Pertemuan 12

Balanced Binary Tree Search

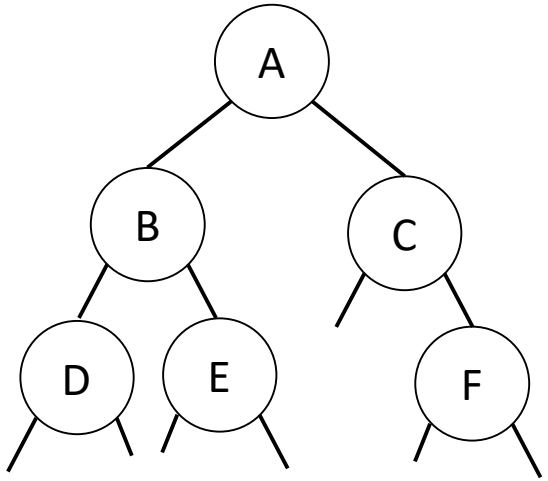
Oleh

Dr. Yoga Religia, S.Kom, M.Kom.

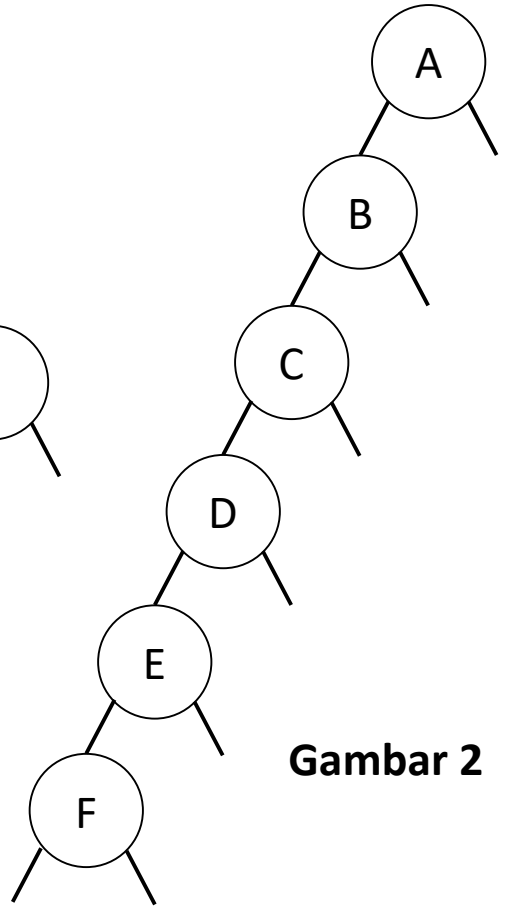
✆ 081390601900

✉ yoga.religia@pelitabangsa.ac.id

Pencarian Pohon Biner Berimbang



Gambar 1



Gambar 2

Waktu pencariannya akan berbeda jika ***balance*** dari pohon biner tersebut lebih baik.

Average Search Time = $\frac{\sum(\sum \boldsymbol{VL_n} * \boldsymbol{L}+\boldsymbol{1})}{\sum \boldsymbol{V}}$

- Pada gambar 1, Waktu rata-rata pencarian = (1*1 + 2*2 + 3*3) / 6 = 2.33
- Pencarian pada gambar 2, Waktu rata-rata pencarian = (1*1 + 1*2 + 1*3 + 1*4 + 1*5 + 1*6) / 6 = 3.5

Contoh Balance Binary Tree

Jika **Depth = 3**, **jumlah simpul = 15**

Maka, **Waktu rata-rata pencarian** = (1*1 + 2*2 + 4*3 + 8*4) / 15 = 3.27

Kedalaman	Jumlah Simpul	Waktu Rata-rata	Kenaikan
0	1	1.00	-
1	3	1.66	66 %
2	7	2.43	46 %
3	15	3.27	34 %
4	31	4.16	27 %
5	63	5.09	22 %

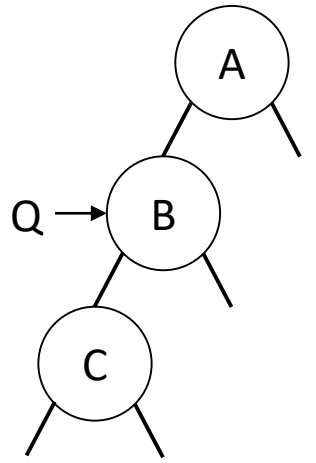
Meskipun jumlah simpul naik 2 kali lipat, karena pohon binernya ***balance***, waktu rata-rata pencarian tidak naik 2 kali lipat.

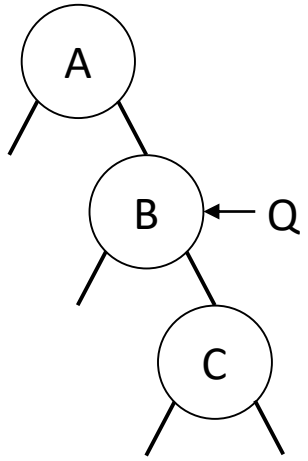
Algoritma Balenching Binary Tree ada 4 :

1. Right Rotation
2. Left Rotation
3. Right Rotation dilanjut Left Rotation
4. Left Rotation dilanjut Right Rotation

Algoritma Right Rotation & Left Rotation

```
Void RightRotation ( ){
    Simpul *Q ;
    Q = Root -> Left ;
    Root -> Left = Q -> Right ;
    Q -> Right = Root ;
    Root = Q ;
}
```

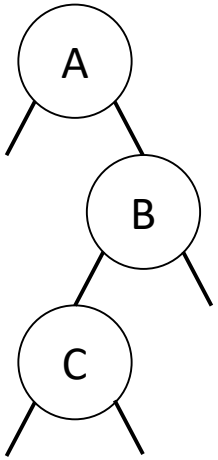




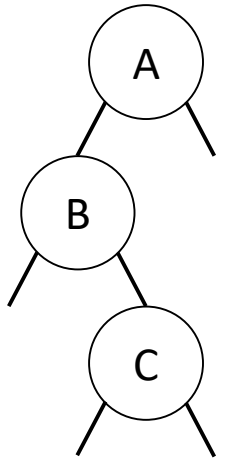

```
Void LeftRotation ( ){
    Simpul *Q ;
    Q = Root -> Right ;
    Root -> Right = Q -> Left ;
    Q -> Left = Root ;
    Root = Q ;
}
```

Latihan Individu : Buatlah algoritmanya agar binary tree tersebut dapat balance

1



2



Jawaban

Nomor 1.

Q = Root -> Right

R = Q -> Left

Q -> Left = R -> Right

R -> Right = Q

Root -> Right = R -> Left

R -> Left = Root

Root = R

Nomor 2.

Q = Root -> Left

R = Q -> Right

Q -> Right = R -> Left

R -> Left = Q

Root -> Left = R -> Right

R -> Right = Root

Root = R