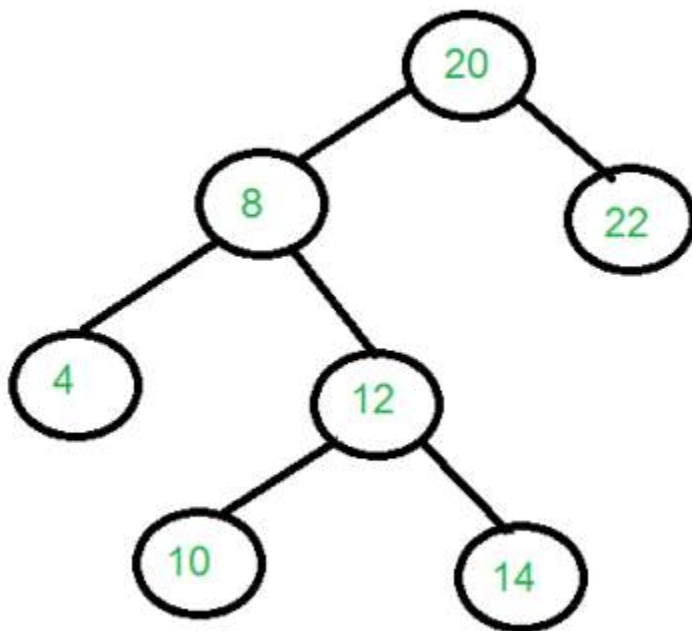


## CONTEST 6: BINARY TREE

DUYỆT CÂY THEO MỨC .....	1
CÂY NHỊ PHÂN HOÀN HẢO .....	3
CÂY NHỊ PHÂN TÌM KIẾM.....	4
DUYỆT SAU CÂY TÌM KIẾM CÂN BẰNG.....	6
ĐẾM SỐ NODE LÁ CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG.....	7
ĐỘ CAO CỦA CÂY.....	8
ĐƯỜNG ĐI TỚI NODE LÁ.....	9
DUYỆT CÂY THEO MỨC ĐẢO NGƯỢC.....	11

### **DUYỆT CÂY THEO MỨC**

Cho cây nhị phân, nhiệm vụ của bạn là duyệt cây theo Level-order. Phép duyệt level-order trên cây là phép thăm node theo từng mức của cây. Ví dụ với cây dưới đây sẽ cho ta kết quả của phép duyệt level-order: 20 8 22 4 12 10 14.



#### **Input:**

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

#### **Output:**

- Đưa ra kết quả phép duyệt level-order theo từng dòng.

#### **Ví dụ:**

Input	Output
2	
2	1 3 2
1 2 R 1 3 L	10 20 30 40 60
4	
10 20 L 10 30 R 20 40 L 20 60 R	

### **CÂY NHỊ PHÂN HOÀN HẢO**

Cho cây nhị phân, nhiệm vụ của bạn là kiểm tra xem cây nhị phân có phải là một cây hoàn hảo hay không (perfect tree)? Một cây nhị phân được gọi là cây hoàn hảo nếu tất cả các node trung gian của nó đều có hai node con và tất cả các node lá đều có cùng một mức.

#### **Input:**

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

#### **Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

#### **Ví dụ:**

Input	Output
3	
6	
10 20 L 10 30 R 20 40 L 20 50 R 30 60 L 30 70 R	Yes
2	Yes
18 15 L 18 30 R	No
5	
1 2 L 2 4 R 1 3 R 3 5 L 3 6 R	

## **CÂY NHỊ PHÂN TÌM KIẾM**

Hãy cài đặt cấu trúc cây nhị phân tìm kiếm (không cần đảm bảo tính chất cân bằng) với các node trên cây chứa các số nguyên dương không quá 1000 và thực hiện các loại truy vấn sau:

1. Dạng 1: Thêm một node vào cây
2. Dạng 2: Xóa một node trên cây và đưa node trái nhất của cây con phải lên thay thế
3. Dạng 3: Duyệt cây theo thứ tự trước và in ra
4. Dạng 4: Duyệt cây theo thứ tự giữa và in ra
5. Dạng 5: Duyệt cây theo thứ tự sau và in ra

### **Input:**

- Gồm nhiều dạng truy vấn, luôn bắt đầu bằng dạng 1, mỗi truy vấn trên một dòng. Mỗi dòng sẽ ghi số thứ tự của dạng truy vấn.
- Truy vấn dạng 1 và 2 sẽ kèm theo một số nguyên dương không quá 1000 là số cần thêm hoặc loại ra khỏi cây (truy vấn dạng 1 được đảm bảo giá trị không trùng với các node đã có, truy vấn dạng 2 luôn đảm bảo chắc chắn đã có node cần loại ở trong cây).
- Cây được đảm bảo luôn có không quá 1000 node.

*Chú ý: Cần đọc dữ liệu đến hết luồng vào, không biết trước số dòng.*

**Output:** Ghi ra thứ tự duyệt khi gặp truy vấn 3,4,5.

**Ví dụ:**

Input	Output
1 30	30 16 10 4 17 57 40
1 15	4 10 16 17 30 40 57
1 50	4 10 17 16 40 57 30
1 10	
1 17	
1 40	
1 57	
1 68	
1 4	
1 16	
2 15	
2 50	
2 68	
3	
4	
5	

### DUYỆT SAU CÂY TÌM KIẾM CÂN BẰNG

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số  $A[] = \{a_0, a_1 \dots a_{n-1}\}$ .  
Đưa ra phép duyệt theo thứ tự sau (post-order) của cây tìm kiếm cân bằng. Ví dụ  
với dãy  $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$  ta sẽ có phép duyệt theo thứ  
tự sau của cây nhị phân tìm kiếm cân bằng với node gốc là 30: 20 15 28 25 23 35 33  
45 40 38 30.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 106$ ). Dòng tiếp theo là  $N$  số của mảng  $A[]$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output
2	20 15 28 25 23 35 33 45 40 38 30
11	1 4 3 2 7 6 10 9 8 5
40 28 45 38 33 15 25 20 23 35 30	
10	
1 2 3 4 5 6 7 8 9 10	

### **ĐẾM SỐ NODE LÁ CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG**

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số  $A[] = \{a_0, a_1 \dots a_{n-1}\}$ . Đếm số node lá của cây nhị phân tìm kiếm cân bằng. Ví dụ với dãy  $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$  ta sẽ có phép duyệt theo thứ tự trước của cây nhị phân tìm kiếm cân bằng với số node lá là 4: 20, 28, 35, 45.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 1000$ ). Dòng tiếp theo là  $N$  số của mảng  $A[]$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output
2	4
11	4
40 28 45 38 33 15 25 20 23 35 30	
10	
1 2 3 4 5 6 7 8 9 10	



### **ĐỘ CAO CỦA CÂY**

Cho một cây có N node, gốc tại 1. Độ cao của một node u được tính bằng khoảng cách từ u tới node gốc. Độ cao của node gốc bằng 0. Nhiệm vụ của bạn là hãy tìm node xa node gốc nhất và in ra độ cao của node này.

#### **Input:**

Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).

Mỗi test bắt đầu bằng số nguyên N ( $1 \leq N \leq 100\,000$ ).

N-1 dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v cho biết node u nối với node v.

#### **Output:**

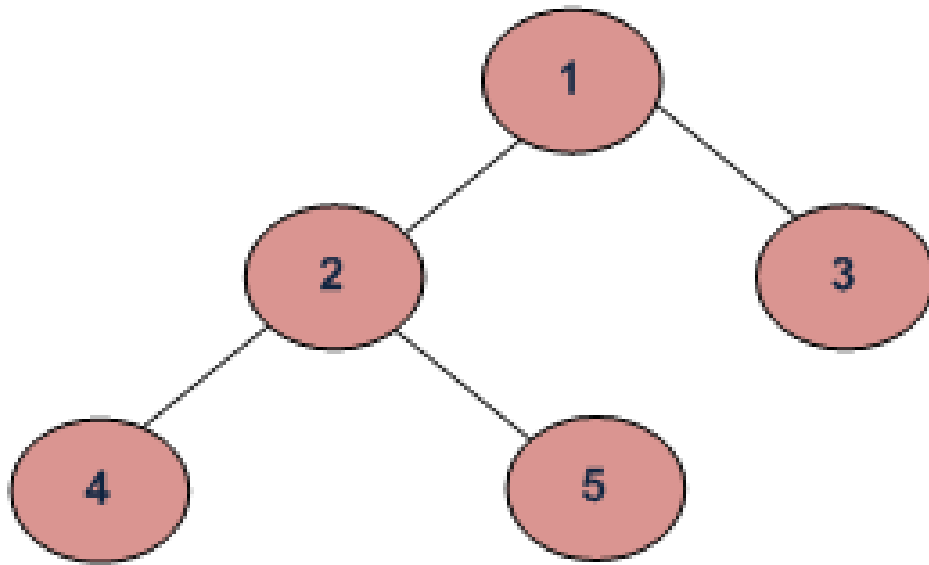
Với mỗi test, in ra một số nguyên là độ cao của cây.

#### **Test ví dụ:**

Input:	Output
2	2
5	3
1 2	
1 3	
2 4	
2 5	
4	
1 2	
2 3	
3 4	

### ĐƯỜNG ĐI TỚI NODE LÁ

Cho một cây có N node, gốc tại 1. Với mỗi node lá, bạn hãy in ra đường đi từ node gốc tới nó.



#### **Input:**

Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).

Mỗi test bắt đầu bằng số nguyên N ( $1 \leq N \leq 1000$ ).

N-1 dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v cho biết node u nối với node v.

#### **Output:**

Với mỗi test, in ra K dòng, trong đó K là số lượng node lá. Mỗi dòng là đường đi từ node gốc tới node lá X. Node lá nào có nhãn nhỏ hơn, in ra trước.

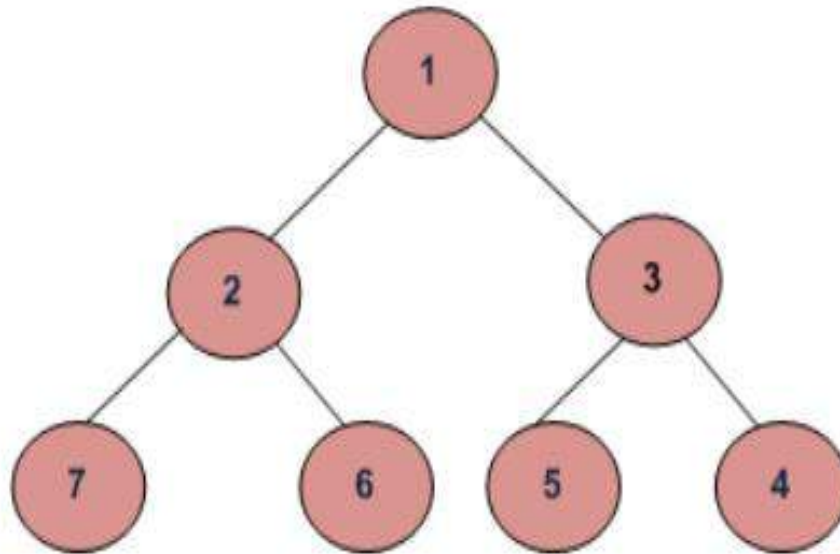
**Ví dụ:**

Input	Output
2	1 3
5	1 2 4
1 2	1 2 5
1 3	1 2 3 4
2 4	
2 5	
4	
1 2	
2 3	
3 4	

### DUYỆT CÂY THEO MỨC ĐẢO NGƯỢC

Cho cây nhị phân, nhiệm vụ của bạn là duyệt cây theo mức đảo ngược (reverse-level-order).

Với cây dưới đây sẽ cho ta kết quả của phép duyệt theo mức đảo ngược là: 7 6 5 4 3 2 1.



#### **Input:**

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

#### **Output:**

- Đưa ra kết quả phép duyệt reverse-level-order theo từng dòng.

#### **Ví dụ:**

Input	Output
2	
2	
1 2 R 1 3 L	3 2 1
4	40 60 20 30 10
10 20 L 10 30 R 20 40 L 20 60 R	