

Pertanyaan

2.1.2 Pertanyaan Percobaan

1. Mengapa dalam binary search tree proses pencarian data bisa lebih efektif dilakukan dibanding binary tree biasa?

Jawab :

Binary search tree proses pencarian data bisa lebih efektif dilakukan dibanding binary tree biasa karena dalam pencariannya binary search tree terdapat left-child itu harus lebih kecil dari pada right-child dan parent-nya. Karena hal tersebut dapat mempermudah dalam pencarian.

2. Untuk apakah di class **Node**, kegunaan dari atribut **left** dan **right**?

Jawab :

Pada class **Node** terdapat atribut left dan right, jika atribut left digunakan untuk menyimpan indeks yang mengarah ke arah left-child (nilai yang lebih kecil dari pada root). Sedangkan atribut right digunakan untuk menyimpan indeks yang mengarah ke arah right-child (nilai yang lebih besar dari pada root).

3. a. Untuk apakah kegunaan dari atribut **root** di dalam class **BinaryTree**?

Jawab :

Untuk menyimpan data yang pertama kali masuk.

b. Ketika objek tree pertama kali dibuat, apakah nilai dari **root**?

Jawab :

nilai dari **root** ketika pertama kali dibuat ialah **null**

4. Ketika tree masih kosong, dan akan ditambahkan sebuah node baru, proses apa yang akan terjadi?

Jawab :

Proses yang akan terjadi ialah sebuah penambahan data atau memasukkan data (add).

5. Perhatikan method **add()**, di dalamnya terdapat baris program seperti di bawah ini.

Jelaskan secara detil untuk apa baris program tersebut?

```
if(data < current.data){  
    if(current.left != null){  
        current = current.left;  
    }else{  
        current.left = new Node(data);  
        break;  
    }  
}
```

Jawab :

Jika nilai dari data lebih kecil dari data terakhir (yang sudah ada) dan Jika current.left tidak bernilai null (ada isinya). Maka, nilai dari current adalah current sebelumnya (current.left)/current yang ada di sebelah kiri, bisa dikatakan juga bahwasanya current.left akan menjadi parent dari data yang baru dimasukkan. Sedangkan, jika current.left bernilai null, maka current.left akan menyimpan data node baru/data yang baru dimasukkan akan menjadi current.left atau left-child.

13.2.1 Pertanyaan Percobaan**

1. Apakah kegunaan dari atribut data dan idxLast yang ada di class **BinaryTreeArray**?

Jawab :

Atribut **data** digunakan untuk menyimpan data array. Sedangkan atribut **idxLast** digunakan untuk menyimpan data batas index.

2. Apakah kegunaan dari method **populateData()**?

Jawab :

method yang berguna untuk menginisialisasi atribut data dan idxLast.

3. Apakah kegunaan dari method **traverseInOrder()**?

Jawab :

Digunakan untuk mencetak semua data yang ada di tree yang mana prosesnya ialah mengunjungi dan cetak seluruh data pada subtree sebelah kiri -> mengunjungi dan cetak data pada root -> mengunjungi dan cetak seluruh data pada subtree sebelah kanan.

4. Jika suatu node binary tree disimpan dalam array indeks 2, maka di indeks berapakah posisi left child dan right child masing-masing?

Jawab :

Posisi left-child berada pada indeks ke-5 dan right-child berada pada indeks ke-6

5. Apa kegunaan statement `int idxLast = 6` pada praktikum 2 percobaan nomor 4?

Jawab :

Untuk menunjukkan bahwa batas indeks array adalah 6.