

Problem 4：資料結構—樹、後序法

子題 1：樹。(程式執行限制時間: 2 秒) 16 分

在資料結構中，樹狀結構是可以用來描述有分支的結構，包含 1 個或多個節點。其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，都只有唯一的節點不重複路徑。

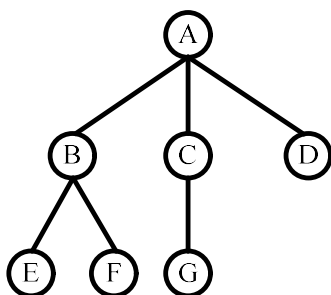


圖 4.1.1

在圖 4.1.1 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。

專有名詞介紹：

- (1) 無父節點的節點為根節點(Root)，如 A。
- (2) 父節點 (Parent)：一節點的上層節點為父節點，如 B 的父節點為 A，如 G 的父節點為 C。
- (3) 子節點 (Children)：一節點的下層節點為子節點，如 B 的子節點有 E 及 F；C 的子節點有 G。
- (4) 兄弟節點 (Siblings)：有共同父節點的節點稱為兄弟節點，如 B、C、D 互為兄弟節點。
- (5) 分支度 (Degree)：一個節點的子樹個數稱為其分支度，如 A 的分支度為 3；B 的分支度為 2；C 的分支度為 1；E 的分支度為 0。
- (6) 樹葉節點(Terminal node)：無子節點的節點，如 D、E、F、G。
- (7) 內部節點 (Non-terminal node)：樹葉以外的節點均為內部節點，如 A、B、C。
- (8) 階層或階度 (Level)：A 為階層 1；B、C、D 為階層 2；E、F、G 為階層 3。
- (9) 高度 (Height)：樹的最大階度，例如圖 4.1.1，因最大階度階度為 3，則其樹的高度為 3。

寫一個程式，讀入一無向圖的資料。在測試檔中，節點的編號不一定是連續的號碼。

輸入說明：

第一列的數字 n 代表共有幾組資料要測試， $2 \leq n \leq 20$ 。

第二列起每一行代表一組測試資料。每組測試資料代表一圖形，內容為邊的資料。每個邊以 2 個整數 i, j 表示， $0 \leq i, j \leq 20$ and $i \neq j$ ，其中 i 和 j 為節點的編號，代表從 i 節點和 j 節點有邊相連，每組測試資料，同一列中，每個邊的資料以一個或多個空白隔開， $|i, j|$ 為邊的個數， $2 \leq |i, j| \leq 20$ 。

輸出說明：

每組測試資料輸出一列。測試資料的每個邊依序加入圖形。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。如果檢測的圖形有迴圈(最多一個迴圈)，則輸出造成迴圈的所有點(由小到大依序輸出)；若該圖是樹，則輸出 T；沒有迴圈又不是樹，則輸出 F。

輸入檔案 1：【檔名：in1.txt】

4

5,8 5,3 5,2 5,4 5,6 1,2 2,0

8,1 1,3 6,2 8,10 7,5 1,4 7,8 8,6 8,0

3,8 6,8 6,4 0,6 8,2 2,0 5,3

1,0 4,3 1,2

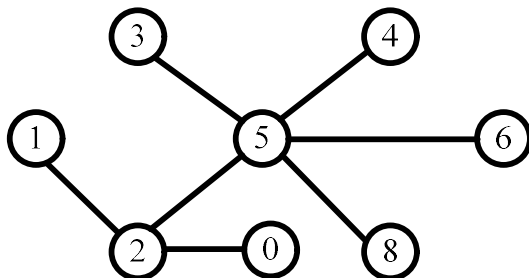


圖 4.1.1.1 (in1.txt)

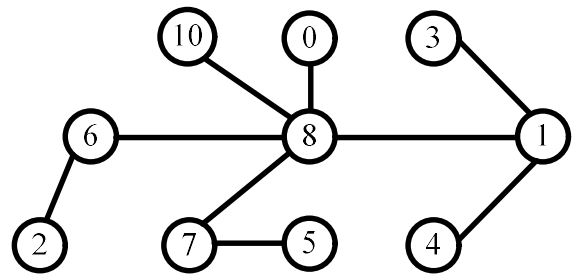


圖 4.1.1.2 (in1.txt)

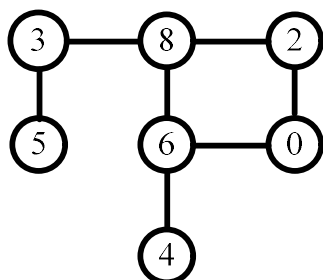


圖 4.1.1.3 (in1.txt)

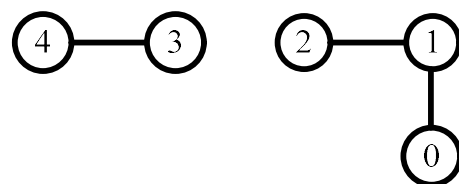


圖 4.1.1.4 (in1.txt)

輸入檔案 2 :【檔名：in2.txt】

5
1,2 2,3 4,0
4,3 2,3 2,1 1,0
1,2 2,3 3,4 1,4 1,5
1,2 1,3 2,3
1,2 2,3 3,4 1,5 5,4

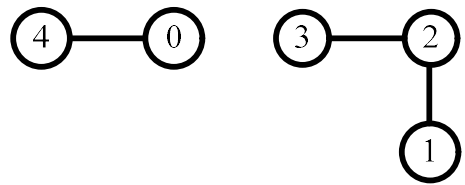


圖 4.1.2.1 (in2.txt)

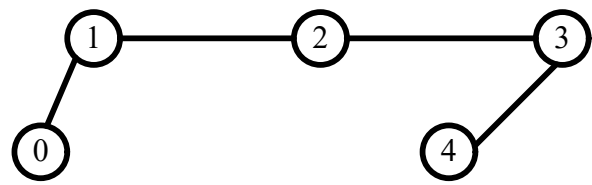


圖 4.1.2.2 (in2.txt)

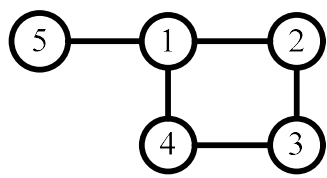


圖 4.1.2.3 (in2.txt)

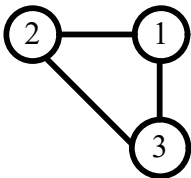


圖 4.1.2.4 (in2.txt)

輸出範例：【檔名：out1.txt】

T
T
0, 2, 6, 8
F

輸出範例：【檔名：out2.txt】

F
T
1, 2, 3, 4
1, 2, 3
1, 2, 3, 4, 5