

商管程式設計 (106-1)

期末考

題目設計：孔令傑

國立臺灣大學資訊管理學系

請至 PDOGS (<http://pdogs.ntu.im/judge/>) 為每一題各上傳一份 Python 3.6 原始碼 (以複製貼上原始碼的方式上傳)。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。**作弊被確認者，整門課的成績將直接被評定為 F，沒有第二種可能 (也不能停修)。**

所有的分數都根據程式運算的正確性給分。PDOGS 會直譯並執行你的程式、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。在所有的題目中，你都可以使用任何方法 (包括課堂上沒教過的)。這次考試的截止時間是 **12 月 25 日下午 12 點 10 分**。

注意：因為一些不明原因，在 PDOGS 上使用 list 的 `split(' ')`，可能會被錯誤地執行而達不到應有的效果。因此在這次考試中，請使用 `split()` 而不要用 `split(' ')`。

第一題

(20 分) 給定三個正整數，請判斷這三個正整數能否組合成一個三角形。如果可以組合成三角形，請判斷會組合成直角三角形、銳角三角形，還是鈍角三角形¹

系統會提供一共 10 組測試資料，每組測試資料裝在一個檔案裡。在每個檔案中，第一列是 3 個整數 a 、 b 和 c ，都介於 1 和 100 之間，任兩個數字之間被一個空白鍵隔開。讀入這些資料之後，請印出一個整數，不能組成三角形印出 0、能組成直角三角形印出 1、能組成銳角三角形印出 2、能組成鈍角三角形則印出 3。

舉例來說，如果輸入是

```
3 4 6
```

則輸出應該是

```
3
```

如果輸入是

```
100 1 99
```

則輸出應該是

```
0
```

¹組成三角形的條件是任意兩邊和要大於第三邊；組成直角三角形的條件是較短的兩邊的平方和要等於最長邊的平方；組成銳角三角形的條件是較短的兩邊的平方和要大於最長邊的平方；組成鈍角三角形的條件是較短的兩邊的平方和要小於最長邊的平方。

第二題

(20 分) 有 n 個學生修了一門課，其中若干人沒有考期末考。在一張表上，我們記錄了這 n 個學生的學號和成績，學號是十位英數字元的字串，成績是整數，但缺考的人的成績欄位會是一個字串「NA」。

系統會提供一共 10 組測試資料，每組測試資料裝在一個檔案裡。在每個檔案中，第一列是一個整數 n ，第 $i+1$ 列是兩個「值」 a_i 和 s_i ，其中 a_i 是十位英數字元的字串，代表第 i 個學生的學號， s_i 如果是整數，會是介於 0 到 100 之間的整數，代表第 i 個學生的成績，或是一個字串「NA」，代表學生沒參加考試。班上至少會有一人參加考試。兩個值之間被一個空白鍵隔開。讀入這些資料之後，請依序印出缺考的人數、將缺考者視為 0 分的全班平均分（無條件捨去到整數位），以及不計入缺考者的全班平均分（無條件捨去到整數位；是捨去 NA 的人，不是捨去 0 分的人）。兩個值中間用一個空白字元隔開，最後一個值後面沒有空白。

舉例來說，如果輸入是

```
6
B90705001 100
B90705002 90
B90705003 78
B90705004 NA
B90705005 0
B90705006 NA
```

則輸出應該是

```
2 44 67
```

如果輸入是

```
3
B90705001 100
B90705002 91
B90705003 81
```

則輸出應該是

```
0 90 90
```

第三題

(30 分) 承上題，現在給定全班學生的考試記錄後，我們想要跨系所做統計。在學號的十位英數字元中，第四位到第六位的子字串是系所代碼，同一個系所的學號的系所代碼就會相同，不同系所的學號的系所代碼就會不同。給定如上一題的資料後，請找出所有系所，並針對每個系所計算扣掉缺考者之後的平均數。

系統會提供一共 15 組測試資料，每組測試資料裝在一個檔案裡，格式方面前 $n + 1$ 列同上一題，第 $n + 2$ 列則包含若干個字串，每個字串有三個字元，代表一個系所代碼，兩個字串之間以一個空白字元分開。第 $n + 2$ 列最多含有 10 個字串。讀入這些資料之後，請按照第 $n + 2$ 的系所代碼的順序，印出該系所學生扣掉缺考者後的平均數，無條件捨去到整數位。如果該系所沒有學生來修課，或者沒有學生參加考試，則印出 NA。兩個值（包含 NA）中間用一個空白字元隔開，最後一個值後面沒有空白。

舉例來說，如果輸入是

```
6
B90705001 100
B90704002 90
B90705003 77
B90705004 NA
B90706005 0
B90706006 NA
706 704 799 705
```

則輸出應該是

```
0 90 NA 88
```

如果輸入是

```
3
B90705001 100
B90705002 91
B90705003 81
999 888 777
```

則輸出應該是

```
NA NA NA
```

第四題

(30 分) 一個資料檔中記錄著一個股票的單日開盤價與收盤價。該檔案一共有三個欄位：日期、開盤價、收盤價。我們定義單日漲幅為該日收盤價減該日開盤價。在整個資料檔之外，我們會另外給你一個分析的年份、月份與天數，請從該年該月一日起，在往後該天中找出單日最高漲幅與單日最高跌幅。給定的資料檔中未必包含所有日期，但日期有從早到晚排序。

系統會提供一共 15 組測試資料，每組測試資料裝在一個檔案裡。第一列含有一個整數 n 表示檔案共有 $n + 1$ 列、一個數字代表西元年份、一個英文字串代表月份，以及一個正整數代表天數。第二列起的每一列含有一個 YYYY/MM/DD 格式的十個字元字串代表日期、精準到兩位小數的正數代表開盤價，以及精準到兩位小數的正數代表收盤價。每一列的兩個值之間都有一個空白字元。讀入資料後，請根據題目規則依序印出搜尋範圍內的單日最高漲幅與單日最高跌幅（兩個數字都應該是非負的），精確到小數點後第二位。兩個值中間用一個空白字元隔開，最後一個值後面沒有空白。如果搜尋範圍內每一天漲幅都是非負的，則單日最高跌幅為 0.00；如果搜尋範圍內每一天漲幅都是非正的，則單日最高漲幅亦為 0.00。你可以假設西元年份是正確合理的、月份的字串都是正確的英文全稱且全部小寫、天數不超過 1000、日期格式都是正確的、日期都是合理的、給定的搜尋範圍中至少有一天的價格有被記錄、日期有從早到晚排序，且日期不重複。你不需要檢查前一天的收盤價是否為後一天的開盤價。

舉例來說，如果輸入是

```
6 2017 december 10
2017/11/30 100.00 10.99
2017/12/01 10.99 11.00
2017/12/02 11.00 15.06
2017/12/10 17.01 15.02
2017/12/11 15.02 100.02
2018/01/05 107.02 1.87
```

則輸出應該是

```
4.06 1.99
```

請注意 12/11 不列入考慮，因為我們只考慮 12/1 到 12/10 內的漲跌。如果輸入是

```
6 2017 september 10
2017/08/30 100.00 10.99
2017/09/01 10.99 11.00
2017/09/02 11.00 15.06
2017/09/10 17.01 19.02
2017/09/11 15.02 100.02
2018/10/05 107.02 1.87
```

則輸出應該是

```
4.06 0.00
```