

# Daun Kiri Daun Kanan

Budi mempunyai pohon spesial yang bernama Budipoh. Pohon itu terhubung dengan program BudiProg di komputernya sehingga dapat membuat pohon itu bersinar. Untuk membuar pohon bersinar maka dia harus menyalakan daun kiri pohon kemudian daun kanan pohon. Untuk menyalakannya, Budi membutuhkan data-data cabang kiri dan cabang kanan secara postorder dari data binary tree yang diinputkan.

Karena sekarang komputer Budi sedang diperbaiki, maka ia meminta bantuanmu sebagai programmer handal untuk membuat program itu, sehingga Budi tidak memilah-milah data secara manual lagi.

## Input Format

- Baris pertama adalah N banyaknya query
- Untuk setiap N query:
- isi x, untuk  $x > 0$ , maka x akan dimasukkan ke dalam data binary tree
- tampil data, untuk menampilkan hasil pilahan

## Constraints

$$1 \leq N \leq 100$$

$$-100000 \leq x \leq 100000$$

## Output Format

Untuk setiap query "tampil data",

- baris pertama outputkan data cabang kiri secara *postorder*
- baris kedua outputkan data root
- baris ketiga outputkan data cabang kanan secara *postorder*
- Jika query "tampil data" lebih dari satu, maka diberikan pemisah '='

Ctt: Hidden output ada di sample output 0,1,2

## Sample Input 0

```
10
isi 10
isi 12
isi 14
tampil data
isi 8
isi 11
isi 20
isi 25
isi 6
tampil data
```

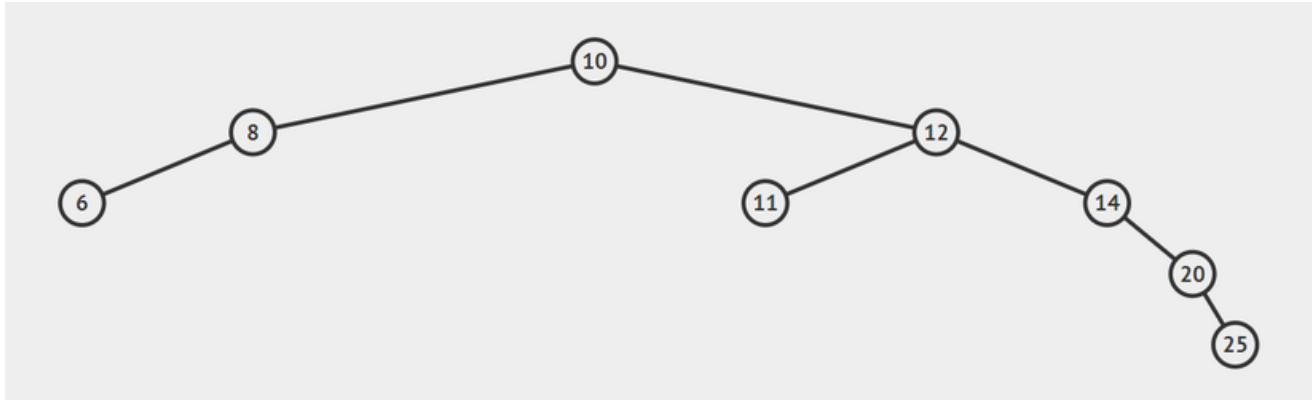
## Sample Output 0

```

    left : Kok gk tumbuh yah :<
root: 10
    right: 14 12
=====
    left : 6 8
root: 10
    right: 11 25 20 14 12

```

## Explanation 0



### Sample Input 1

```

3
isi -10
isi -200
tampil data

```

### Sample Output 1

```

Isi dulu dong @_@

```

### Sample Input 2

```

7
isi -50
isi 40
tampil data
isi 20
tampil data
isi 100
tampil data

```

### Sample Output 2

```

Belum tumbuh nih 40nya :(
=====
    left : 20
root: 40
    right: Kok gk tumbuh yah :<
=====
    left : 20
root: 40
    right: 100

```

