

I. 作者簡介

- A. 姓名：胡瑀真
- B. 系級：測量系 116 級
- C. 學號：F64126147
- D. e-mail：F64126147@gs.ncku.edu.tw

II. 程式簡介

- A. 依據題意，宣告兩個 C 型態字串陣列 P 和 Q，並將字串長度令為 21，使 P 和 Q 皆可輸入長度為 20 的字串（因結束字元 \0 做結尾並佔一個長度單位）。在使用者使用 cin 輸入 P 和 Q 後，便使用 cout 印出所求 P。

```
7      cout<<"請輸入兩個字串P與Q："<<endl;
8      char P[21], Q[21];
9      cin>>P>>Q;
10     //A
11     cout<<"A.\t";
12     cout<<P<<endl;
```

圖一、A 題之程式碼

- B. 宣告一個 C++型態字串 P2 等於 C 型態字串陣列 P，以使用「字串名稱.length()」的功能。再運用 for 迴圈，令 i 為 0，每次迴圈 i 即加 1，並且在不小於 P2 的長度時才結束迴圈，使迴圈執行 P2 長度的次數。每次迴圈中運用 if 敘述判斷 P2[i] 的 ascii code 是否位於 48 和 57 之間，若是，則用 cout 印出 P2[i]。

```
14     //B
15     cout<<"B.\t";
16     string P2=P;
17     for(int i=0;i<P2.length();i++){
18         if(P2[i]>=48 && P2[i]<=57){
19             cout<<P2[i];
20         }
21     }
22     cout<<endl;
```

圖二、B 題之程式碼

C. 使用 cout 印出字串陣列 P 和 Q，即印出兩字串之串接字串。

```
24 //C
25 cout<<"C.\t";
26 cout<<P<<Q<<endl;
```

圖三、C 題之程式碼

D. 同 B 題概念，宣告一個 C++ 型態字串 Q2 等於 C 型態字串陣列 Q，以使用「字串名稱.length()」的功能。

運用兩個 for 迴圈，在第一個迴圈中令 i 為 Q2 的長度減 1，每次迴圈 i 減 1，在小於 0 時才結束迴圈，使迴圈執行 Q2 長度的次數。又每次迴圈皆用 cout 印出 Q2[i]，使 Q2 字串以降幂方式印出。第二個迴圈則改用字串 P2 並同理編寫。兩迴圈皆運行結束後，即可見所求。

```
28 //D
29 cout<<"D.\t";
30 string Q2=Q;
31 ✓ for(int i=Q2.length()-1;i>=0;i--){
32     cout<<Q2[i];
33 }
34 ✓ for(int i=P2.length()-1;i>=0;i--){
35     cout<<P2[i];
36 }
37 cout<<endl;
```

圖四、D 題之程式碼

E. 宣告一整數變數 number 為 0，負責儲存 P、Q 字串共有幾個數字。

運用兩個 for 迴圈，第一個迴圈中令整數 i 為 0，每次迴圈 i 即加 1，並且在不小於 P2 的長度時才結束迴圈，使迴圈執行 P2 長度的次數。每次迴圈即運用 if 敘述判斷 P2[i] 的 ascii code 是否位於 48 和 57 之間，若是，則使 number 加 1。第二個迴圈則改用字串 P2 並同理編寫。兩迴圈皆運行結束後，使用 cout 印出 number 之值，即為所求。

```
39 //E
40 cout<<"E.\t";
41 int number=0;
42 ✓ for(int i=0;i<P2.length();i++){
43 ✓     if(P2[i]>=48 && P2[i]<=57){
44         number+=1;
45     }
46 }
47 ✓ for(int i=0;i<Q2.length();i++){
48 ✓     if(Q2[i]>=48 && Q2[i]<=57){
49         number+=1;
50     }
51 }
52 cout<<number<<endl;
```

圖五、E 題之程式碼

F. 宣告一整數變數 sum 為 0，負責儲存 P、Q 字串的數字總和。

同 E 題概念編寫兩個 for 迴圈，第一個迴圈每次運行時，若 P2[i] 符合 if 敘述內容，則使 sum 加上 P2[i] 的 ascii code 減 48 (因 ascii code 減去 48 才會是字串中阿拉伯數字的值)，第二個迴圈改用字串 Q2[i] 並同理編寫。兩迴圈皆運行結束後，使用 cout 印出 sum 之值，即為所求。

```
54 //F
55 cout<<"F.\t";
56 int sum=0;
57 for(int i=0;i<P2.length();i++){
58     if(P2[i]>=48 && P2[i]<=57){
59         sum+=P2[i]-48;
60     }
61 }
62 for(int i=0;i<Q2.length();i++){
63     if(Q2[i]>=48 && Q2[i]<=57){
64         sum+=Q2[i]-48;
65     }
66 }
67 cout<<sum<<endl;
```

圖六、F 題之程式碼

G. 使用雙重 for 迴圈。第一層迴圈中令整數 i 為 0，每次迴圈 i 即加 1，並且在不小於 P2 的長度時才結束迴圈，使迴圈執行 P2 長度的次數。第二層迴圈中令整數 j 為 0，每次迴圈 j 即加 1，並且在不小於 Q2 的長度時才結束迴圈，使迴圈執行 Q2 長度的次數。每次迴圈即運用 if 敘述判斷 Q2[i] 是否等於 P2[j]，若是，則用 cout 印出 P2[j]。在雙重迴圈運行完畢後，即可見所求。

為方便 H 題的編寫，在進入雙重迴圈前先令一 C 型態字串陣列 both，用以儲存 G 題答案 (P、Q 字串之共同子字串)，並將字串長度令為 21 (因兩字串最多只會完全相同 20 個字元)。再宣告一整數 a 為 0 以便字串陣列 both 的儲存。

每次迴圈中，若 if 敘述判斷為是，則令 both[a] 等於 P2[j]，再使 a 加 1。在雙重迴圈運行完畢後，宣告一個 C++ 型態字串 both2 等於 C 型態字串陣列 both，以使用「字串名稱.length()」的功能。

```

69      //G
70      cout<<"G.\t";
71      char both[21];
72
73      int a=0;
74      for(int i=0;i<Q2.length();i++){
75          for(int j=0;j<P2.length();j++){
76              if(Q2[i]==P2[j]){
77                  cout<<P2[j];
78                  both[a]=P2[j];
79                  a++;
80              }
81          }
82      }
83      string both2=both;
84      cout<<endl;

```

圖七、G 題之程式碼

H. 宣告一整數 b 為 0，以便後續差集子字串的判斷。

類似 G 題使用雙重 for 迴圈，第一層迴圈中令整數 i 為 0，每次迴圈 i 即加 1，並且在不小於 P2 的長度時才結束迴圈，使迴圈執行 P2 長度的次數。第二層迴圈中令整數 j 為 0，每次迴圈 j 即加 1，並且在不小於 both2 的長度時才結束迴圈，使迴圈執行 both2 長度的次數。

每次迴圈中用 if 敘述判斷 P2[i] 是否不等於 both2[j]，若是，則使 b 加 1。每次第二層迴圈運行完畢後，使用 if 敘述判斷 b 是否等於 both2 的長度，若等於則代表 P2[i] 不和 both2 中的任一字元相同，因此用 cout 印出 P2[i]。此 if 敘述執行完畢後，重新令 b 為 0 以便下次判斷。在雙重迴圈運行完畢後，即可見所求。

```

86      //H
87      cout<<"H.\t";
88      int b=0;
89      for(int i=0;i<P2.length();i++){
90          for(int j=0;j<both2.length();j++){
91              if(P2[i]!=both2[j]){
92                  b++;
93              }
94          }
95          if(b==both2.length()){
96              cout<<P2[i];
97          }
98          b=0;
99      }
100     cout<<endl;

```

圖八、H 題之程式碼

- I. 運用 for 迴圈，令 i 為 0，每次迴圈 i 即加 1，並且在不小於 P2 的長度時才結束迴圈，使迴圈執行 P2 長度的次數。每次迴圈皆運用 if 敘述判斷 P2[i] 的 ascii code 是否位於 48 和 57 之間，若是，則強制轉型 P2[i] 為整數型態並加 1，再將以上內容強制轉型為 char 後，用 cout 印出。若不符合 if 敘述，則執行 else 敘述中內容：直接印出 P2[i]。迴圈結束後，即可見所求。

```
102 //I
103 cout<<"I.\t";
104 ✓ for(int i=0;i<P2.length();i++){
105 ✓     if(P2[i]>=48 && P2[i]<=57){
106         cout<<char(int(P2[i]+1));
107     }
108 ✓     else{
109         cout<<P2[i];
110     }
111 }
112 cout<<endl;
```

圖九、I 題之程式碼

- J. 運用 for 迴圈，令 i 為 0，每次迴圈 i 即加 1，並且在不小於 Q2 的長度時才結束迴圈，使迴圈執行 Q2 長度的次數。每次迴圈皆運用 if 敘述判斷 Q2[i] 的 ascii code 是否位於 65 和 90 之間，若是，則強制轉型 Q2[i] 為整數型態並加 32，再將以上內容強制轉型為 char 後，用 cout 印出。若不符合 if 敘述，則執行 else if 敘述判斷 Q2[i] 的 ascii code 是否位於 97 和 122 之間，若是，則強制轉型 Q2[i] 為整數型態並減 32，再將以上內容強制轉型為 char 後，用 cout 印出。若不符合 if 和 else if 敘述，則執行 else 敘述並直接印出 Q2[i]。迴圈結束後，即可見所求。

```
114 //J
115 cout<<"J.\t";
116 ✓ for(int i=0;i<Q2.length();i++){
117 ✓     if(Q2[i]>=65 && Q2[i]<=90){
118         cout<<char(int(Q2[i]+32));
119     }
120 ✓     else if(Q2[i]>=97 && Q2[i]<=122){
121         cout<<char(int(Q2[i]-32));
122     }
123 ✓     else{
124         cout<<Q2[i];
125     }
126 }
```

圖十、J 題之程式碼

K. 整體編寫巧思：

1. 宣告一字串 both 並運用 G 題程式碼，使 both 為 G 題的答案，以方便編寫 H 題程式。
2. 在程式一開始，使用 while 迴圈，使每次輸入 P 和 Q，即運行 A 到 J 各題，以方便助教們與我輸入測資測試程式。

5	int main() {	127	cout<<endl<<endl;
6	while(true){	128	}
7	cout<<"請輸入兩個字串P與Q："<<endl;	129	return 0;
		130	}

圖十一、左邊為程式初始，右邊為程式末尾

3. 運用老師在課堂所講的概念「在迴圈中命名的參數隨迴圈結束而消失」，令所有第一層 for 迴圈的初始值皆為 i，第二層 for 迴圈的初始值皆為 j，以方便編寫和查閱。
4. 印出題號時，運用跳脫序列「\t」達到跳格的效果，以符合輸出範例之格式。

III. 結果簡介

A. 輸入格式與運作流程：

輸入兩字串 P 與 Q，兩字串以 \n (enter 鍵) 隔開 (如圖十二)。輸入完後按下 enter 鍵，即可看見輸出畫面。

又因使用 while 迴圈，故可在看到輸出畫面的同時，看見輸入畫面並再次輸入 (如圖十三)。

Run
請輸入兩個字串P與Q： AbCdE1234 AcE34
Run
請輸入兩個字串P與Q： AbCdE1234 AcE34 A. AbCdE1234 B. 1234 C. AbCdE1234AcE34 D. 43EcA4321EdCbA E. 6 F. 17 G. AE34 H. bCd12 I. AbCdE2345 J. aCe34 請輸入兩個字串P與Q：

圖十二 (上) 和十三 (下)、輸入格式

B. 結束畫面：第一筆為範例測資，第二、三比為自訂測資

▼

Run

請輸入兩個字串P與Q：

AbCdE1234
AcE34

A. AbCdE1234
B. 1234
C. AbCdE1234AcE34
D. 43EcA4321EdCbA
E. 6
F. 17
G. AE34
H. bCd12
I. AbCdE2345
J. aCe34

請輸入兩個字串P與Q：

BDUdj3782k
b0347UDK

A. BDUdj3782k
B. 3782
C. BDUdj3782kb0347UDK
D. KDU7430bk2873jdUDB
E. 7
F. 34
G. 37UD
H. Bdj82k
I. BDUdj4893k
J. Bo347udk

請輸入兩個字串P與Q：

37yaiH
D03AJKH

A. 37yaiH
B. 37
C. 37yaiHD03AJKH
D. HKJA30DHiay73
E. 3
F. 13
G. 3H
H. 7yai
I. 48yaiH
J. do3ajkh

請輸入兩個字串P與Q：

圖十四、輸出畫面