每一組有一列輸入，輸入格式為：

$$n \ d_1 \ t_1 \ b_1 \ d_2 \ t_2 \ b_2 \ \dots \ d_n \ t_n \ b_n$$

其中  $n < 100$  表示計畫的個數，對於任意  $i$ ， $d_i$  為一整數用來表示第  $i$  個計畫的結案日期， $t_i$  表示第  $i$  個計畫所需的天數， $b_i$  表示第  $i$  個計畫如期或提前結案的獎金。

#### 輸出格式

每組測試資料中有可能獲得的最高獎金金額。

#### 輸入範例

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

#### 輸出範例

```
5
3
4
```