

106 計算機概論練習題

- 一、本練習題集共 12 題，若有修習 106 下計算機概論與實習(三)的同學，請在寒假期間**至少完成 5 題**，開學後須繳交，會登錄為一次作業成績。
- 二、5 題可從練習題集 12 題中**任選**。若未完成 5 題，則視完成題數，每題 20 分計算，如：完成 2 題 40 分、4 題 80 分，依此類推，完成 5 題則為 100。
- 三、完成超過 5 題則進行加分，目前尚未進行計概(三)的配分，會在下學期開學時公告加分機制。
- 四、題目難易度並未依順序排序。
- 五、輸出及輸入格式請依照題目規定，若差異過大則視為錯誤。
- 六、請利用 C 語言撰寫程式。
- 七、若有任何問題，請寄信向助教詢問：
楊易哲：d0308897@mail.fcu.edu.tw
杜宜霖：yilin585198@gmail.com
簡瑞梓：d0440817@fcu.edu.tw

***** 請勿抄襲他人程式碼 *****

可以互相討論作法，但請不要抄襲

不了解程式碼而照抄永遠不會進步

若真心關愛您的同學，請不要直接給予程式碼供其抄襲

一、 身份證字號檢驗(字元處理)

我國的身分證字號有底下這樣的規則，因此對於任意輸入的身分證字號可以有一些基本的判斷原則，請您來判斷一個身分證字號是否是正常的號碼(不代表確有此號、此人)。

(1) 英文代號以下表轉換成數字

A=10 台北市	J=18 新竹縣	S=26 高雄縣
B=11 台中市	K=19 苗栗縣	T=27 屏東縣
C=12 基隆市	L=20 台中縣	U=28 花蓮縣
D=13 台南市	M=21 南投縣	V=29 台東縣
E=14 高雄市	N=22 彰化縣	W=32 金門縣
F=15 台北縣	O=35 新竹市	X=30 澎湖縣
G=16 宜蘭縣	P=23 雲林縣	Y=31 陽明山
H=17 桃園縣	Q=24 嘉義縣	Z=33 連江縣
I=34 嘉義市	R=25 台南縣	

(2) 英文轉成的數字，個位數乘 9 再加上十位數的數字

(3) 各數字從右到左依次乘 1、2、3、4 8

(4) 求出(2),(3) 及最後一碼的和

(5) (4)之解 除 10 若整除，則為 real，否則為 fake

例： T112663836

$$2 + 7*9 + 1*8 + 1*7 + 2*6 + 6*5 + 6*4 + 3*3 + 8*2 + 3*1 + 6 = 180$$

除以 10 整除，因此為 real

輸入說明

本題為重複輸入，有多筆測資。

先輸入一個 N，代表有 N 筆測資。接下來輸入 N 組身分證號碼。

輸出說明

針對每一組身分證號碼檢測並輸出 real 或 fake。

每筆結果印出後加入換行符(‘\n’)。

input

2

T112663836

S154287863

output

real

fake

二、 求數列的第 n 項(Array)

有一個數列，第一項的值為 1，第二項的值為第一項加 1，第三項的值為第二項加 2，第四項的值為第三項加 3 ... 第 k 項的值為第 k-1 項的值加上 k-1。

給一個數字 n，請印出這個數列的第 n 項。

輸入說明

本題為重複輸入，有多筆測資。

先輸入一個 X，代表有 X 筆測資。

接下來輸入 X 筆資料，每筆輸入含有一個 n， $1 \leq n \leq 500$ 。

輸出說明

針對每一組測資印出數列的第 n 項。

每筆結果印出後加入換行符('\n')。

input

2

1

4

output

1

7

三、 伐木工的鬍子(順序判斷)

有一個無聊工頭與一群伐木工，工頭很喜歡找伐木工們的麻煩，他會要求伐木工們以十個一組按照他們的鬍子長短依序排成一列。

你請寫一個程式判斷伐木工是否以由長到短，或以由短到長的順序排成一列。
不會有人的鬍子一樣長。

輸入說明

第一列有一個整數 N 表示測試資料的組數，接下來有 N 列，每列有 10 個**相異**的正整數表示每人鬍子的長短。

輸出說明

請以 "Ordered" 表示有照順序排列，以 "Unordered" 表示沒有照順序排列，第一列請輸出 "Lumberjacks:"，每列最後加入換行符('\n')。

input

```
3
13 25 39 40 55 62 68 77 88 95
88 62 77 20 40 10 99 56 45 36
91 78 61 59 54 49 43 33 26 18
```

output

```
Lumberjacks:
Ordered
Unordered
Ordered
```

四、 賽道編排(Array, Sorting)

徑賽短中距離賽跑，皆會依照選手實力進行最佳最公平的分組及賽道，當參賽者眾多的時候，將設置預賽或複賽，分組及賽道編排時必須依循兩個原則：

1. 實力優秀者，儘量排在中間的賽道。
2. 實力相當者，絕對避免排在同一組。

分組及賽道編排的方法如下：

1. 常態分組：方法是使用 S 型的分配，比如說選手共有 31 人，每 8 人一組(因為每組要 8 個賽道)。步驟流程可以依照下列的方法實施。

*依照個人最佳成績排序。

*總人數/8 = 總組數。如 $31 / 8 = 4$ ，必須分四組。

*如分 4 組，每個人的分組依成績排序，設定為 123443211234.....，依照這個規則填滿為止。

2. 賽道編排：在每個分組上面，成績越好的，越往中間集中。方法就是分組排序號，以 45362718 的順序排賽道。

輸入說明

1. 第一個數字 N 代表共有幾位選手， $N \leq 200$ ，且 N 必為 8 的倍數。
2. 每列有兩項資料，用空白隔開 1 10.80，1 代表選手，10.80(秒)是他的最佳成績。假設每一位選手的最佳成績不重複。

輸出說明

每列共有資料兩個部分，第一個資料代表分組，後面 8 筆資料代表選手所排的賽道順序。如：1 15 10 2 6 11 5 13 16，最前面的 1 代表第一組，15 10 2 6 11 5 13 16 都是選手編號，順序就是賽道的順序，最左邊為第 1 道，最右邊為第 8 道。

input

16
1 10.80
2 10.35
3 10.02
4 10.44
5 11.32
6 09.93
7 11.52
8 11.53
9 12.34
10 11.42
11 10.32
12 10.28
13 12.21
14 12.54
15 12.26
16 13.40

output

1 15 10 2 6 11 5 13 16
2 9 7 4 3 12 1 8 14

五、 MOD3

相信對同學們來說，判斷一個數除以三的餘數是多少是沒問題的！
現在，給你很多數字，請你輸出 $3k$ 、 $3k+1$ 、 $3k+2$ 的數字各有幾個。
**** Hint：將數字進行逆推並各自加總 ****

輸入說明

第一行有一個正整數 n ，代表接下來有幾個數字， $0 < n \leq 10,000$ ，
接著有 n 個介於 1 到 50,000 之間的數字，請你做判斷。

輸出說明

輸出三個數字(以空白隔開)，
分別為在這 n 個數字中，三的倍數、三的倍數+1、三的倍數+2 的數量。
最後加入換行符('\n')。

input

5
1
2
3
4
5

output

1 2 2

六、 猜數字遊戲

在這個遊戲中，你要在心理想一個四位數數字讓對手去猜，每一位數都**不可以重覆**，例如 2343 裡有兩個 3，就是個不合法的數字。每次對手猜測你的數字時，你要根據對手的猜測給予提示。如果對手的四位數中有若干位數和你的答案相同，**而且在正確的位置上**，我們稱這些位數為 A。例如你的答案為 1536，對方猜 1234，那麼 1 和 3 就是 A。但是如果**數字相同，位置不同**，我們稱這些位數為 B。例如你的答案為 1536，對手猜 2345，那麼 3 和 5 就是 B。當然對手的猜測很有可能同時有 A 也有 B，例如你的答案是 1536，對手的猜測為 3456，那麼你應該給對手的提示就是 1A2B。如果所得到的提示是 4A0B，就代表完全答對！

輸入說明

輸入的第一行有一個四位數的整數，代表你心中所想的數字。
接下來每次有一個皆不重複的四位數整數，代表對方的猜測。

輸出說明

對於對手的每一次猜測，請輸出 xAxB 的提示於一行。
並在猜中時(4A0B)結束。每列最後加入換行符('\n')。

input

1536
1234
2345
3456
2480
1536

output

2A0B
0A2B
1A2B
0A0B
4A0B

七、 Cryptanalysis

密碼翻譯 (cryptanalysis) 是指把某個人寫的密文 (cryptographic writing) 加以分解。這個程序通常會對密文訊息做統計分析。你的任務就是寫一個程式來對密文作簡單的分析。

輸入說明

輸入的第 1 列有一個正整數 n ，代表以下有多少列需要作分析的密文。
接下來的 n 列，每列含有 0 或多個字元 (可能包含空白字元)。

輸出說明

每列包含一個大寫字元 (A~Z) 和一個正整數。表該字母出現次數，中間使用空白分隔，一列輸出後加入換行符 ('\\n')。

*** 如果某一字元未出現在輸入中，那他也不應出現在輸出中 ***

input

3

This is a test.

Count me 1 2 3 4 5.

Wow!!!! Is this question easy?

output

S 7

T 6

I 5

E 4

O 3

A 2

H 2

N 2

U 2

W 2

C 1

M 1

Q 1

Y 1

八、 完全平方和

給你一個範圍 a 到 b ，請你找出 a 與 b 之間所有完全平方數的和。

例如：範圍 $[3, 25]$ 中所有完全平方數的和就是 $4 + 9 + 16 + 25 = 54$ 。

**** 1 也是完全平方數 ****

Hint：利用 `math.h` 的 `sqrt()` 判斷是否為完全平方數，或是直接找出小於等於 1000 的所有完全平方數

輸入說明

輸入的第一列有一個整數 T ，代表以下有多少組測試資料。

每組測試資料為兩列，包含兩個數 a 與 b ($0 \leq a \leq b \leq 1000$)。

輸出說明

每組測試資料輸出一列，內容為 a 及 b 間所有完全平方數的和。

每列最後加入換行符('\n')。

input

2

1 5

5 35

output

5

50

九、駱駝的交易

在西元 800 年左右，巴格達的一個商人 El Mamum 手上有一個公式：

$1+2*3*4+5$ ，這個公式是用來做為駱駝交易的。

由於這個公式沒有括號來決定運算的先後順序，所以答案可能有很多種。所以，他要求僕人提供他一個方法來解析這個公式，好讓他可以在買賣駱駝時獲益最大。

你受 El Mamum 的委託寫一個程式找出一個不含括號的公式可能的解析結果中最大值及最小值分別是多少。

Hint : 你可能會使用到 `long long int`，印出格式為 `%lld`。（“l”為英文字母小寫）

輸入說明

輸入的第一列有一個正整數，代表以下有多少組測試資料。

每組測試資料最多有 12 個數字，每個數字均介於 1 到 20 之間。這些數字中以加號（+）或乘號（*）來分隔。

請參考 Sample Input。

輸出說明

對每組測試資料輸出一列，公式可能的解析結果中最大值及最小值。

輸出格式請參考 Sample Output。

input

5

$1+2*3*4+5$

$4*18+14+7*10$

$3+11+4*1*13*12*8+3*3+8$

$20*20*20*20*20*20*20*20*20*20*20*20$

$1+2+3+4+5$

output

The maximum and minimum are 81 and 30.

The maximum and minimum are 1560 and 156.

The maximum and minimum are 339768 and 5023.

The maximum and minimum are 4096000000000000 and 4096000000000000.

The maximum and minimum are 15 and 15.

十、 Self Numbers

在 1949 年印度數學家 D.R Kaprekar 發現了一種數字：Self-numbers。

對任何正整數 n ，定義 $d(n)$ 為 n 加上其各數字的和。例如：

$d(75)=75+7+5=87$ 。給任一個正整數 n 當作一個起始點，你可以產生無限的數字序列： $n, d(n), d(d(n)), d(d(d(n))), \dots$ 例如：如果你從 33 開始，下一個數字是 $33+3+3=39$ ，再下一個數字是 $39+3+9=51$ ，再下一個數字是 $51+5+1=57$ 。所以你可以產生以下的序列：

33, 39, 51, 57, 69, 84, 96, 111, 114, 120, 123, 129, 141,

我們稱 n 為 $d(n)$ 的 generator。在上面的例子中 33 是 39 的 generator，39 是 51 的 generator，51 是 57 的 generator，以下類推。有些數有不只一個 generator，例如：101 有 2 個 generators，91 和 100。如果一個數沒有 generator，那他就是一个 self-number。比 100 小的 self-number：1, 3, 5, 7, 9, 20, 31, 42, 53, 64, 75, 86, 97

本問題是：找出所有小於或等於 10,000 的 self-numbers。

輸入說明

沒有輸入。

輸出說明

輸出小於或等於 10,000 的 self-numbers。

每列最後加入換行符('\n')。

input

沒有輸入

output

1

3

5

7

9

20

31

42

53

64

|

|

<-- (許多 self-numbers)

|

9903

9914

9925

9927

9938

9949

9960

9971

9982

9993

|

|

|

十一、 Fibonacci

Fibonacci 數列(0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55,...)的定義是：

$$F_0 = 0$$

$$F_1 = 1$$

$$F_i = F_{i-1} + F_{i-2} \quad \text{for all } i \geq 2$$

請寫一支程式計算某一項 Fibonacci 數。

*** 請注意程式執行時間 ***

輸入說明

每組測試資料一列，各有一個整數 n ($0 \leq n \leq 5000$)，代表要求的第 n 項 Fibonacci 數。

*** 此題為重複輸入測資，不須判斷何時結束 ***

輸出說明

對每組測試資料請輸出第 n 項 Fibonacci 數，請參考 Sample Output。

每列最後加入換行符('\n')。

input

5

7

11

200

output

The Fibonacci number for 5 is 5

The Fibonacci number for 7 is 13

The Fibonacci number for 11 is 89

The Fibonacci number for 200 is 280571172992510140037611932413038677189525

十二、 Cipher

Bob 和 Alice 開始使用一種全新的編碼策略。他們採用秘密鍵 (secret keys) 的方式來編碼及解碼。在 1996 年 2 月 16 日在費城他們上次見面的時候，他們選定了他們的秘密鍵。這些秘密鍵是由 1 到 n 的整數所構成，但是排列的順序卻是他們任意挑選的。在編碼的時候採用以下的原則：

把要編碼的訊息 (明文) 寫在秘密鍵下面，每個字元與秘密鍵的一數字對齊。位於位置 i 的字元經編碼後其位置為 a_i ， a_i 為秘密鍵中第 i 個位置的值。明文中的每個字元編碼後就得到密文了。這密文還可以用同樣的策略再加密，經過了 k 次加密後他們交換他們的密文。

明文的長度一定小於等於 n 。如果明文的長度小於 n ，就在其後方加上空白字元使得其長度剛好為 n 。你的任務是幫助 Bob 和 Alice 寫一個程式讀入秘密鍵，以及 k ，以及一連串要加密的明文，然後輸出密文。

輸入說明

輸入含有多組測試資料，每組測試資料的第一列有一個整數 n ($0 < n \leq 200$)，代表秘密鍵的長度。

接下來的一列為秘密鍵，含有 n 個整數，內容為 1 到 n 的某種排列。再接下來的每列為 k 與明文，其中以一空白字元分隔。

請注意：每列以換行符 (' $\backslash n$ ') 當作結束，但是換行符不列入明文。當遇到 $k=0$ 時代表此組測試資料結束。

另外， k 可能相當大 (但是不會超出 C 語言中 `int` 的範圍)，所以請注意你程式的演算法，否則執行時間過長則視為錯誤。

當遇到 $n=0$ 時代表整個輸入結束。請參考 Sample Input。

輸出說明

對每一組測試資料中的每列明文，輸出加密後的密文 (長度一定為 n)。每組測試資料後請空一行。請參考 Sample Output。

input

10

4 5 3 7 2 8 1 6 10 9

1 Hello Bob

2 Hello Bob

1995 CERC

0

5

1 2 3 5 4

1 Hello

0

0

output

BolHeol b

lelBo Hob

C RCE

Helol