



**NAMA : KEVIN AVICENNA WIDIARTO**

**NIM : L200200183**

**Modul : 9**

**Praktikum Algoritma Struktur Data**

## MODUL 9

Soal-soal untuk mahasiswa

### 1. Menentukan Nilai N

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. Menentukan 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. Menentukan Nilai H

A.  $h = 3$

Jumlah max simpul = level 0 + level 1 + level 2

$$= 1 + 2 + 4$$

$$= 7$$

B.  $h = 4$

Jumlah max simpul = level 0 + level 1 + level 2 + level 3

$$= 1 + 2 + 4 + 8$$

$$= 15$$

C.  $h = 5$

Jumlah max simpul = level 0 + level 1 + level 2 + level 3 + level 4

$$= 1 + 2 + 4 + 8 + 16$$

$$= 31$$

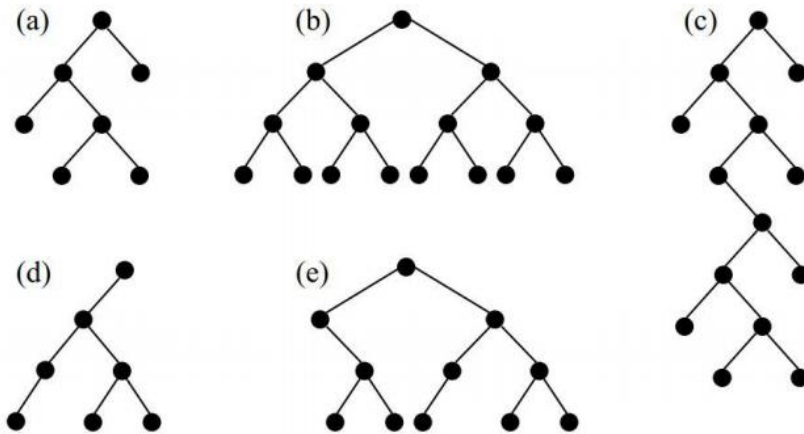
D.  $h = 6$

Jumlah max simpul = level 0 + level 1 + level 2 + level 3 + level 4 + level 5

$$= 1 + 2 + 4 + 8 + 16 + 32$$

$$= 63$$

4. Diberikan pohon-pohon biner seperti di bawah



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

- a = penuh
- b = sempurna
- c = komplet dan penuh
- d = komplet
- e = komplet

b. Tentukan ukuran tiap pohon.

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

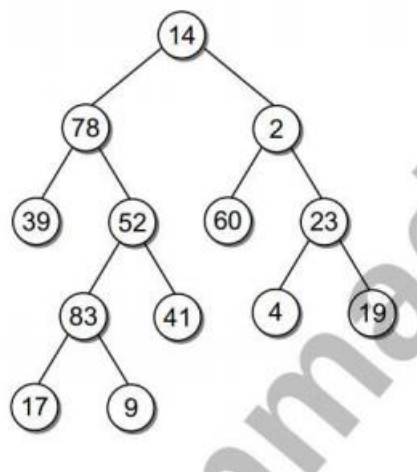
c. Tentukan ketinggian tiap pohon.

- 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 :

- I. Preorder traversal = 14-78-39-52-83-17-9-41-2-60-23-4-19
- II. Inorder traversal = 39-78-17-83-9-52-41-14-60-2-4-23-19
- III. 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.  $83 = 15 - 78 - 52 - 83$

II.  $39 = 14 - 78 - 39$

III.  $4 = 14 - 2 - 23 - 4$

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

(f) Perhatikan simpul 52. Tentukan

I. Keturunannya = 83, 41

II. Leluhurnya = 78, 14

III. 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

6.

```
File Edit Format Run Options Window Help
class SimpulPohonBiner(object):
    def __init__(self, data):
        self.data = data
        self.kiri = None
        self.kanan = None
def ukuranpohon(root, count=0):
    if root is None:
        return count
    return ukuranpohon(root.kiri, ukuranpohon(root.kanan, count+1))

IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
===== RESTART: E:\PRAK ASD\Modul 9\pohonbiner.py =====
>>> ukuranpohon(A)
9
>>>
```

7.

```
pohonbiner.py - E:\PRAK ASD\Modul 9\pohonbiner.py (3.10.2)
File Edit Format Run Options Window Help
class SimpulPohonBiner(object):
    def __init__(self, data):
        self.data = data
        self.kiri = None
        self.kanan = None
def ukuranpohon(root, count=0):
    if root is None:
        return count
    return ukuranpohon(root.kiri, ukuranpohon(root.kanan, count+1))
def tinggipohon(root):
    if root is None:
        return 0
    else:
        return max(tinggipohon(root.kiri), tinggipohon(root.kanan)) + 1

IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
===== RESTART: E:\PRAK ASD\Modul 9\pohonbiner.py =====
>>> tinggipohon(A)
4
```

8.

File Edit Format Run Options Window Help

```
class SimpulPohonBiner(object):
    def __init__(self, data):
        self.data = data
        self.kiri = None
        self.kanan = None
def ukuranpohon(root, count=0):
    if root is None:
        return count
    return ukuranpohon(root.kiri, ukuranpohon(root.kanan, count+1))

def tinggipohon(root):
    if root is None:
        return 0
    else:
        return max(tinggipohon(root.kiri), tinggipohon(root.kanan))+1
def cetak(subpohon, count = 0 ):
    if subpohon is not None:
        print(subpohon.data + ', level ' +str(count))
        (cetak(subpohon.kiri, count+1), cetak(subpohon.kanan, count+1))
```

IDLE Shell 3.10.2

File Edit Shell Debug Options Window Help

```
>>>
===== RESTART: E:\PRAK ASD\Modul 9\pohonbiner.py =====
>>> cetak(A)
Ambarawa, level 0
Bantul, level 1
Denpasar, level 2
Enrekang, level 2
Halmahera Timur, level 3
Cimahi, level 1
Flores, level 2
Garut, level 2
Indramayu, level 3
>>>
```