

【101 學年度 高一程式設計入門 寒假作業】

一、心理測驗一：生命顏色

(1) 這生命顏色是法國著名心理及靈術家 Ganaesia 所研究出來的 ...

只要把出生年月日相加，便可找出自己的生命顏色，從而窺探內心深處真正的自己。

生命顏色的計算方法：

把你的西元出生 年 月 日 用的數字寫出來，然後將數字一個一個相加，直到結果變成單一數字為止。這個數字就是你的「生命數字」。

「生命數字」所對應的「生命顏色」解析如下：

- | | |
|-------------------|-------------------|
| 1 = 紅色：熱情衝動的莽小子 | 6 = 黃色：樂觀開朗的傻小子 |
| 2、4 = 藍色：理性助人的小天使 | 7 = 橙色：依賴成性的小BABY |
| 3 = 綠色：大而化之的牆頭草 | 8 = 黑色：人格分裂的小惡魔 |
| 5 = 白色：單純天真的幻想家 | 9 = 紫色：深謀遠慮的思想家 |

EX：1978 年 8 月 23 日出生的人

$1 + 9 + 7 + 8 + 8 + 2 + 3 = 38$ $3+8=11$ $1+1=2$ => 藍色

(2) 程式螢幕輸出畫面均需參考下面範例輸出結果：

【Hint】：輸出輸入、算術、if...else、switch...case

<p>【範例輸入】： 請輸入西元出生年： 2000 請輸入出生月： 3 請輸入出生日： 15</p> <p>【範例輸出】： 你的生命顏色是藍色 你是個理性助人的小天使</p>	<p>【範例輸入】： 請輸入西元出生年： 2001 請輸入出生月： 11 請輸入出生日： 25</p> <p>【範例輸出】： 你的生命顏色是綠色 你是個 大而化之的牆頭草</p>
---	---

二、心理測驗二：從生日看前世

對生活在世上的每一個人來說，經常會和周遭原本的陌生人因為工作、情感或是種種原因產生關聯。其實，透過一個小測驗，就可以明白你和對方在前世與今生，究竟是什麼關係。

【計算方法】

分別將出生年月日拆開算至個位數為止。最後再將各別的誕生數合計，同樣算至個位數為止。就可以找到彼此的關係數字了。

【範例】

你的生日是：1973年3月25日
 $= 1 + 9 + 7 + 3 + 3 + 2 + 5 = 30$
 $\Rightarrow 3 + 0 = 3 \Rightarrow$ 誕生數

他（她）的生日是：1970年8月10日
 $= 1 + 9 + 7 + 0 + 8 + 1 + 0 = 26$
 $\Rightarrow 2 + 6 = 8 \Rightarrow$ 誕生數

兩人的關係數字： $3 + 8 = 11 \Rightarrow 1 + 1 = 2$

【程式回傳值詳細資料】

關係數字 1 / 你們得關係是主從關係，反映於實際方面可能是師父徒弟、或是上司下屬等模式不過，由那一位主導，則還不一定。在前世中，如果你曾經對他人百般要求，則現今這一世，也必須要忍耐接受別人的的要求。假使你們前世的關係很和善，則這一世可以相互共生，發揮很好的關係。

關係數字 2 / 擁有這個關係的雙方，發展成戀人的關係機率很大。在前世關係中，你們可能在情感上有很強的關聯性，屬於親兄弟、戀人機率高，透過言語表達時，你們有很深的默契，能深入溝通。而這種關係，還可能反應在信仰、或藝術層面。相對在這一世，你們在精神方面的相通性，也會超出人們預料。

關係數字 3 / 你們會成為好朋友，彼此能夠補強對方的缺點，而朝越來越緊密的友情之路前進。依這種情形看，在前世你們的關係也是一樣的，不僅在個性、興趣方面相近，當初見面時的情況，就像是早就見過面似的。可以開懷暢談，當你需要幫助時，對方往往扮演最好的協助者角色。

關係數字 4 / 屬於同事關係，平淡無味但可以為共同目標而奮鬥。在前世的時候，你很能夠犧牲自我為別人做一些事，而在有必要共同完成一件事時，盡責盡力去完成。你們兩個人不僅懂得忍耐的哲學，更會成為最佳拍檔。

關係數字 5 / 是人生旅途中的伴旅，雖然不會很深入，但彼此很知心。在前世中，你們二人也是一樣互相幫助的旅人。所以，現在當一方陷入危機時， 另一方就會適時出現，並予以協助，兩人關係協調。

關係數字 6 / 像家人一般，正是你們的感覺。這意味著你們可以在一種平穩 且彼此相互依賴的關係中並存。依這種關係推論，過去的你們不外乎是親戚、情人、或家人的關係。而透過自然和藝術的層面，你們更擁有良好的溝通交流。這一世，還是有傾向往自然和藝術方面聯結。

關係數字 7 / 你們有著深遠的精神相通層面，這說明了，你們的關係在求道 與宗教方面，會有良好的依持行爲出現。在前世上，不論是宗教、哲學等範疇，都是你們探討的問題，就算別人覺得你們孤立不太能溝通，但你們卻可以在相互理解中得到慰藉。

關係數字 8 / 你們是很好的競爭對手。在前世關係中，你們會因為某些事情 而一競高下，最後，卻在相互理解中釋懷。到了這一世，這種競爭的關係， 反而讓你們感強發展得更好，不管有多少競爭場面出現，最後都可以在協調 中得到解決。

關係數字 9 / 你們屬於相互奉獻的代表，在情感方面可以相互理解。前世中， 你們最初是因為一些煩惱或糾葛，而彼此溝通不良。最後，卻可以克服困難， 而讓雙方在友情上有所協助，並建構出一種協調的關係。

三、編碼問題

假設有一種 8 個 bits 的編碼規則，常被用來傳遞資料，這種編碼規則被設計為可以自動校正，如果傳輸的過程產生資料錯誤的話，這種編碼方式可以自己校正出正確的碼。它的編碼方式如下：

此一 8 個位元的碼，由左至右的每一個位元，我們稱之為 b_1 、 b_2 、 b_3 、 b_4 、 b_5 、 b_6 、 b_7 、 b_8 。其中， b_1 、 b_2 、 b_4 、 b_8 為 parity bits。其中， b_3 、 b_5 、 b_6 、 b_7 為 data bits。 b_1 的值是由 b_1 、 b_3 、 b_5 、 b_7 經過 even parity check 運算而來，也就是說， b_1 、 b_3 、 b_5 、 b_7 這四個 bit 裏面，為 1 的 bit 數要為偶數。 b_2 的值，則由 b_2 、 b_3 、 b_6 、 b_7 經過 even parity check 運算而來，也就是說， b_2 、 b_3 、 b_6 、 b_7 這四個 bit 裏面，為 1 的 bit 數要為偶數。 b_4 的值，由 b_4 、 b_5 、 b_6 、 b_7 經過 even parity check 運算而來，也就是說， b_4 、 b_5 、 b_6 、 b_7 這四個 bit 裏面，為 1 的 bit 數要為偶數。 b_8 的值恆為 0。它的解碼方式如下：當接收到一個 8 bits 的碼，檢驗所有的 parity bit，如果 parity bit 有誤，則記錄下發生錯誤的 parity bit 位置，所有錯誤 parity bit 位置的和，即為發生錯誤的 data bit 位置。

例如：錯誤發生在 b_2 和 b_4 的話，那麼 $2+4=6$ ，即代表 b_6 的值是錯的，需要更正。請寫一程式，依上述內容編碼，並輸出至螢幕。

如：

```
【輸入】 0001 【輸出】 11010010
【輸入】 1010 【輸出】 10110100
【輸入】 0010 【輸出】 01010100
【輸入】 1011 【輸出】 01100110
```

四、解碼問題（同上）

寫一程式，依上述內容解碼，並輸出至螢幕。

如：

```
【輸入】 10001110 【輸出】 0011
【輸入】 00110110 【輸出】 1001
【輸入】 11001100 【輸出】 0110
【輸入】 10000000 【輸出】 0000
```

五、($3n + 1$) 的問題：

遊戲規則如下：

1. 輸入 n
2. 印出 n
3. 如果 $n = 1$ 結束
4. 如果 n 是奇數 那麼 $n=3*n+1$
5. 否則 $n=n/2$
6. GOTO 2

例如輸入 22, 得到的數列：

22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

據推測此演算法對任何整數而言會終止（當列印出 1 的時候）。雖然此演算法很簡單，但以上的推測是否真實卻無法知道。然而對所有的 n ($0 < n < 1,000,000$) 來說，以上的推測已經被驗證是正確的。

給一個輸入 n ，透過以上的演算法我們可以得到一個數列（1 作為結尾）。此數列的長度稱為 n 的 `cycle-length`。上面提到的例子，22 的 `cycle length` 為 16。

問題來了：對任 2 個整數 i ， j 我們要知道介於 i ， j （包含 i ， j ）之間的數所產生的數列中最大的 `cycle length` 是多少。（ $i > 0$ ， $j < 1000$ ）

如：

【輸入】1 10	【輸出】20
【輸入】100 200	【輸出】125
【輸入】201 210	【輸出】89

六、ClockHands

在一般的時鐘上通常有兩根指針：時針、分針。這個題目是告訴你幾點幾分，請你的程式回應時針和分針之間的角度。請注意：所有的角度請回應最小的正角度。例如：9:00 是 90 度，不是 -90 度，也不是 270 度。

【Input】

輸入是一連串的時間，每個時間一行，以下列的格式出現： H:M 。

1<= H <= 12 00<= M <=59

如果時間是 0:00 代表輸入結束了。請注意：H 可能是 1 或 2 位數，M 則總是以 2 位數出現 . 。 (也就是你在一般電子錶上看到的模式。)

【Output】

請輸出對應輸入時間的時針和分針間的最小正角度。此角度應該介於 0 到 180 度。每個輸出單獨一行。每個輸出至小數點以下第三位。

【Sample Input】

12:00
9:00
8:10
0:00

【Sample Output】

0.000
90.000
175.000