

NAMA : Alfarez Syahputra Kuri

NIM : L200160067

KELAS : A

## MODUL 9

1. Diberikan pohon biner dengan ukuran  $n$ , berapakah jumlah level minimum yang bisa dimuatnya ? Berapakah jumlah level maksimum nya? Tentukan untuk nilai  $n$  dari

(a) .  $n = 10$

Level minimum = 3

Level maximum = 9

(b) .  $n = 35$

Level minimum = 6

Level maximum = 34

(c) .  $n = 76$

Level minimum = 7

Level maximum = 75

(d) .  $n = 345$

Level minimum = 8

Level maximum = 344

2. Gambarkanlah semua bentuk pohon biner berukuran 5 yang mungkin. Ada berapa kemungkinan ?

$$C_n = \frac{(2n)!}{(n+1)! \cdot n!}$$

$$= \frac{(2 \cdot 5)!}{(5+1)! \cdot 5!}$$

$$= \frac{10!}{6! \cdot 5!}$$

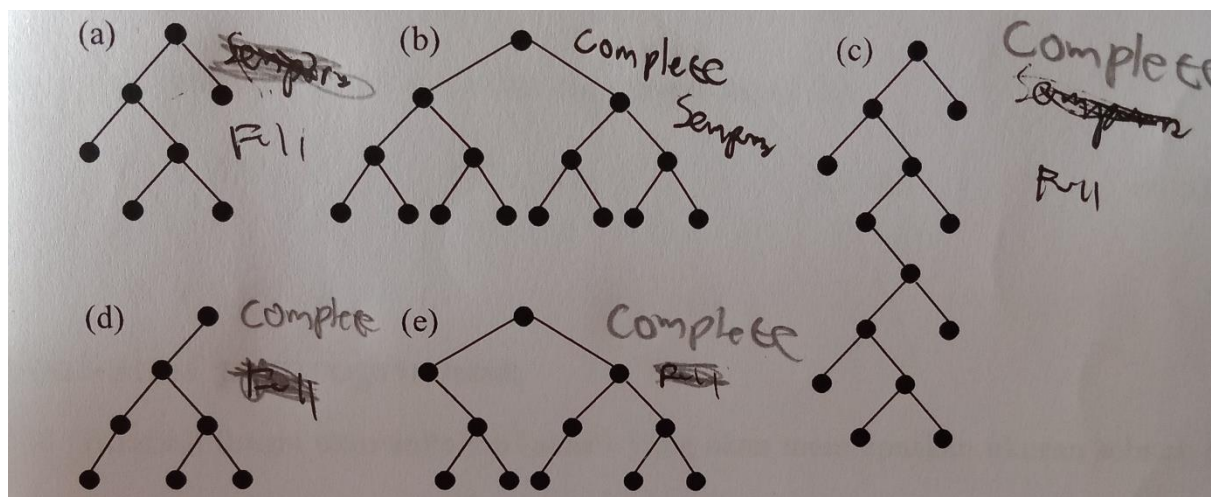
$$= \frac{3628800}{86400}$$

$$= 42 \text{ kemungkinan}$$

3. Berapakah jumlah simpul maksimum suatu pohon biner dengan jumlah level  $h$ ?  
Tentukan untuk nilai  $h$  berikut

<p>(a) <math>h = 3</math>  Jumlah max simpul = level 0 + level 1 + level 2  <math>= 1 + 2 + 4</math>  <math>= 7</math></p>	<p>(b) <math>h = 4</math>  Jumlah max simpul = level 0 + level 1 + level 2 + level 3  <math>= 1 + 2 + 4 + 8</math>  <math>= 15</math></p>
<p>(c) <math>h = 5</math>  Jumlah max simpul = level 0 + level 1 + level 2 + level 3 + level 4  <math>= 1 + 2 + 4 + 8 + 16</math>  <math>= 31</math></p>	<p>(d) <math>h = 6</math>  Jumlah max simpul = level 0 + level 1 + level 2 + level 3 + level 4 + level 5  <math>= 1 + 2 + 4 + 8 + 16 + 32</math>  <math>=</math></p>

4. Diberikan pohon-pohon biner seperti di bawah



(a) Tunjukkan semua properti struktural yang berlaku pada tiap-tiap pohon di atas :  
*penuhi* , *sempurna* , *komplet*. Ingat bahwa sebuah pohon biner bisa saja bersifat penuh  
sekaligus sempurna dan sebagainya.

(b) Tentukan ukuran tiap pohon.

$a = 7$	$b = 15$	$c = 14$
$d = 7$	$e = 11$	

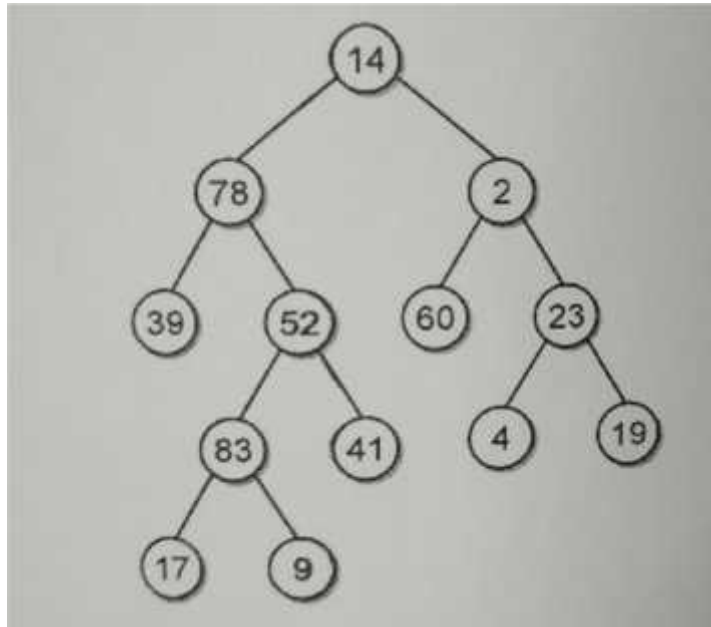
(c) Tentukan ketinggian tiap pohon.

$a = 4$	$b = 4$	$c = 8$
$d = 4$	$e = 4$	

(d) Tentukan lebar tiap pohon.

a = 2	b = 8	c = 2
d = 3	e = 5	

5. Perhatikan pohon biner berikut.



(a) Tunjukkan urutan pengunjungan simpul untuk :

- Preorder traversal = 14-78-39-52-83-17-9-41-2-60-23-4-19
- Inorder traversal = 39-78-17-83-9-52-41-14-60-2-4-23-19
- Postorder traversal = 39-17-9-83-41-52-78-60-4-19-23-2-14

(b) Simpul mana saja yang merupakan simpul daun ?

39, 17, 9, 41, 60, 4, 19

(c) Simpul mana saja yang merupakan simpul dalam ?

14, 78, 52, 83, 2, 23

(d) Simpul mana saja yang berada di level 4?

17, 9

(e) Tulis semua simpul yang berada di dalam jalur dari simpul akar menuju simpul

I. 83 = 14-78-52-83

II. 39 = 14-78-39

III. 4 = 14-2-23-4

IV. 9 = 14-78-52-83-9

(f) Perhatikan simpul 52. Tentukan

- Keturunannya = 83, 41
- Leluhurnya = 78, 14
- Saudaranya = 39

(g) Tentukan kedalaman dari tiap-tiap simpul ini :

- I. 78 = level 1
- II. 41 = level 2
- III. 60 = level 2
- IV. 19 = level 3

### Soal-soal pemograman

6. Buatlah fungsi **ukuranPohon** (akar) yang akan mendapatkan ukuran sebuah pohon biner.

```
class simpulpohonbiner(object):
    def __init__(self, data):
        self.data = data
        self.kiri = None
        self.kanan = None

    def ukuranpohon (akar, count = 0):
        if akar is None :
            return count
        return ukuranpohon(akar.kiri, ukuranpohon(akar.kanan, count +))

a = simpulpohonbiner ('Ambawara')
b = simpulpohonbiner ('Bantul')
c = simpulpohonbiner ('Cimahi')
d = simpulpohonbiner ('Denpasar')
e = simpulpohonbiner ('Enrekang')
f = simpulpohonbiner ('Flores')
g = simpulpohonbiner ('Garut')
h = simpulpohonbiner ('Halmahera Timur')
i = simpulpohonbiner ('Indramayu')

a.kiri = b; a.kanan = c
b.kiri = d; ab.kanan = c
c.kiri = f; c.kanan = g
e.kiri = h
g.kanan = i
```

```
===== RESTART: C:/Users/USER/Biner.py =====
>>> ukuranpohon(a)
9
>>> |
```

7. Buatlah sebuah fungsi **tinggiPohon** (akar) yang akan mendapatkan ketinggian sebuah pohon biner.

```
#tinggi
class tinggipohonbiner(object):
    def __init__(self, data):
        self.data = data
        self.kiri = None
        self.kanan = None

    def tinggipohon (akar, count = 0):
        if akar is None :
            return 0
        else :
            return max(tinggipohon(akar.kiri), tinggipohon(akar.kanan))+1

a = simpulpohonbiner ('Ambawara')
b = simpulpohonbiner ('Bantul')
c = simpulpohonbiner ('Cimahi')
d = simpulpohonbiner ('Denpasar')
e = simpulpohonbiner ('Enrekang')
f = simpulpohonbiner ('Flores')
g = simpulpohonbiner ('Garut')
h = simpulpohonbiner ('Halmahera Timur')
i = simpulpohonbiner ('Indramayu')

a.kiri = b; a.kanan = c
b.kiri = d; b.kanan = e
c.kiri = f; c.kanan = g
e.kiri = h
g.kanan = i

===== RESTART: C:/Users/USER/Biner.py =====
>>> tinggipohon(a)
4
>>> |
```

8. Buatlah sebuah fungsi yang mencetak data tiap simpul sekaligus level dimana simpul itu berada. Silahkan memilih akan memakai *preorder traversal*, *inorder traversal*, atau *postorder traversal*. Contoh sepotong hasilnya adalah seperti di bawah ini (jika kamu memakai *preorder traversal*).

```
>>> cetakDataDanLevel (A)
Ambarawa, level 0
```

Bantul, level 1

Denpasar, level 2

Enrekang, level 2

Halmahera Timur, level 3

Cimahi, level 1

