



全國高級中等學校 103 學年度商業類學生技藝競賽

【程式設計】職種【術科】試卷

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。所有的輸出字母為大寫，選手請注意。

Problem 1：數學問題

子題 1：判斷二個數字是否為孿生質數。(程式執行限制時間: 2 秒) **12 分**

孿生質數的定義：如果 i 和 $i+2$ 都是質數，則稱 i 和 $i+2$ 是一對孿生素數；換句話說， i 和 $i+2$ 都是質數，而且這二個質數 i 和 $i+2$ 的差為 2。讀入欲檢查的二個數字，若二個數字是孿生質數則印出“Y”，若不是則印出“N”。

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，為二數字 x, y ， $2 \leq x, y \leq 65535$ 。

輸出說明：

每筆測試資料輸出一列。判斷每筆測試資料是否為孿生質數，若此二個數字是孿生質數則印出 Y，若不是則印出 N。**請選手注意，這題輸出 Y 和 N 全部大寫。**

輸入檔案 1 :【檔名：in1.txt】

5
2,3
11,13
12,14
3,5
5,7

輸入檔案 2 :【檔名：in2.txt】

5
17,19
29,31
15,17
19,21
41,43

輸出範例 :【檔名：out.txt】

N
Y
N
Y
Y

Y
Y
N
N
Y

子題 2：買郵票。(程式執行限制時間: 2 秒) 10 分

小朋友要到郵局買郵票，假設要買兩種不同面值的郵票，但有可能會剛好遇到某個面值的郵票已售完。題目會告訴你買了兩種不同面值的郵票共 a 張，這兩種郵票面值分別為 b 元和 c 元($b < c$)，共花了 d 元。輸入資料為四個整數 a, b, c, d ， $1 \leq a, b, c, d \leq 300$ ，整數與整數之間用一個逗號隔開。請寫程式算出 $bx + cy = d$ 的 x, y 。

假設 1：小明到郵局買郵票，買了兩種不同面值的郵票共 10 張，這兩種郵票分別為 5 元和 20 元，共花了 110 元。則輸入資料 a, b, c, d ，分別為 10, 5, 20, 110；

若 5 元郵票有 x 張，20 元郵票有 y 張，算出 x, y ：

$$5x + 20y = 110$$

則這組輸入資料之解為： $x = 6, y = 4$ ； x, y 應為大於 0 或等於 0 的整數， $0 \leq x, y$ and $x, y \in N$ 。

假設 2：小華到郵局買郵票，買了兩種郵票共 12 張，這兩種郵票分別為 3 元和 10 元，共花了 92 元。則輸入資料 a, b, c, d ，分別為 12, 3, 10, 92；

若 3 元郵票有 x 張，10 元郵票有 y 張，算出 x, y ：

$$3x + 10y = 92$$

則這組輸入資料之解為： $x = 4, y = 8$ 。

假設 3：張三到郵局買郵票，買了兩種郵票共 15 張，這兩種郵票分別為 5 元和 10 元，共花了 150 元。則輸入資料 a, b, c, d ，分別為 15, 5, 10, 150；

若 5 元郵票有 x 張，10 元郵票有 y 張，算出 x, y ：

$$5x + 10y = 150$$

則這組輸入資料之解為： $x = 0, y = 15$ 。

假設 4：李四到郵局買郵票，買了兩種郵票共 8 張，這兩種郵票分別為 3 元和 6 元，共花了 24 元。則輸入資料 a, b, c, d ，分別為 8, 3, 6, 24；

若 3 元郵票有 x 張，6 元郵票有 y 張，算出 x, y ：

$$3x + 6y = 24$$

則這組輸入資料之解為： $x = 8, y = 0$ 。

假設 5：王五到郵局買郵票，買了兩種郵票共 3 張，這兩種郵票分別為 1 元和 2 元，共花了 5 元。則輸入資料 a, b, c, d ，分別為 3, 1, 2, 5；

若 1 元郵票有 x 張，2 元郵票有 y 張，算出 x, y ：

$$x + 2y = 5$$

則這組輸入資料之解為： $x = 1, y = 2$ 。

輸入說明：

第 1 列的數字 n 代表有幾筆資料要測試， $1 \leq n \leq 5$ ，之後每列為每筆的測試資料，共有四個整數 a, b, c, d ， $1 \leq a, b, c, d \leq 300$ ，整數與整數之間用一個逗號隔開。

輸出說明：

每筆測試資料輸出一列，為一筆輸入資料之解，輸出資料為 x, y ，輸出的順序顛倒將不予計分。 x, y 為大於 0 或等於 0 的整數 ($0 \leq x, y$ and $x, y \in N$)。 x, y 皆為整數，不用考慮小數點後的有效位數，也不會有無法算出 x 及 y 值的測試資料。

輸入檔案 1 :【檔名：in1.txt】

3

10,5,20,110

12,3,10,92

15,5,10,150

輸入檔案 2 :【檔名：in2.txt】

2

8,3,6,24

3,1,2,5

輸出範例：【檔名：out.txt】

6,4

4,8

0,15

8,0

1,2

Problem 2：字串

子題 1：在兩字串中，共同出現的英文字母。(程式執行限制時間: 2 秒) **9 分**

(這題題目，輸入的兩字串都是英文小寫字母組成)

給定兩個英文字母組成的字串 `string_a` 與 `string_b`，請印出在兩字串中，皆出現的英文字母，出現的英文字母由 `a~z` 的順序印出，若同字母在兩字串中皆出現，如果出現次數不只一次，也只印出一次；換句話說按照英文字母由 `a~z` 的順序，找出在兩個字串 `string_a` 與 `string_b` 都出現過的英文字母。

例如在下列這組測試資料，在兩字串中，皆出現的英文字母加了底線說明，`n`、`o` 及 `w` 在兩字串中皆出現，所以這組測試資料輸出 `now`。

download

women

在下列這組測試資料中，`a` 及 `n` 在兩字串中皆出現，雖然 `a` 及 `n` 在二字串中出現次數不只一次，也只印出一次，所以這組測試資料輸出 `an`。

banana

naan

在下列這組測試資料中，沒有共同的英文字母，所以這組測試資料輸出大寫 `N`。

in

out

輸入說明：

第 1 列的數字 n 代表有幾組資料要測試， $1 \leq n \leq 5$ ，第二列起為測試資料，每兩列一組，每組的第一個字串為 `string_a`，第二個為 `string_b`。每個字串都是小寫的英文字母，其長度介於 1(含)~15(含)。之後每二列為每組的測試資料，即是要比較的兩字串。

輸出說明：

每組測試資料輸出一列。按照小寫英文字母 `a~z` 的順序，輸出兩字串中，皆出現的小寫英文字母；如果兩字串沒有共同的英文字母則輸出大寫 `N`。所以請選手注意，小寫的 `n` 代表在兩字串中皆出現的英文字母；大寫的 `N` 是兩字串中沒有共同的英文字母。**請選手注意，這題輸出大小寫意義有所不同。**

輸入檔案 1：【檔名：in1.txt】

3
download
women
banana
naan
then
street

輸入檔案 2：【檔名：in2.txt】

3
in
ten
google
yahoo
in
out

輸出範例：【檔名：out.txt】

now
an
et

n
o
N

子題 2：計算兩個人之間共同朋友的數量。(程式執行限制時間: 2 秒) 11 分

在社群網站中，每個人都可以跟其他人互相加為好友。兩個人之間可能會有一些共同的朋友，而共同朋友的數量越多，代表這兩個人的交友圈重疊性越高。請寫一支程式來完成這個功能，計算兩個人之間共同朋友的數量。

假設系統內部使用數字 ID 記錄好友，而不是使用名字或帳號，而 ID 的數字範圍為 1~65535。在輸入檔案中，每組輸入資料有兩列，分別代表兩位使用者的好友名單，每列第一個數字 k ， $1 \leq k \leq 20$ ，代表這個使用者有幾個好友，後面會接著 k 個”不同”的數字，代表他好友的 ID；數字與數字之間用一個或多個空白隔開，好友的 ID 是任意排列。請印出這兩個人之間有幾位共同朋友。

在下列這組資料中，兩個人之間皆出現的數字 ID 加了底線說明：

3 1 3 5

6 3 1 6 8 10 12

輸出 2

在下列這組資料中，兩個人之間皆出現的數字 ID 加了底線說明，但這組資料沒有相同數字 ID：

3 65535 19 3333

1 55555

輸出 0

在下列這組資料中，兩個人之間皆出現的數字 ID 加了底線說明：

8 12 21 26 29 32 777 567 65534

6 21 22 23 25 32 26

輸出 3

輸入說明：

第 1 列的數字 n 代表有幾組資料要測試， $1 \leq n \leq 5$ ，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。

輸出說明：

每組測試資料輸出一列。計算這兩位使用者之間有幾位共同朋友。

輸入檔案 1：【檔名：in1.txt】

```
2
3 1 3 5
6 3 1 6 8 10 12
3 6 5 5 3 5 19 3 3 3 3
1 5 5 5 5 5
```

輸入檔案 2：【檔名：in2.txt】

```
1
8 12 21 26 29 32 77 7 56 7 65 5 34
6 21 22 23 25 32 26
```

輸出範例：【檔名：out.txt】

```
2
0

3
```

Problem 3：其他

子題 1：撲克牌遊戲（程式執行限制時間: 2 秒） 14 分

許多人常喜歡玩撲克牌，每副撲克牌共有 52 張牌，有四種花色：黑桃、紅桃、方塊、和梅花。在撲克牌的玩法中，A 可作 1 點或 14 點，而 2-10 則等同於該牌之點數，另外 J、Q、K 分別為 11、12、13 點。在測試檔案中，每位玩家只會分到六張牌。下表將 52 張牌分別對應到數字 1~52，在測試檔案中，將以下表的數字代表某張牌。

| 點數 花色 | A | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | J | Q | K |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 黑桃 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 紅桃 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 方塊 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 梅花 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 |

五張牌的相關的牌型如下：

「同花順」為同花色且連續點數的五張牌，相同花色的「順子」，得分7分；

「四條」為四張同點數的牌，外加任一單張的五張牌，得分6分；

「葫蘆」為三張同點數，另兩張同點數的牌；即一個「一對」和「三條」所組成的五張牌；得分5分；

「順子」為五張點數連續的牌，點數各差 1 點的連續牌，從 A-2-3-4-5(1-2-3-4-5)，到 10-J-Q-K-A(有10-11-12-13-14，但沒有J-Q-K-A-2)，得分4分；

「三條」五張牌中包含三張同點數的牌，得分3分；

「兩對」五張牌中包含兩對兩同點數的牌，但不是四張相同點數的牌(非四條)，得分2分；

「一對」五張牌中包含只有兩張同點數的牌，得分1分；

「雜牌」指不屬於以上任何一種組合，得分 0 分。

如上表所示，每張牌以一個數字(1~52)代表，例如：以 18 代表紅桃 5。本題目判斷手上的六張牌中，任取五張找出得分最高的分數是屬於以上那一種牌型。

例如測試資料 3 44 4 6 7 5，分別對應到六組的五張牌、牌型和如下：

44 4 6 7 5：一對，1 分。

3 4 6 7 5：同花順，7 分。

3 44 6 7 5：一對，1 分。

3 44 4 7 5：一對，1 分。

3 44 4 6 5：一對，1 分。

3 44 4 6 7：順子，4，分。

得分最高的分數為 7 分，所以這筆測試資料輸出 7。

輸入說明：

每個輸入資料含多個玩家取得的撲克牌資料，在檔案 in1.txt 和 in2.txt 中，每個玩家分別各使用一副牌，第 1 列的數字 n 代表有幾筆玩家資料要測試， $1 \leq n \leq 5$ ，第二列起為測試資料，每列為每筆的測試資料，代表每個玩家拿到的六張撲克牌，每張牌以空白隔開，而空白不限定一個。

輸出說明：

按照每個玩家手上的六張牌，任取五張判斷屬於那一種牌型，以得分 0(含)~7(含)代替牌型，程式應判斷牌型 6 種牌型組合，輸出得分最高的分數。

輸入範例 1：【檔名：in1.txt】

```
2
3 44 4 6 7 5
19 12 1 32 45 25
```

輸入範例 2：【檔名：in2.txt】

```
2
14 16 18 19 20 21
4 17 30 43 9 22
```

輸出範例：【檔名：out.txt】

```
7
5

0
6
```

子題 2：費氏數列。(程式執行限制時間: 2 秒) 16 分

費氏數列 $\text{Fib}(i)$ 的前兩項為 0 與 1，之後的每一項次為其前兩項次相加的和。如下表所示：

| | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|----|----|-----|
| i 項次 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... |
| $\text{Fib}(i)$ 項次值 | 0 | 1 | 1 | 2 | 3 | 5 | 8 | 13 | 21 | ... |

在 10 進制轉 16 進制的數字系統中，例如 12_{10} 可轉成 C_{16} ，其中 C_{16} 是以 16 進制為"基底"；在 10 進制轉 2 進制的數字系統中，例如 12_{10} 可轉成 1100_2 ，其中 1100_2 是以 2 進制為"基底"。

所有正整數都可以用費氏數列中部份項次的和表示；換句話說，所有正整數都可以用費氏數列中取部份"不重複"的項次表示。例如，21 可以用不同的組合表示： $\{21\}$ 、 $\{13, 8\}$ 或 $\{13, 5, 3\}$ ，這些集合內的值都是費氏數列中的項次值，集合內的值加總都為 21；再例如 16 則可用 $\{13, 3\}$ 、 $\{8, 5, 3\}$ 、 $\{13, 2, 1\}$ 或 $\{8, 5, 2, 1\}$ 表示。所有正整數都可以用不重複的費氏數列為"基底"的項次表示。但可看出其表示式可能不只一種。為唯一表達，我們規定任兩個被選中的項次不能在費氏數列中相鄰；因為任兩個相鄰的費氏數列項次的和就等於下一個費氏數列項次；如此保證所有正整數都只有唯一的一組以費氏數列為"基底"表示式，費氏數列基底表示式從 $\text{Fib}(2)$ 開始計算。

我們以 16 為例說明， $16=13+3$ ，表示 16 要取二項： $\text{Fib}(7)$ 和 $\text{Fib}(4)$ ，而不取 $\text{Fib}(6)$ 、 $\text{Fib}(5)$ 、 $\text{Fib}(3)$ 、 $\text{Fib}(2)$ 的項次及其他項次。若以 1 代表採用某項次之值，0 代表不採用，則： $16=100100$ ，如下表所示：

| | | | | | | |
|-----------------|----------|---|---|---|---|---|
| i | 7 | 6 | 5 | 4 | 3 | 2 |
| $\text{Fib}(i)$ | 13 | 8 | 5 | 3 | 2 | 1 |
| $16=13+3$ | 1 | 0 | 0 | 1 | 0 | 0 |

我們以 8 為例來說明，表示 8 要取一項： $\text{Fib}(6)$ ，而不取 $\text{Fib}(5)$ 、 $\text{Fib}(4)$ 、 $\text{Fib}(3)$ 、 $\text{Fib}(2)$ 的項次及其他項次。所以 $8=10000$ ，如下表所示：

| | | | | | |
|-----------------|----------|---|---|---|---|
| i | 6 | 5 | 4 | 3 | 2 |
| $\text{Fib}(i)$ | 8 | 5 | 3 | 2 | 1 |
| 8 | 1 | 0 | 0 | 0 | 0 |

我們以 6 為例來說明，表示 6 要取一項： $\text{Fib}(5)$ 和 $\text{Fib}(2)$ ，而不取 $\text{Fib}(4)$ 、 $\text{Fib}(3)$ 的項次及其他項次。所以 $6=1001$ ，如下表所示：

| | | | | |
|-----------------|----------|---|---|---|
| i | 5 | 4 | 3 | 2 |
| $\text{Fib}(i)$ | 5 | 3 | 2 | 1 |
| $6=5+1$ | 1 | 0 | 0 | 1 |

輸入一個以十進位數，請寫出該值以費氏數列為基底的表示式，**最左邊的數字一定要為 1，不可為 0。**

輸入說明：(這題題目修正部份，加了底線來說明。)

第一列的數字 n 代表有幾組資料要測試， $1 \leq n \leq 5$ 。第二列起為測試資料，每列為每筆的測試資料，每一列為一正整數 x ，其值為 $1 \leq x < 10000$ 。

輸出說明：

計算每筆測試資料中，請以下列的格式輸出答案，等號右邊的長度 ≤ 17 。

提示：輸入一個以十進位數，請寫出該值以費式數列為基底的表示式，其表示式最大值的長度為 17，換句話說表示式最大值為

10101010101010101

輸入檔案 1：【檔名：in1.txt】

4
1
2
3
4

輸入檔案 2：【檔名：in2.txt】

4
5
6
8
16

輸出範例：【檔名：out.txt】

1=1
2=10
3=100
4=101

5=1000
6=1001
8=10000
16=100100

Problem 4：資料結構—樹、陣列

子題 1：凱撒密碼。(程式執行限制時間: 2 秒) 13 分

在 ASCII Code 中，每個字元需要使用 8 bit 來存資料。當檔案只包含 0123456789 十種字元時，可以重新編二進制碼以節省空間，假設“二進制的新編碼”如下：

| 二進制 | 字元 |
|--------|----|
| 00 | 0 |
| 01 | 1 |
| 100 | 2 |
| 101 | 3 |
| 1100 | 4 |
| 1101 | 5 |
| 11100 | 6 |
| 11101 | 7 |
| 111100 | 8 |
| 111101 | 9 |

凱撒密碼是廣為人知的代換密碼。為了用凱撒密碼法加密訊息，每個明文的字母將會被其位置的後 3 個字母替代。因此字母 A 將會被字母 D 替代、字母 B 將會被字母 E 替代、字母 C 將會被字母 F 替代，以此類推，最後，X、Y 和 Z 將分別被替代為 A、B 和 C。例如，“WIKIPEDIA”將被加密成“ZLNLSHGLD”。凱撒密碼把字母向後移“3”位。

明文字母表： A BCDEF GHIJK LMNOP QRSTU VWXYZ

密文字母表： D EFGHI JKLMN OPQRS TUVWX YZABC

我們以每個單字用二個數字表示“密文字母表”：

| | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 |
| D | E | F | G | H | I | J | K | L | M | N | O | P |

| | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| Q | R | S | T | U | V | W | X | Y | Z | A | B | C |

在輸入檔案中，密文表示的數字已用“二進制的新編碼”表示，00 100 對應到數字 02，“密文字母表”02 對應到英文字母 E。

”二進制的新編碼” 01 11100 對應到數字 16，”密文字母表”16 對應到英文字母 S。

”二進制的新編碼” 100 1101 對應到數字 25，”密文字母表”25 對應到英文字母 B。

”二進制的新編碼” 00 111101 對應到數字 09，”密文字母表”09 對應到英文字母 L。

”二進制的新編碼” 01 11101 對應到數字 17，”密文字母表”17 對應到英文字母 T。

(輸入檔案會省略空白，空白的存在是為了方便讀題)

寫一程式，把輸入的”二進制的新編碼”對應到的數字，每筆的測試資料對應到一個 2 位數碼(01、02、03、…、25、26)，再查”密文字母表”，輸出數碼所對應到的一個大寫英文字母。

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，為二進制的新編碼。**每列二進制的新編碼對應到包含 2 個數字的數碼**，值可能為 01、02、03、…、25、26 其中之一。

輸出說明：

每筆測試資料輸出一列。為數碼在密文字母表中所對應到的大寫英文字母，**請選手注意，這題輸出全部大寫，若輸出為小寫，不予計分。**

輸入檔案 1：【檔名：in1.txt】

```
3
00100
0111100
1001101
```

輸入檔案 2：【檔名：in2.txt】

```
2
00111101
0111101
```

輸出範例：【檔名：out.txt】

```
E
S
B

L
T
```

子題 2：列出所有樹的某節點到根節點之路徑長度。(程式執行限制時間: 2 秒) 15 分

在資料結構中，樹狀結構是可以用來描述有分支的結構，其包含 1 個或多個節點。每棵樹存在一個特殊的節點，稱為根節點(root)，可連結若干子樹，也可以沒有子樹；從任一節點到根節點，都只有唯一一條的節點不重複路徑。例如 F 到 A 的路徑為 $F \rightarrow B \rightarrow A$ ，其路徑長度為 2，此路徑 F 到 A，A 為此樹之根節點，路徑中間所經過的節點集合為 {B}。

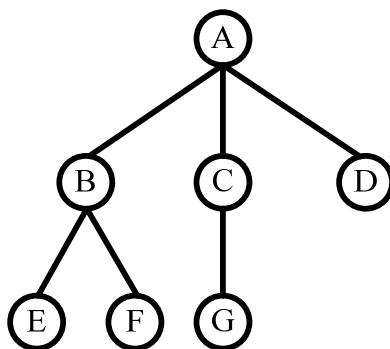


圖 4.2.1

在圖 4.2.1 中，有編號的圓形代表節點，A 為根節點，B、C 及 D 均為 A 的子節點，各節點之間不會有迴圈，且所有節點之間都有一個或多個邊相連通。任一樹狀結構的總邊數等於其總節點數減 1，在樹上任意添加一條邊，就會產生迴圈；在樹上任意刪除一條邊，一顆樹就裂成兩棵樹(森林)。沒有迴圈的圖，就是樹或森林。若為無根樹則是任一節點皆可為根節點。

寫一個程式，讀入多棵樹狀結構的資料，依每組測試資料算出某節點到根節點之路徑長度，需分別算出各棵樹的結果(各路徑長度為中間所經過的節點形成的集合之元素個數加 1，若中間所經過的節點集合為空集合則路徑長度為 1)。

輸入說明：

第一列的數字 n 代表共有幾組資料要測試， $2 \leq n \leq 5$ 。第二列起則是每一組測試資料。每組測試資料代表一個樹狀結構，每組測試資料中的第一列值為 m, k, v ， $3 \leq m \leq 256, 2 \leq k \leq 8, 0 \leq v \leq 255$ ； m 為節點的個數、 k 為樹的個數、 v 為某節點(v 不會是根節點)。之後的 m 列的內容為 k 棵樹 k 個邊的資料，每列的內容有 $k + 1$ 個值，第一個值為節點 i ；之後 k 個值依序代表節點 i 在第 0 棵樹到第 $k-1$ 棵樹中的父節點。同一列中，**每個父節點的資料以一個或多個空白隔開**。若父節點編號為 999，則 i 為這組測試資料的根節點。在測試資料中，所有樹的根節點編號均相同。且在測試資料中，內容依節點編號依序描述，即節點 i 的值為 $0, 1, 2, \dots, m-1$ 遞增。接下來為下一組的測試資料。

輸出說明：

計算每組測試資料中，分別算出各棵樹某個節點到根節點之路徑長度，輸出的數字和數字之間需以逗號隔開，每組測試資料輸出一列。

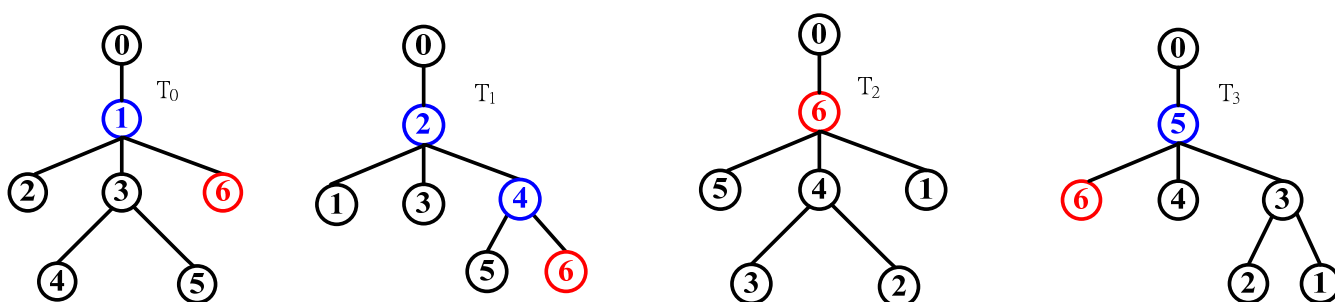
在 in1.txt 檔案中，第一組測試資料的第一列值 $m, k, v = 7, 4, 6$ ，代表共有 7 個節點、4 棵樹，程式應該算出節點 v ，即節點 6，到不同樹的根節點之路徑長度。

在 m 列資料中，第一列的資料：0 999 999 999 999。每列的內容為 $k + 1$ 個值，第一個值為節點 i ，之後 k 個值分別代表節點 i 在第 0~3 棵樹中的父節點。因為父節點為 999，則節點 $i=0$ 為這組測試資料的根節點，所以這組測試資料，根節點為 0。

在 m 列資料中，第二列的資料：1 0 2 6 3。第一個值 1，節點 i 為 1，之後 k 個值分別代表節點 i 在第 0~3 棵樹中的父節點，依序為 0 2 6 3。

在 m 列資料中，第三列的資料：2 1 0 4 3。第一個值 2 節點 i 為 2；之後 k 個值分別代表節點 i 在第 0~3 棵樹中的父節點，依序為 1 0 4 3。

在 m 列資料中，第四列的資料：3 1 2 4 5。第一個值 3 節點 i 為 3；之後 k 個值分別代表節點 i 在第 0~3 棵樹中的父節點，依序為 1 2 4 5。



在這 4 棵樹中，節點 6 到根節點 0 路徑，中間所經過的節點集合分為 $\{1\}$ 、 $\{4, 2\}$ 、 $\{\}$ 和 $\{5\}$ ，路徑長度為中間所經過的節點形成的集合之元素個數加 1；所以這組測試資料輸出為 2,3,1,2。

7,4,6

0 999 999 999 999

1 0 2 6 3

2 1 0 4 3

3 1 2 4 5

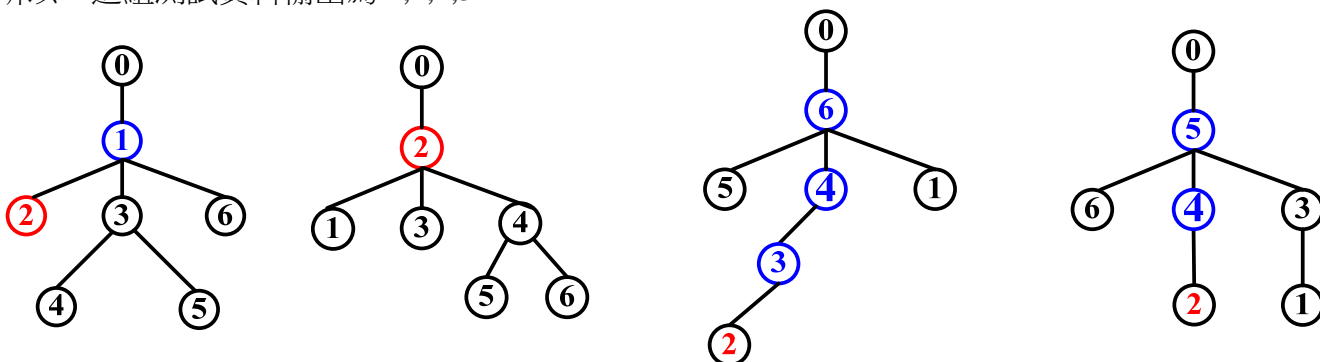
4 3 2 6 5

5 3 4 6 0

6 1 4 0 5

第二組測試資料的第一列值 $m, k, v = 7, 4, 2$ ，代表共有 7 個節點、4 棵樹，在這 4 棵樹中，分別算出節點 2 到根節點 0 的路徑長度。

在這 4 棵樹中，節點 2 到根節點 0 路徑，中間所經過的節點集合分為 $\{1\}$ 、 $\{\}$ 、 $\{3,4,6\}$ 和 $\{4,5\}$ ，所以，這組測試資料輸出為 2,1,4,3。



7,4,2

0 999 999 999 999

1 0 2 6 3

2 1 0 3 4

3 1 2 4 5

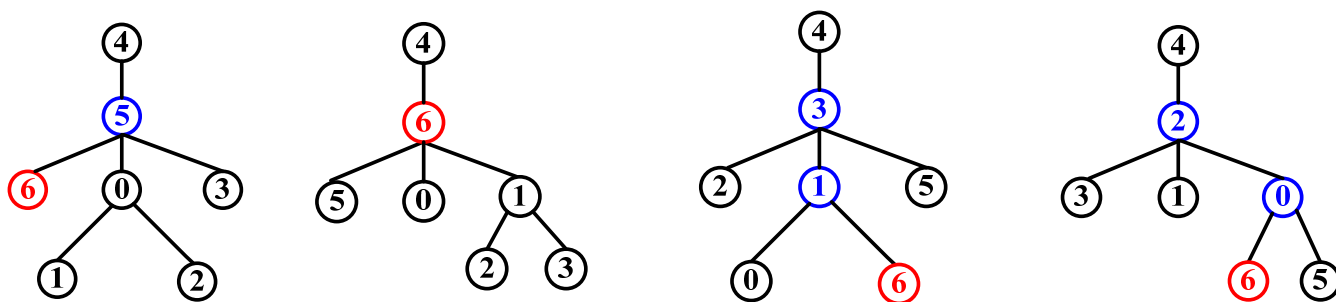
4 3 2 6 5

5 3 4 6 0

6 1 4 0 5

第三組測試資料的第一列值 $m, k, v = 7, 4, 6$ ，代表共有 7 個節點、4 棵樹，在這 4 棵樹中，分別算出節點 6 到根節點 4 的路徑長度。

在這 4 棵樹中，節點 6 到根節點 4 路徑，中間所經過的節點集合分為 $\{5\}$ 、 $\{\}$ 、 $\{1,3\}$ 和 $\{0,2\}$ ，所以，這組測試資料輸出為 2,1,3,3。



7,4,6

0 5 6 1 2

1 0 6 3 2

2 0 1 3 4

3 5 1 4 2

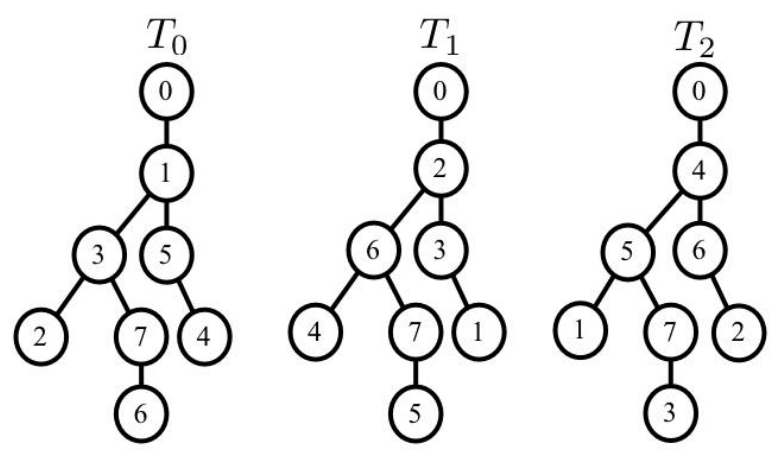
4 999 999 999 999

5 4 6 3 0

6 5 4 1 0

第四組測試資料的第一列值 $m, k, v = 8, 3, 1$ ，代表共有 8 個節點、3 棵樹，在這 3 棵樹中，分別算出節點 1 到根節點 0 的路徑長度。

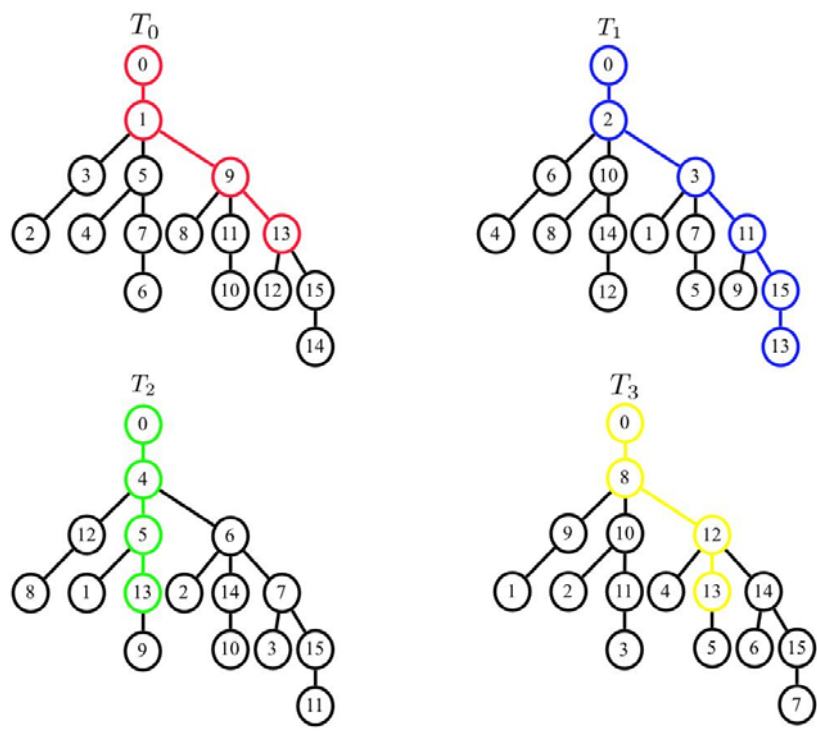
在這 3 棵樹中，節點 1 到根節點 0 路徑，中間所經過的節點集合分為{}、{3,2}和{5,4}，所以，這組測試資料輸出為 1,3,3。



8,3,1

| | | | |
|---|-----|-----|-----|
| 0 | 999 | 999 | 999 |
| 1 | 0 | 3 | 5 |
| 2 | 3 | 0 | 6 |
| 3 | 1 | 2 | 7 |
| 4 | 5 | 6 | 0 |
| 5 | 1 | 7 | 4 |
| 6 | 7 | 2 | 4 |
| 7 | 3 | 6 | 5 |

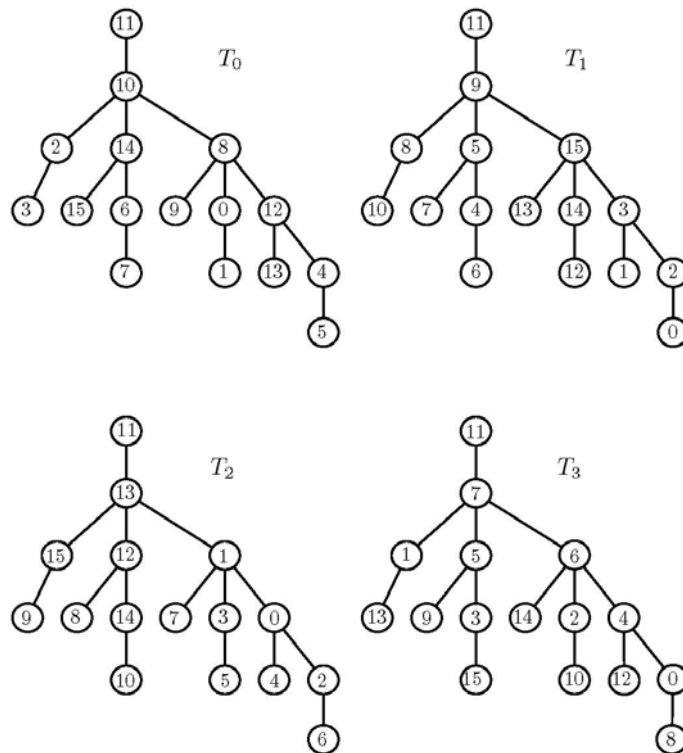
在 in2.txt 檔案中，第一組測試資料的第一列值 $m, k, v = 16, 4, 13$ ，代表共有 16 個節點、4 棵樹，在這 4 棵樹中，分別算出節點 13 到根節點 0 的路徑長度。
 在這 4 棵樹中，節點 13 到根節點 0 路徑，中間所經過的節點集合分為 $\{9, 1\}$ 、 $\{15, 11, 3, 2\}$ 、 $\{5, 4\}$ 和 $\{12, 8\}$ ，所以，這組測試資料輸出為 3,5,3,3。



| | | | | |
|---------|-----|-----|-----|-----|
| 16,4,13 | | | | |
| 0 | 999 | 999 | 999 | 999 |
| 1 | 0 | 3 | 5 | 9 |
| 2 | 3 | 0 | 6 | 10 |
| 3 | 1 | 2 | 7 | 11 |
| 4 | 5 | 6 | 0 | 12 |
| 5 | 1 | 7 | 4 | 13 |
| 6 | 7 | 2 | 4 | 14 |
| 7 | 5 | 3 | 6 | 15 |
| 8 | 9 | 10 | 12 | 0 |
| 9 | 1 | 11 | 13 | 8 |
| 10 | 11 | 2 | 14 | 8 |
| 11 | 9 | 3 | 15 | 10 |
| 12 | 13 | 14 | 4 | 8 |
| 13 | 9 | 15 | 5 | 12 |
| 14 | 15 | 10 | 6 | 12 |
| 15 | 13 | 11 | 7 | 14 |

第二組測試資料的第一列值 $m, k, v = 16, 4, 15$ ，代表共有 16 個節點、4 棵樹，在這 4 棵樹中，分別算出節點 15 到根節點 11 的路徑長度。

在這 4 棵樹中，節點 15 到根節點 11 路徑，中間所經過的節點集合分為 $\{14, 10\}$ 、 $\{9\}$ 、 $\{13\}$ 和 $\{3, 5, 7\}$ ，所以，這組測試資料輸出為 3, 2, 2, 4。



16, 4, 15

| | | | | |
|----|-----|-----|-----|-----|
| 0 | 8 | 2 | 1 | 4 |
| 1 | 0 | 3 | 13 | 7 |
| 2 | 10 | 3 | 0 | 6 |
| 3 | 2 | 15 | 1 | 5 |
| 4 | 12 | 5 | 0 | 6 |
| 5 | 4 | 9 | 3 | 7 |
| 6 | 14 | 4 | 2 | 7 |
| 7 | 6 | 5 | 1 | 11 |
| 8 | 10 | 9 | 12 | 0 |
| 9 | 8 | 11 | 15 | 5 |
| 10 | 11 | 8 | 14 | 2 |
| 11 | 999 | 999 | 999 | 999 |
| 12 | 8 | 14 | 13 | 4 |
| 13 | 12 | 15 | 11 | 1 |
| 14 | 10 | 15 | 12 | 6 |
| 15 | 14 | 9 | 13 | 3 |

輸入檔案 1 :【檔名 : in1.txt】

4

7,4,6

0 999 999 999 999

1 0 2 6 3

2 1 0 4 3

3 1 2 4 5

4 3 2 6 5

5 3 4 6 0

6 1 4 0 5

7,4,2

0 999 999 999 999

1 0 2 6 3

2 1 0 3 4

3 1 2 4 5

4 3 2 6 5

5 3 4 6 0

6 1 4 0 5

7,4,6

0 5 6 1 2

1 0 6 3 2

2 0 1 3 4

3 5 1 4 2

4 999 999 999 999

5 4 6 3 0

6 5 4 1 0

8,3,1

0 999 999 999

1 0 3 5

2 3 0 6

3 1 2 7

4 5 6 0

5 1 7 4

6 7 2 4

7 3 6 5

輸入檔案 2 :【檔名 : in2.txt】

2

16,4,13

0 999 999 999 999

1 0 3 5 9

2 3 0 6 10

3 1 2 7 11

4 5 6 0 12

5 1 7 4 13

6 7 2 4 14

7 5 3 6 15

8 9 10 12 0

9 1 11 13 8

10 11 2 14 8

11 9 3 15 10

12 13 14 4 8

13 9 15 5 12

14 15 10 6 12

15 13 11 7 14

16,4,15

0 8 2 1 4

1 0 3 13 7

2 10 3 0 6

3 2 15 1 5

4 12 5 0 6

5 4 9 3 7

6 14 4 2 7

7 6 5 1 11

8 10 9 12 0

9 8 11 15 5

10 11 8 14 2

11 999 999 999 999

12 8 14 13 4

13 12 15 11 1

14 10 15 12 6

15 14 9 13 3

輸出範例：【檔名：out.txt】

2,3,1,2

2,1,4,3

2,1,3,3

1,3,3

3,5,3,3

3,2,2,4