

程式設計 (105-2)

作業一

作業設計：孔令傑
國立臺灣大學資訊管理學系

繳交作業時，請至 PDOGS (<http://pdogs.ntu.im/judge/>) 為第一、二題上傳一個 PDF 檔，再為第三題與第四題各上傳一份 C++ 原始碼 (以複製貼上原始碼的方式上傳)。第四題是 bonus 加分題。每位學生都要上傳自己寫的解答。不接受紙本繳交；不接受遲交。請以英文或中文作答。

這份作業的截止時間是 **2017 年 2 月 27 日凌晨一點**。在你開始前，請閱讀課本的第 1.7-1.9、1.14、2.1-2.8、3.1-3.5 和 3.7-3.9 節¹。為這份作業設計測試資料並且提供解答的助教是林敬傑 (Jack Lin)。

第一題

(25 分) 敬傑想要寫一個 C++ 程式，讓使用者輸入一個非負整數 n ，如果 n 是奇數就印出 $3n + 1$ ， n 是偶數則印出 $\frac{n}{2}$ 。他這樣寫：

```
// Hi everyone, I am Jack Lin
#include <iostream>
using namespace std;

int main()
{
    int n = 0;
    cin >> n;
    int r = n % 2;

    if(r == 1)
        cout << 3n + 1;
    if(r == 0)
        cout << n / 2;

    return 0;
}
```

讓我們假設使用者只會輸入一個小於 10000 的非負整數。

- (5 分) 這個程式有 syntax error 嗎？有的話，請指出來並修正它。
- (5 分) 敬傑是透過先執行 $r = n \% 2$ ，然後檢驗 $r == 1$ 是否為真來決定該印出什麼。請用白話解釋這兩個運算在做什麼，以及為什麼能達到敬傑的目標。

¹課本是 Deitel and Deitel 著的 *C++ How to Program: Late Objects Version* 第七版。

- (c) (5 分) 請修改敬傑的程式碼，修正 syntax error、把變數的名稱改成含有較高資訊量的，並且加上簡單扼要但有效的提示語 (prompt) 和註解。
- (d) (10 分) 敬傑的哥哥敬豪也在學寫程式，他不慎把 `if(r == 1)` 和 `if(r == 0)` 寫成 `if(r = 1)` 和 `if(r = 0)` 了，請問此時如果輸入 16、29、0，各會在螢幕上看到什麼結果？為什麼？

第二題

(15 分) 敬傑寫了一個程式如下：

```
#include <iostream>
using namespace std;
int main()
{
    int n, b = 0, c = 1, d = -1;
    cin >> n;
    while(d < 0){
        if(n / c == 0)
            d=b;c=c*10;b=b+1;}
    cout <<d<<"\n";return 0;
}
```

讓我們假設使用者只會輸入一個小於 1000000 的非負整數。

- (a) (5 分) 請把這個程式做良好的排版。
- (b) (10 分) 請閱讀並測試這個程式，然後解釋最後印出的 d 跟輸入的 n 之間的關係，並說明它是怎麼做到的 (也就是逐行解釋 main function 裡面的程式碼)。

第三題

(60 分) 給定一個介於 0 和 9999 之間的整數 n ，我們可以把 n 表示成

$$n = 1000a_3 + 100a_2 + 10a_1 + a_0,$$

而 $a_i \in \{0, 1, \dots, 9\}$, $i = 0, \dots, 3$ 。很顯然地，此時 a_3 表示千位數、 a_2 表示百位數，依此類推。例如 $n = 1403$ 的話，我們就有 $a_3 = 1$ 、 $a_2 = 4$ 、 $a_1 = 0$ ，以及 $a_0 = 3$ 。

有一個遊戲是這麼玩的：每個人說一個 0 到 9999 之間的整數 n_0 ，然後把 n 的中間兩位數 a_2 和 a_1 取出來，組合成一個介於 0 到 99 之間的整數 $10a_2 + a_1$ ，將之平方之後得到 n_1 ，再把 n_1 的中間兩位數取出來，平方之後得到 n_2 ，如此反覆下去，直到得到某個 n_i 它跟某個已經出現過的 n_j 一模一樣 ($i > j$)，那麼這個人的分數就是 i 。舉例來說：

1. 如果我選的數字 $n_0 = 8649$ ，那我依序就會得到 $n_1 = 64^2 = 4096$ 、 $n_2 = 9^2 = 81$ 、 $n_3 = 8^2 = 64$ 、 $n_4 = 6^2 = 36$ 、 $n_5 = 3^2 = 9$ 、 $n_6 = 0^2 = 0$ ，以及 $n_7 = 0^2 = 0$ 。因為 n_7 跟 n_6 一樣都是 0，所以分數是 7 分。

2. 如果我選的數字 $n_0 = 5555$ ，那我依序就會得到 $n_1 = 55^2 = 3025$ 、 $n_2 = 2^2 = 4$ 、 $n_3 = 0^2 = 0$ 以及 $n_4 = 0^2 = 0$ 。因為 n_4 跟 n_3 一樣都是 0，所以分數是 4 分。

這個遊戲希望愈高分愈好（不然 $n_0 = 0$ 就無敵了），不過要很高分似乎不太容易。你能找到超過 10 分的 n_0 嗎？事實上，多試幾次的话會發現這個數列 $\{n_k\}_{k=0,1,\dots}$ 經常（還是總是？）趨近於 0。這是必然的嗎？

在本題中，讓我們暫時先忘掉上面這個問題，來做個簡單的程式練習就好。你將會被給予上述的整數 n_0 ，請印出 n_1 。如果 n_1 是個四位數字，就直接印出來；如果不是，請在前面補足夠數量的 0 使得你印出的是等價於 n_1 的四位數字。舉例來說：

1. 如果程式讀入的值是 8649，你應該印出 4096。
2. 如果程式讀入的值是 3025，你應該印出 0004。

輸入輸出格式

系統會提供一共 20 組測試資料，每組測試資料裝在一個檔案裡。在每個檔案中，會有一個整數 n_0 代表起始數字， n_0 會介於 0 到 9999 之間（包含 0 跟 9999）。讀入 n_0 之後，請依照題目的規則輸出一個前面可能要補零的四位數字。

舉例來說，如果輸入是

8649

則輸出應該是

4096

如果輸入是

3025

則輸出應該是

0004

小提醒：當你要讀取 PDOGS 提供給你的輸入時，就直接使用 `cin >>` 運算，並且想像會有個人把這些數字用鍵盤輸入給你的程式就好了。然後就直接用 `cout <<` 運算照規定輸出你的答案，就好像有個人會在「螢幕前」閱讀你的輸出結果、檢驗其正確性，並且給分。PDOGS 會自動重複這件事 20 次，每次輸入一組測試資料。每組測試資料的評分都是獨立的。由於 PDOGS 完全依照你輸出的答案做自動批改，題目沒有要求你輸出的東西就不要輸出，不然答案就會被評定為錯。舉例來說，如果剛剛那題你輸出：

Dear TA, please see my answer: 4096

或

4 0 9 6

那 PDOGS 就會認為你的答案是錯的。

你上傳的原始碼裡應該包含什麼

你的.cpp 原始碼檔案裡面應該包含讀取測試資料、做運算，以及輸出答案的 C++ 程式碼。當然，你應該寫適當的註解。針對這個題目，你**不可以**使用上課沒有教過的方法。

評分原則

- 這一題的其中 40 分會根據程式運算的正確性給分。PDOGS 會編譯並執行你的程式、輸入測試資料，並檢查輸出的答案的正確性。一筆測試資料佔 2 分。
- 這一題的其中 20 分會根據你所寫的程式的品質來給分。助教會打開你的程式碼並檢閱你的程式的運算邏輯、可讀性，以及可擴充性（順便檢查你有沒有使用上課沒教過的語法，並且抓抓抄襲）。請寫一個「好」的程式吧！

第四題 (bonus)

(20 分) 承上題，你將會被給予上述的整數 n_0 ，以及一個次數 $k \in \{1, 2, \dots, 10\}$ 。這兩個數字會被放在同一列，並且以一個空白隔開。請按照上述的規則產生 n_1, n_2, \dots, n_k ，並且依序把這 k 個整數都印出來。同樣地，每個印出來的數字前面都要補零。兩個印出的數字之間用一個空白隔開。

舉例來說，如果輸入是

```
8649 5
```

則輸出應該是

```
4096 0081 0064 0036 0009
```

如果輸入是

```
1234 8
```

則輸出應該是

```
0529 2704 4900 8100 0100 0100 0100 0100
```

輸出時請務必小心，錯一個空白就是全錯。請注意這一行的最後面沒有任何符號也沒有空白。

小提醒：除非有特別說明，否則在本學期所有作業與考試的輸入與輸出中都使用半形符號。

針對這個題目，你**可以**使用任何方法。這一題的 20 分都根據程式運算的正確性給分，一筆測試資料佔 2 分。