



全國高級中等學校 107 學年度商業類學生技藝競賽

【程式設計】職種【術科】模擬試題卷

選手證號碼：_____ 姓名：_____

各個子題均提供 2 組測試輸入檔，檔名分別是「in1.txt」及「in2.txt」。選手製作的程式，應依序讀入「in1.txt」及「in2.txt」檔，程式執行後，並產生 1 個輸出檔「out.txt」。(即，每個程式讀入 2 個輸入檔，產生 1 個輸出檔。)在輸出檔中，選手應先輸出「in1.txt」產生的結果，再輸出「in2.txt」的結果，兩組結果間用 1 行「空白行」隔開。不影響結果的空白鍵，不列入扣分。若程式執行檔執行結果未依序、不全或無法執行，該子題以零分計算。

題目	子題	題目內容	
Problem1	子題 1	剪刀、石頭、布	
	子題 2	邏輯表示式	
Problem2	子題 1	排列	
	子題 2	十進位轉二進位	
Problem3	子題 1	集合	102 2-1
	子題 2	模數 (Modulo)	
Problem4	子題 1	最長共同子序列(Longest common subsequence)	105 4-1
	子題 2	最小成本生成樹	104 4-2

Problem 1 :

子題 1：剪刀、石頭、布

剪刀(Y)、石頭(O)、布(P)是一種猜拳遊戲

剪刀(Y)、石頭(O)、布(P)各代表一個「武器」，互相剋制的原則是：

剪刀(Y)被石頭(O)砸碎（石頭(O)勝利）

布(P)被剪刀(Y)剪開（剪刀(Y)勝利）

石頭(O)被布(P)包裹（布(P)勝利）

如果雙方出示了一樣的手勢，就是平手

輸入說明：

第 1 列的數字 n 代表有幾組資料要測試， $2 \leq n \leq 20$ ，第二列起為測試資料，每列為一組測試資料，每組測試資料為二位玩家 1、玩家 2 出的猜拳。

輸出說明：

每組測試資料輸出一列，判斷每組玩家 1、玩家 2 的猜拳，輸出結果為 1、2 或 0，分別代表勝利者為玩家 1、玩家 2 或平手 0。

輸入檔案 1：【檔名：in1.txt】

3
Y, P
Y, O
Y, Y

輸入檔案 2：【檔名：in2.txt】

3
O, P
O, O
O, Y

輸出範例：【檔名：out.txt】

1
2
0

2
0
1

子題 2：邏輯表示式

此問題為給定「數字或含小數後一位數」、「+」、「-」及「==」組成「邏輯表示式」，判斷此邏輯表示式的敘述是否為真？在此子題中，邏輯表示式的內容只包括「數字或含小數後一位數」、「+」、「-」、「==」及「空格」。其中「+」代表數字運算的「加法」，而「-」代表「減法」；「==」代表邏輯運算的「是否相等」。

輸入說明：

第一列的數字 n 代表有幾筆資料要測試， $2 \leq n \leq 5$ ，之後每列為每筆的測試資料，為邏輯表示式。輸入的字串，不存在邏輯表示式語法的錯誤，選手可不必另外檢查。每行有一個邏輯表示式，每行最多 80 個字。「數字或含小數後一位數」 < 100 。在邏輯表示式中的空白不具運算意義。

輸出說明：

每筆測試資料輸出一列。在「==」運算前後比較值「相同」時，其結果為「T」；若「不同」時，其結果為「F」。輸出均為大寫，選手請注意。

輸入檔案 1：【檔名：in1.txt】

4
25 + 4 == 29
18 + 30 == 58 - 10
20 + 10 - 50 == 40 - 60
20 == 10 + 10 + 10 + 10 - 32

輸入檔案 2：【檔名：in2.txt】

4
24 - 4 + 10 == 10 - 40
0.2 + 0.2 == 0.4
1.1 + 1.2 == 2.3
0.3 - 0.2 == 0.1

輸出範例：【檔名：out.txt】

T
T
T
F

F
T
T
T

Problem 2：數學問題

子題 1：排列。

在排列問題中將一組數字進行排列，可以得到不同的數字順序；例如 1,2,3,4 這數的排列有 24 組，數列順序如下：

(1)1234	(7)2134	(13)3124	(19)4123
(2)1243	(8)2143	(14)3142	(20)4132
(3)1324	(9)2314	(15)3214	(21)4213
(4)1342	(10)2341	(16)3241	(22)4231
(5)1423	(11)2413	(17)3412	(23)4312
(6)1432	(12)2431	(18)3421	(24)4321

輸入說明：

第一列的數字 n 代表有幾筆資料要測試， $2 \leq n \leq 5$ ，之後每列為每筆的測試資料，共有四個 0~9 不同的整數 i, j, k, l 。各個數字間以“,”隔開。

輸出說明：

每筆測試資料輸出一列。輸出以 i, j, k, l 這四值排列合順序中，找出第 2 大的值。

輸入檔案 1：【檔名：in1.txt】

2
1,2,3,4
7,8,4,3

輸入檔案 2：【檔名：in2.txt】

3
6,2,9,5
7,0,3,9
2,3,5,0

輸出範例：【檔名：out.txt】

4312
8734

9625
9703
5302

子題 2：十進位轉二進位

設計一含小數點的十進位轉二進位的程式。給定一個十進位數字(有小數部分,整數部分 ≤ 100 ,小數點部分 ≤ 0.96875),將其轉換成二進位數字。

輸入說明：

第一列的數字 n 代表有幾筆資料要測試, $2 \leq n \leq 5$, 之後每列為每筆的測試資料, 含小數點的十進位 (有小數部分, 整數部分 ≤ 100 , 小數點部分 ≤ 0.96875)。

輸出說明：

每筆測試資料輸出一列, 將含小數點的十進位轉二進位, 二進位小數點部分固定取五位。

輸入檔案 1：【檔名：in1.txt】

```
2
10.5
90.25
```

輸入檔案 2：【檔名：in2.txt】

```
3
10.1
33.7
21.25
```

輸出範例：【檔名：out.txt】

```
1010.10000
1011010.01000

1010.00011
100001.10110
10101.01000
```

Problem 3 :

子題 1：集合。

集合裡的「東西」，叫作元素(e)。若 x 是集合 A 的元素，記作 $x \in A$ 。若集合 A 沒有元素，為空集合，記作 \emptyset 。

給定集合 A 及 B ，定義運算 \cup 如下： $A \cup B = \{e | e \in A \text{ OR } e \in B\}$ 。 $A \cup B$ 稱為 A 和 B 的聯集 (Union)。

$$\{1, 3\} \cup \{2, 4\} = \{1, 2, 3, 4\}$$

$$\{1, 2, 3\} \cup \{3, 4, 5\} = \{1, 2, 3, 4, 5\}$$

$$\{1, 2\} \cup \{2, 1\} = \{1, 2\}$$

給定集合 A 及 B ，定義運算 \cap 如下： $A \cap B = \{e | e \in A \text{ AND } e \in B\}$ 。 $A \cap B$ 稱為 A 和 B 的交集 (Intersection)。

$$\{1, 3\} \cap \{2, 4\} = \emptyset$$

$$\{1, 2, 3\} \cap \{3, 4, 5\} = \{3\}$$

$$\{1, 2\} \cap \{2, 1\} = \{1, 2\}$$

給定集合 A 及 B ，定義運算 $-$ 如下： $A - B = \{e | e \in A \text{ AND } e \notin B\}$ 。 $A - B$ 稱為 B 對於 A 的差集 (Set difference)。

$$\{1, 3\} - \{2, 4\} = \{1, 3\}$$

$$\{1, 2, 3\} - \{3, 4, 5\} = \{1, 2\}$$

$$\{1, 2\} - \{2, 1\} = \emptyset$$

給定集合 A 及 B ，定義運算 \oplus 如下： $A \oplus B = (A \cup B) - (A \cap B)$ 。 $A \oplus B$ 稱為 A 和 B 的對稱差集 (Symmetric difference)。

$$\{1, 3\} \oplus \{2, 4\} = \{1, 2, 3, 4\}$$

$$\{1, 2, 3\} \oplus \{3, 4, 5\} = \{1, 2, 4, 5\}$$

$$\{1, 2\} \oplus \{2, 1\} = \emptyset$$

輸入說明：

第 1 列的數字 n 代表有幾筆資料要測試， $1 \leq n \leq 5$ ，第二列起為測試資料，之後每列為每筆的測試資料，分別代表 A 和 B 二組集合，以“ $\{ \}$ ”代表一組集合，集合與集合間以“, ”隔開，集合中的元素亦以“, ”隔開。集合內的元素沒有排序，集合內的元素為數字 0 到數字 9 任選，集合內的元素不會重複，且 A 和 B 二組集合不會是空集合。

輸出說明：

每筆測試資料輸出一列。輸出每筆測試資料 A 和 B 二組集合的聯集 $A \cup B$ 、交集 $A \cap B$ 、差集 $A - B$ 及對稱差集 $A \oplus B$ (集合內的元素由小到大，依序輸出)，以“ $\{ \}$ ”代表一組集合，並以“, ”隔開，集合中的元素以“, ”隔開，若集合為空集合則輸出 N (不用加“ $\{ \}$ ”)。

輸入檔案 1：【檔名：in1.txt】

2

{1, 3}, {2, 4}

{1, 2, 3}, {3, 4, 5}

輸入檔案 2：【檔名：in2.txt】

1

{1, 2}, {2, 1}

輸出範例：【檔名：out.txt】

{1, 2, 3, 4}, N, {1, 3}, {1, 2, 3, 4}

{1, 2, 3, 4, 5}, {3}, {1, 2}, {1, 2, 4, 5}

{1, 2}, {1, 2}, N, N

子題 2：模數 (Modulo)

在程式設計中取餘數的運算(不考慮運算的商)，模數 (Modulo) MOD 運算，它的定義是取得兩個整數相除後結果的餘數。

令一個整數 n 與正整數 a ，並 n 除以 a 得到商為 q ，與餘數 b ，如下：

$$\begin{aligned} n &= a \times q + b, 0 \leq b < a, q = \lfloor n/a \rfloor \\ -11 \text{ MOD } 3 &= 1, -11 = 3 \times (-4) + 1 \\ -1 \text{ MOD } 13 &= 12, -1 = 13 \times (-1) + 12 \end{aligned}$$

例如: $19 \bmod 7 = 5$ ，因為 19 除以 7 商 2 餘 5。餘數 5 即執行 MOD 運算後的結果。

例如: $54 \bmod 24 = 6$ 、 $-1 \bmod 13 = 12$ 、 $-2 \bmod 11 = 9$ 。

輸入說明：

第一列的數字 m 代表有幾筆資料要測試， $2 \leq m \leq 5$ ，之後每列為每筆的測試資料，共有二個整數 ≤ 65535 ，分別代表整數 n 與正整數 a 。各個整數間以“,” 隔開。

輸出說明：

每筆測試資料輸出一列。輸出取餘數的運算。

輸入檔案 1：【檔名：in1.txt】

2
19, 7
54, 24

輸入檔案 2：【檔名：in2.txt】

3
-11, 3
-1, 13
-2, 11

輸出範例：【檔名：out.txt】

5
6

1
12
9

Problem 4 :

子題 1：最長共同子序列(Longest common subsequence)。

「最長共同子序列」 (Longest Common Subsequence, LCS)。LCS 是兩個序列(sequence)，在各自所有的子序列(subsequence)之中，一模一樣而且最長的那個子序列(subsequence)。

給 2 個字串，請你輸出他們的最長共同子序列 (Longest common subsequence) 的長度。也就是說，在這兩個字串各自所有的子序列之中，內容相同而且長度最長的那個子序列。舉例來說有兩個字串 abcdghxy 和 aedfhrz，它們的最長共同子序列為 adh，長度為 3。

j		0	1	2	3	4	5	6	7
i		Y	a	e	d	f	h	r	z
0	X	0	0	0	0	0	0	0	0
1	a	0	<div><div>↖</div><div><u>1</u></div></div>	<div><div>←</div><div><u>1</u></div></div>	<div><div>←</div><div><u>1</u></div></div>	<div><div>←</div><div><u>1</u></div></div>	<div><div>←</div><div><u>1</u></div></div>	<div><div>←</div><div><u>1</u></div></div>	<div><div>←</div><div><u>1</u></div></div>
2	b	0	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>
3	c	0	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>
4	d	0	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↘</div><div><u>2</u></div></div>	<div><div>←</div><div><u>2</u></div></div>	<div><div>←</div><div><u>2</u></div></div>	<div><div>←</div><div><u>2</u></div></div>	<div><div>←</div><div><u>2</u></div></div>
5	g	0	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>
6	h	0	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↘</div><div><u>3</u></div></div>	<div><div>←</div><div><u>3</u></div></div>	<div><div>←</div><div><u>3</u></div></div>
7	x	0	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>3</u></div></div>	<div><div>↑</div><div><u>3</u></div></div>	<div><div>↑</div><div><u>3</u></div></div>
8	y	0	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>1</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>2</u></div></div>	<div><div>↑</div><div><u>3</u></div></div>	<div><div>↑</div><div><u>3</u></div></div>	<div><div>↑</div><div><u>3</u></div></div>

<http://lcs-demo.sourceforge.net/>

定義：給定兩個序列(Sequence) $X = \langle x_1, x_2, \dots, x_m \rangle$ 和 $Y = \langle y_1, y_2, \dots, y_n \rangle$ ，求 X 和 Y 所構成的最長共同子序列 $LCS(X, Y) = Z = \langle z_1, z_2, \dots, z_k \rangle$ 為何？

[注意] 子字串(Substrings)和子序列(Subsequences)的差別

例如有一條字串(String) $S = "a t g a t g c a a t"$ ，則

子字串(Substrings) of S: "g a t g c", "t g c a a t"

子序列(Subsequences) of S : "a g g t" , "a a a a"

解法

$$C[i,j] = \begin{cases} 0, & \text{if } i = 0 \text{ or } j = 0 \\ C[i-1, j-1] + 1, & \text{if } i, j > 0 \text{ and } x_i = y_j \\ \text{Max}(C[i, j-1], C[i-1, j]), & \text{if } i, j > 0 \text{ and } x_i \neq y_j \end{cases}$$

若*i*或*j*為0，表示*X*或*Y*這兩條序列的其中一條為空序列。

若 $x_i = y_j$ ，則 $c[i, j]$ 所表示的序列長度，是由 $\langle x_1, x_2, \dots, x_{i-1} \rangle$ 和 $\langle y_1, y_2, \dots, y_{j-1} \rangle$ 兩序列所構成之最長共同子序列的長度，即 $c[i-1, j-1]$ 再加上1。

若 $x_i \neq y_j$ ，則 $c[i, j]$ 的序列長度是由下列兩個不同的最長共同子序列長度當中之最大值所構成：

1. $\langle x_1, x_2, \dots, x_{i-1} \rangle$ 和 $\langle y_1, y_2, \dots, y_j \rangle$ 兩序列所構成之最長共同子序列的長度 $c[i-1, j]$

2. $\langle x_1, x_2, \dots, x_i \rangle$ 和 $\langle y_1, y_2, \dots, y_{j-1} \rangle$ 兩序列所構成之最長共同子序列的長度 $c[i, j-1]$

```
function LCSLength(X[1..m], Y[1..n])
    C = array(0..m, 0..n)
    for i := 0..m
        C[i,0] = 0
    for j := 0..n
        C[0,j] = 0
    for i := 1..m
        for j := 1..n
            if X[i] = Y[j]
                C[i,j] := C[i-1,j-1] + 1
            else
                C[i,j] := max(C[i,j-1], C[i-1,j])
    return C[m,n]
```

https://en.wikipedia.org/wiki/Longest_common_subsequence_problem

輸入說明：

第一列的數字 n 代表有幾組資料要測試， $2 \leq n \leq 8$ ，第二列起為每組的測試資料，之後每二列為每組的測試資料。每組測試資料為二列字串，字串內容為英文字母或數字，大小寫的英文字母不同，每列最多有 80 個字元。

輸出說明：

對輸入的每組測試資料，輸出它們最長共同子序列的長度。

輸入檔案 1：【檔名：in1.txt】

2

ACCGATGCAGCGCTC

CCGATGA

abcdghxy

aedfhrz

輸入檔案 2：【檔名：in2.txt】

2

13567

24680

ab12

Abc1

輸出範例：【檔名：out.txt】

7

3

1

2

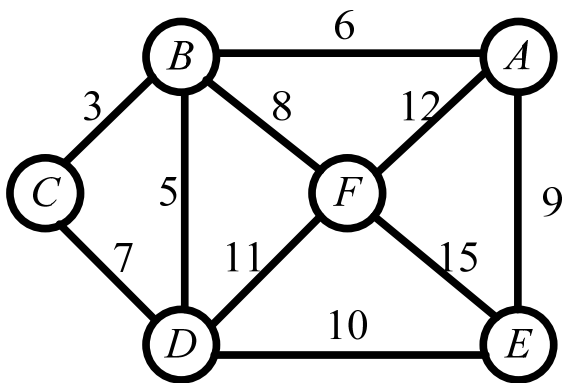
子題 2：最小成本生成樹。

以有線電視電纜的架設為例，若只能沿著街道佈線，則以街道為邊，而路口為節點，其中必然有一最小成本生成樹能使佈線成本最低。

給定一個圖形中，有許多條邊(線)連結了所有的節點，這些邊都有一個數值，代表此邊的成本。我們可以去掉圖形中的某些邊，使得剩下的邊能連結所有的節點，且邊的數量比節點的數量少 1，這些節點和留下的邊為一生成樹。一個圖形的生成樹有許多個，其中邊的總成本最低者為最小成本生成樹。最小成本生成樹不可以有循環(迴路)；最小成本生成樹不必是唯一的。

Kruskal 演算法：

假設節點數為 n ，Kruskal 演算法是將各邊先依成本(權重值)的大小由小到大排列，接著從成本(權重值)最低的邊開始加入最小成本生成樹，如果加入的邊會造成循環(迴路)則捨棄不用，直到加了 $n - 1$ 個邊為止。舉例說明如何以 Kruskal 演算法得到下圖中的最小成本生成樹：

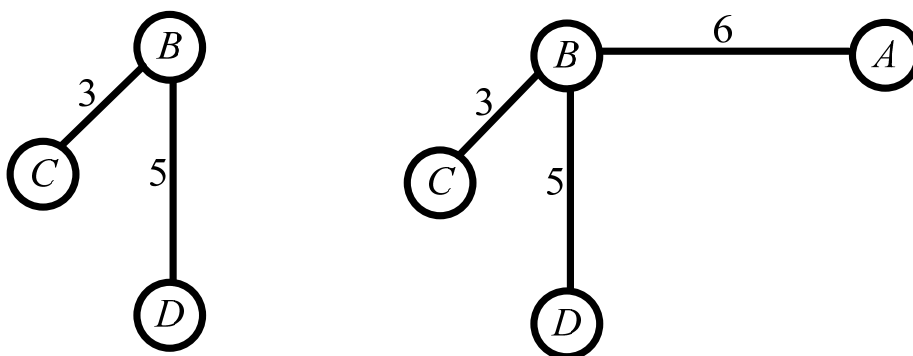


步驟一：將所有邊線的成本(權重值)列出並由小到大排序：

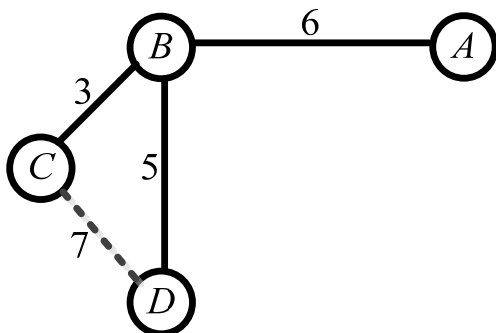
節點	節點	成本(權重值)
B	C	3
B	D	5
A	B	6
C	D	7
B	F	8
A	E	9
D	E	10
D	F	11
A	F	12
E	F	15

步驟二：選擇成本(權重值)最低的一條邊做為加入最小成本生成樹的起點，邊(B,C)->3。

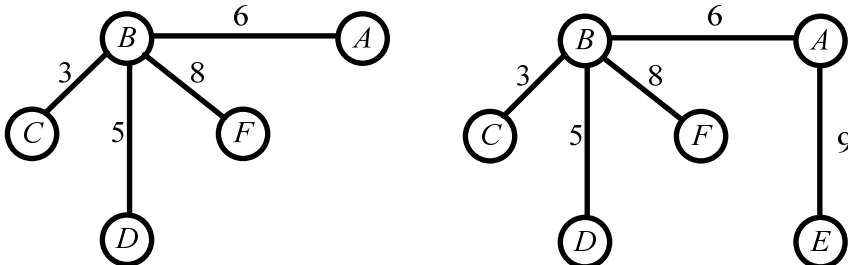
步驟三：依步驟一所建立表格，依序加入邊至最小成本生成樹。



步驟四：邊(C-D) 加入會形成迴路，所以捨棄不用。



重複步驟三和步驟四，直到加了 $n - 1$ 個邊為止，完成圖：



最小成本生成樹的值為邊的成本之總合 $3+5+6+8+9=31$

輸入說明：

第一列的數字 x 代表共有幾組資料要測試， $2 \leq x \leq 5$ 。

第二列起每一列代表一組測試資料。每組測試資料代表一圖形，內容為邊的資料。每個邊以 i, j, k 表示，其中 i 和 j 為節點的編號，為大寫英文字母(沒有順序)，代表從 i 節點和 j 節點有邊相連， k 為邊的成本(正整數) $1 \leq N \leq 65535$ ，每個邊的資料以空白() 隔開，而空白不限定一個， $|i, j|$ 為邊的個數， $3 \leq |i, j| \leq 20$ 。

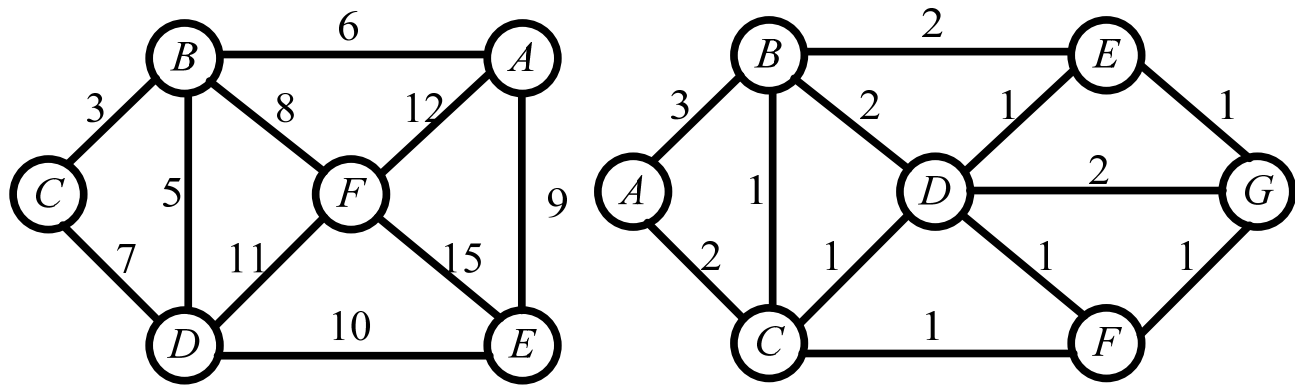
輸出說明：

每組測試資料輸出一列。輸出每組測試資料最小成本生成樹的值。

輸入檔案 1：【檔名：in1.txt】

2

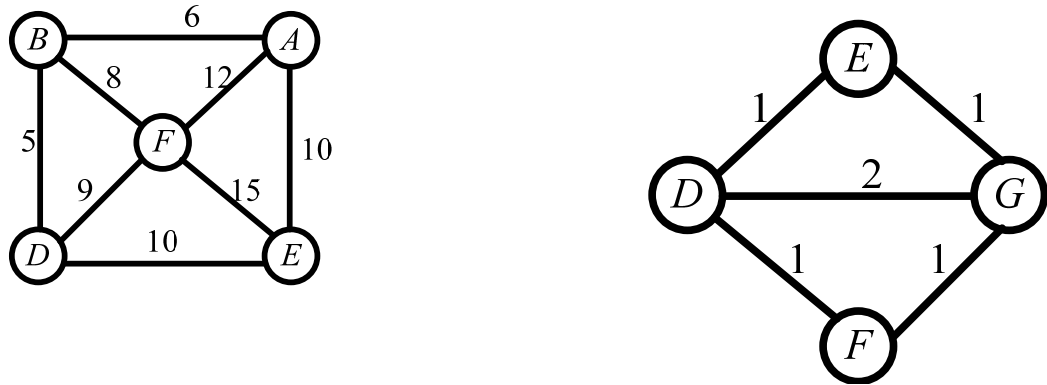
A,B,6 A,E,9 B,C,3 B,D,5 C,D,7 B,F,8 D,E,10 D,F,11 A,F,12 E,F,15
A,B,3 A,C,2 B,C,1 B,D,2 C,D,1 B,E,2 C,F,1 D,E,1 D,F,1 D,G,2 E,G,1 F,G,1



輸入檔案 2：【檔名：in2.txt】

2

B,A,6 B,F,8 B,D,5 D,E,10 D,F,9 A,F,12 A,E,10 E,F,15
D,E,1 D,G,2 D,F,1 E,G,1 F,G,1



輸出範例：【檔名：out.txt】

31

7

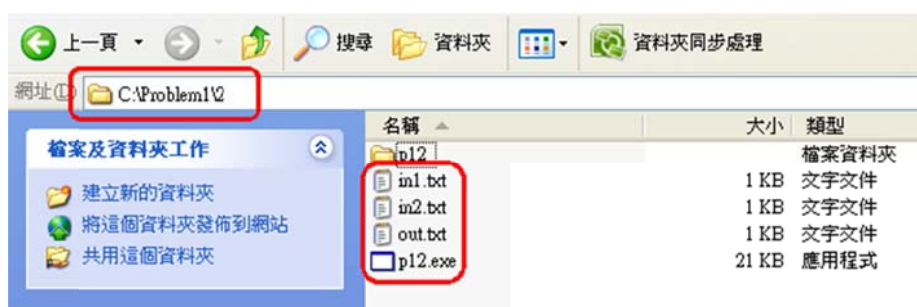
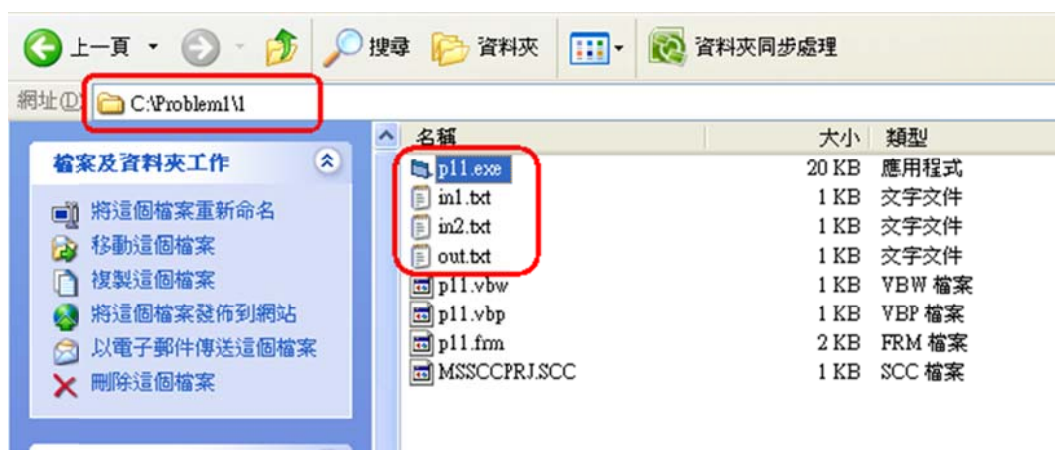
29

3

選手在作答前，先將主辦單位發放的”光碟”內資料，全部複製到選手使用的電腦硬碟 C 槽中。光碟內容包括 4 個資料夾（「Problem1」到「Problem4」），各資料夾內有「1」及「2」兩個子資料夾。在子資料夾中，已存有該子題的輸入資料檔。選手在競賽時，各子題的程式及輸出檔，應產生在硬碟對應的子資料夾中。考試結束時，在硬碟 C 槽的考試資料夾(含子題的整個專案、程式、輸入及輸出檔)，請選手全部再「備份」回原隨身碟中。

1. 各個子題均提供 2 組測試輸入檔，檔名分別是「in1.txt」及「in2.txt」。選手製作的程式，應依序讀入「in1.txt」及「in2.txt」檔，並產生 1 個輸出檔「out.txt」。(即，每個程式讀入 2 個輸入檔，產生 1 個輸出檔。)在輸出檔中，選手應先輸出「in1.txt」產生的結果，再輸出「in2.txt」的結果，兩組結果間用 1 行「空白行」隔開。不影響結果的空白鍵，不列入扣分。若程式執行檔執行結果未依序、不全或無法執行，該子題以零分計算。
2. 程式原始檔的「專案名稱」請依規定命名，程式中使用的「表單」名稱請與專案名稱命名相同。若選手在同一專案中使用多個表單，依開發環境內定名稱命名即可。輸入資料檔、輸出資料檔、程式執行檔請依”存放路徑及檔名”規定存放。若選手使用有別於 VB 之開發環境，專案（程式）名稱仍依規定。

各個子題均提供 2 組測試輸入檔，檔名分別是「in1.txt」及「in2.txt」。選手製作的程式，應將「in1.txt」「in2.txt」「out.txt」及 p??.exe(例如：p11.exe 或 p12.exe)，這四個檔案放在指定的地方。如下圖所示：



題目	子題	檔案類型	存放路徑及檔名
Problem1	子題 1	輸入資料檔	C:\Problem1\1\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem1\1\out.txt
		程式執行檔	C:\Problem1\1\p11.exe
	子題 2	輸入資料檔	C:\Problem1\2\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem1\2\out.txt
		程式執行檔	C:\Problem1\2\p12.exe

題目	子題	檔案類型	存放路徑及檔名
Problem2	子題 1	輸入資料檔	C:\Problem2\1\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem2\1\out.txt
		程式執行檔	C:\Problem2\1\p21.exe
	子題 2	輸入資料檔	C:\Problem2\2\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem2\2\out.txt
		程式執行檔	C:\Problem2\2\p22.exe

題目	子題	檔案類型	存放路徑及檔名
Problem3	子題 1	輸入資料檔	C:\Problem3\1\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem3\1\out.txt
		程式執行檔	C:\Problem3\1\p31.exe
	子題 2	輸入資料檔	C:\Problem3\2\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem3\2\out.txt
		程式執行檔	C:\Problem3\2\p32.exe

題目	子題	檔案類型	存放路徑及檔名
Problem4	子題 1	輸入資料檔	C:\Problem4\1\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem4\1\out.txt
		程式執行檔	C:\Problem4\1\p41.exe
	子題 2	輸入資料檔	C:\Problem4\2\in1.txt 和 in2.txt
		輸出資料檔	C:\Problem4\2\out.txt
		程式執行檔	C:\Problem4\2\p42.exe