Pertemuan 14

Waktu : 135 menit

Tujuan Pembelajaran : Mahasiswa mampu menjelaskan teknik pemrograman menggunakan Tree.

Substansi Materi : Binary Search Tree, AVL Tree

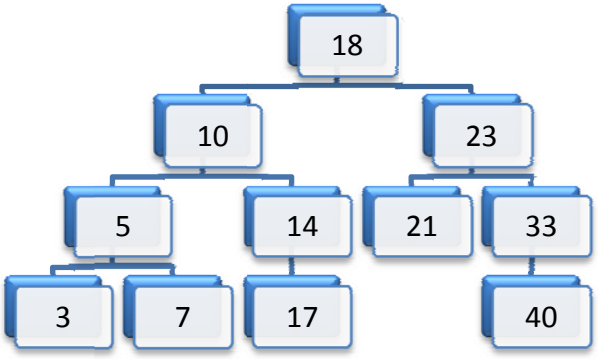
Tabulasi Kegiatan Perkuliahan

No	Tahap Kegiatan	Kegiatan Pengajar	Kegiatan Mahasiswa	Media & Alat	Waktu
1	Pendahuluan	1. Membuka pertemuan 2. Mengulang materi pertemuan sebelumnya	Menyimak Bertanya	Papan Tulis	20 Menit
2	Penyajian Materi	1. Binary Search Tree 2. AVLTree	Menyimak Bertanya Menjawab Pertanyaan	Papan Tulis	80 Menit
3	Penutup	1. Menyimpulkan materi pertemuan 2. Memberikan tugas kecil 3. Menutup pertemuan	Menyimak	Papan tulis	35 Menit

MATERI KULIAH

Binary Search Tree

Binary Search Tree adalah Binary Tree dengan sifat bahwa semua left child harus lebih kecil daripada right child dan parentnya. Juga semua right child harus lebih besar dari left child serta parentnya. Binary search tree dibuat untuk mengatasi kelemahan pada binary tree biasa, yaitu kesulitan dalam melakukan searching / pencarian node tertentu dalam binary tree. Contoh binary search tree umum adalah:



Gambar 2. Binary Search Tree secara umum

Pada dasarnya operasi dalam Binary Search Tree adalah sama dengan Binary Tree biasa, kecuali pada operasi insert, update dan delete.

- Insert : Pada Binary Search Tree, insert dilakukan setelah ditemukan lokasi yang tepat.
- Update : Seperti pada Binary Tree biasa, namun jika update berpengaruh pada posisi node tersebut, sehingga menyebabkan Tree bukan Binary Search Tree lagi, maka harus dilakukan perubahan pada tree dengan cara melakukan rotasi supaya tetap menjadi Binary Search Tree kembali.
- Delete : Seperti halnya update, delete dalam binary search tree juga turut mempengaruhi struktur dari tree tersebut.

AVLTree

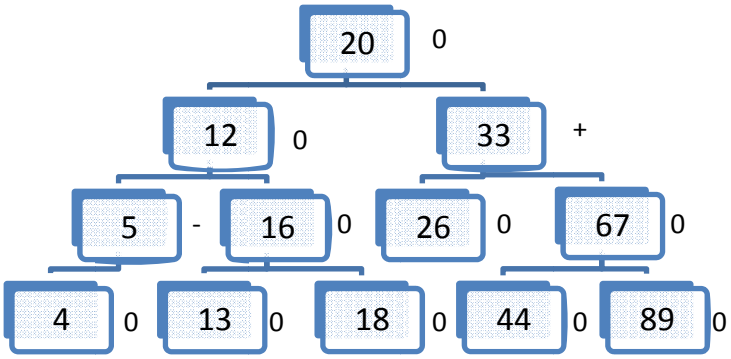
Adalah binary search tree yang memiliki perbedaan tinggi /level antara subtree kiri dan subtree kanan maksimal adalah 1. Avl tree muncul untuk menyeimbangkan binary search tree. Dengan avl tree waktu pencarian dan bentuk tree dapat dipersingkat dan disederhanakan

Selain avl tree terdapat pula height balanced n tree , yakni binary search tree yang memiliki perbedaan level antara subtree kiri dan subtree kanan maksimal adalah n sehinga avl tree adalah height balanced 1 tree

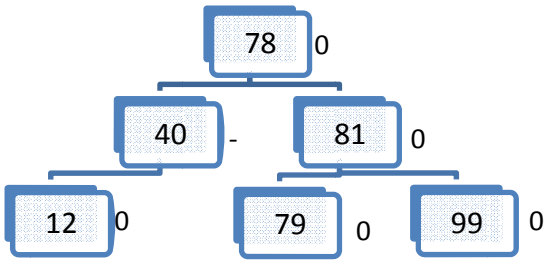
Untuk mempermudah menyeimbangkan tree, maka digunakan symbol-simbol bantu.

- - (tanda minus) : digunakan apabila subtree kiri lebih pan-jang dari subtree kanan
- + (tanda plus): digunakan apabila subtree kanan lebih panjang dari subtree kiri
- 0 (nol) : digunakan apabila subtree kiri dan subtree kanan mempunyai height yang sama.

Contoh AVLTree

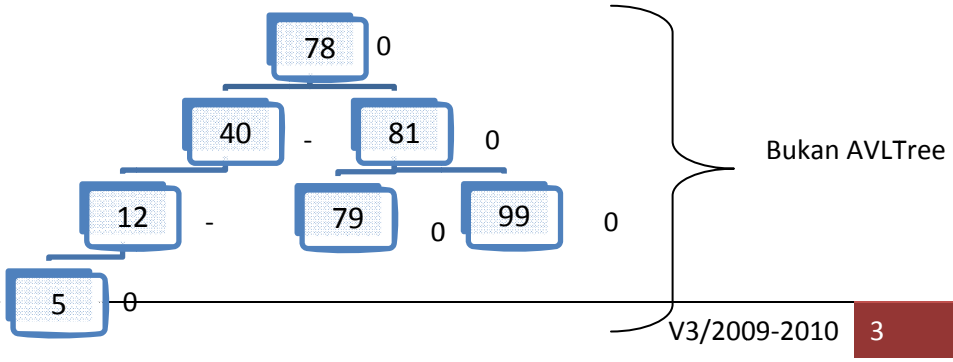


Contoh Operasi Insert pada AVLTree



Keadaan AVLTree mula-mula

Insert (5)



Supaya menjadi AVL Tree perlu dilakukan Single Rotation

