

1. Explain the differences between linear and non-linear data structures!

Ans :

Data structure linear merupakan data structure yang objectnya digambarkan dalam satu garis lurus.

Data structure non-linear merupakan data structure yang objectnya tidak digambarkan dalam satu garis lurus.

2. Describe the following terminology in a tree: base root, key, edge, siblings, parent, child, and leaf!

Ans :

Base root = node yang berada di paling atas sebuah tree

Key = value yang dimiliki oleh suatu node

Edge = garis yang menghubungkan node parent dengan node childnya.

Siblings = node-node yang memiliki parents yang sama

Parent = suatu node yang memiliki satu atau lebih anak

Child = anak node yang diturunkan oleh node lain.

Leaf = node yang tidak memiliki anak

3. Explain the following types of binary trees: full, complete, and perfect!

Ans :

- Full Binary Tree  
Merupakan suatu binary tree yang setiap nodenya harus memiliki 2 anak atau sama sekali tidak ada anak.
- Complete Binary Tree  
Merupakan suatu binary tree di mana setiap level tree, terkecuali level paling bawahnya, harus mempunyai node sebanyak jumlah maksimumnya,
- Perfect Binary Tree  
Merupakan suatu binary tree di mana setiap nodenya HARUS memiliki 2 anak dan semua leafnya berada di level yang sama.

4. What makes a tree balanced?

Ans :

Balance factor =  $|\text{height}(\text{root} \rightarrow \text{left}) - (\text{root} \rightarrow \text{right})| \leq 1$

Sebuah tree dikatakan balanced apabila tree tersebut memiliki height  $O(\log N)$ , dimana  $N$  merupakan jumlah node dalam tree.

Selain itu, kita dapat mengetahui sebuah tree termasuk balanced atau tidak dengan cara melihat apakah selisih height dari subtree kiri dengan height dari subtree kanannya  $\leq 1$ .

5. Explain the four properties of a binary tree!

Ans :

Untuk menghitung jumlah maksimum node pada level  $k \rightarrow 2^k$

Untuk menghitung jumlah maksimum node untuk suatu binary tree dengan height  $H \rightarrow 2^{H+1}-1$

Untuk mendapatkan minimum height suatu tree dengan node sebanyak  $N \rightarrow \lceil \log_2(N) \rceil$

Untuk mendapatkan maximum height suatu tree dengan node sebanyak  $N \rightarrow N-1$

Min level  $\rightarrow$  height yang bisa dibuat node-node agar bisa menjadi compact(sependek mungkin)

Max level  $\rightarrow$  height yg bisa dibuat node2 agar bisa sepanjang mungkin

6. Explain the intuition of implementing a binary tree using an array!

Ans :

Note :  $p$  merupakan index node yang sedang dicek(parent).

Root selalu berada di index 0.

Left child berada di index  $2p+1$

Right child berada di index  $2p+2$

Parent berada di index  $(p-1)/2$

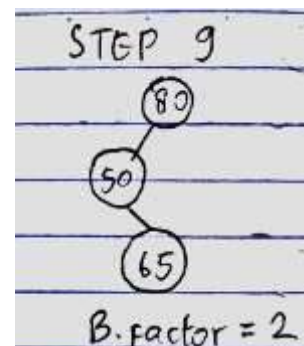
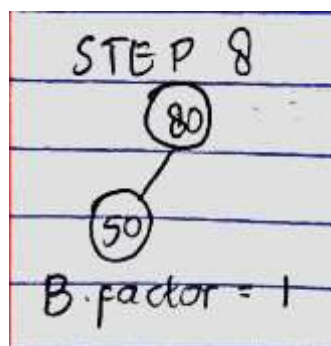
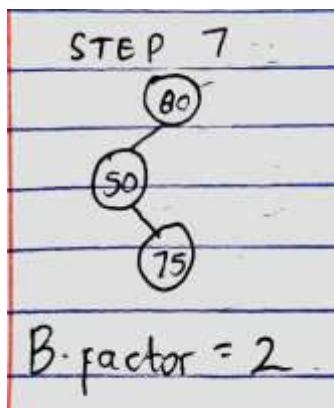
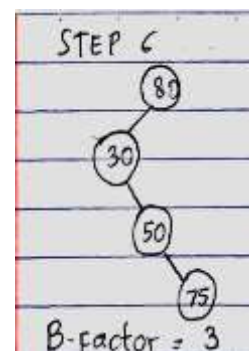
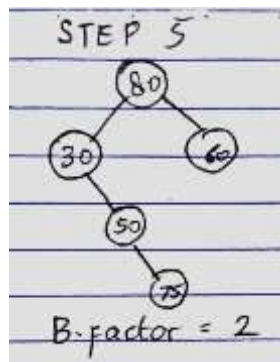
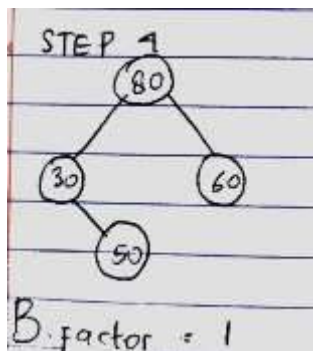
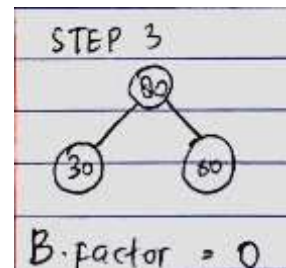
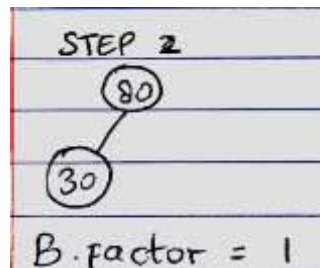
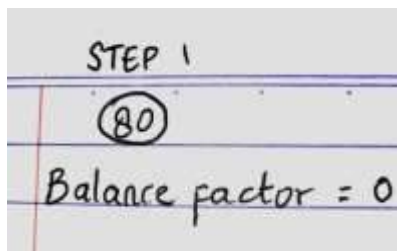
7. Explain the differences between inorder successor and inorder predecessor!

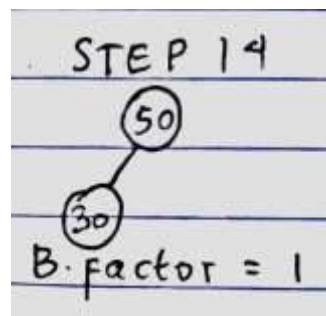
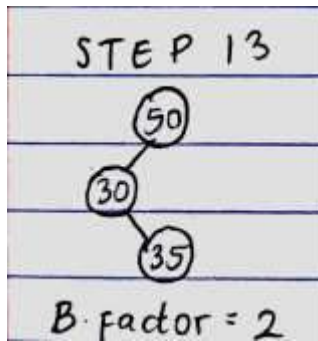
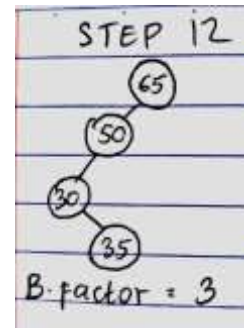
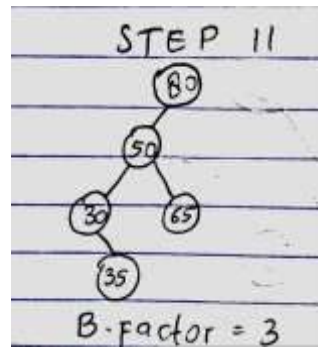
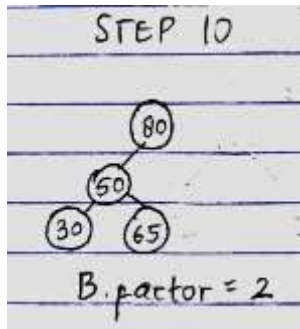
Ans :

Inorder successor merupakan node pengganti yang berada di satu posisi setelah elemen root saat node-node dalam tree disusun secara inorder(sorted).

Inorder predecessor merupakan node pengganti yang berada di posisi satu posisi di belakang elemen root saat node-node dalam tree disusun secara inorder(sorted).

8. Draw the following binary search tree step by step (14 pictures): - Insert 80, 30, 60, 50, 75 - Delete 60, 30, 75 - Insert 65, 30, 35 - Delete 80, 65, 35





Notes for number 8: for each picture, please write the balance factor of each node. Balance factor is defined as Balance Factor =  $|\text{height of left subtree} - \text{height of right subtree}|$