

Nama : Nugroho Prihananto

NIM : L200170186

Kelas : E

MODUL 9

Soal-soal untuk mahasiswa

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

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 ?

Jawab :

$$\begin{aligned} C_n &= (2n)! / ((n+1)! * n!) \\ &= (2*5)! / (5+1)! * 5! \\ &= 10! / 6! * 5! \\ &= 3628800 / 86400 \\ &= 42 \text{ kemungkinan} \end{aligned}$$

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

a. $h = 3$

$$\begin{aligned} \text{Jumlah max simpul} &= \text{level 0} + \text{level 1} + \text{level 2} \\ &= 1 + 2 + 4 \\ &= 7 \end{aligned}$$

b. $h = 4$

$$\begin{aligned}
 \text{Jumlah max simpul} &= \text{level 0} + \text{level 1} + \text{level 2} + \text{level 3} \\
 &= 1 + 2 + 4 + 8 \\
 &= 15
 \end{aligned}$$

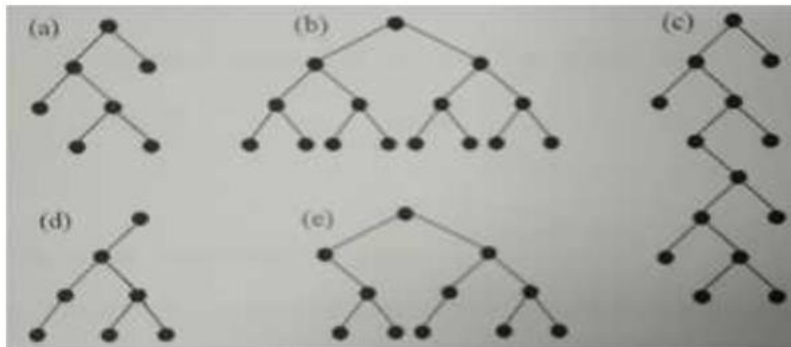
c. $h = 5$

$$\begin{aligned}
 \text{Jumlah max simpul} &= \text{level 0} + \text{level 1} + \text{level 2} + \text{level 3} + \text{level 4} \\
 &= 1 + 2 + 4 + 8 + 16 \\
 &= 31
 \end{aligned}$$

d. $h = 6$

$$\begin{aligned}
 \text{Jumlah max simpul} &= \text{level 0} + \text{level 1} + \text{level 2} + \text{level 3} + \text{level 4} + \text{level 5} \\
 &= 1 + 2 + 4 + 8 + 16 + 32 \\
 &= 63
 \end{aligned}$$

4. Diberikan pohon-pohon biner seperti di bawah

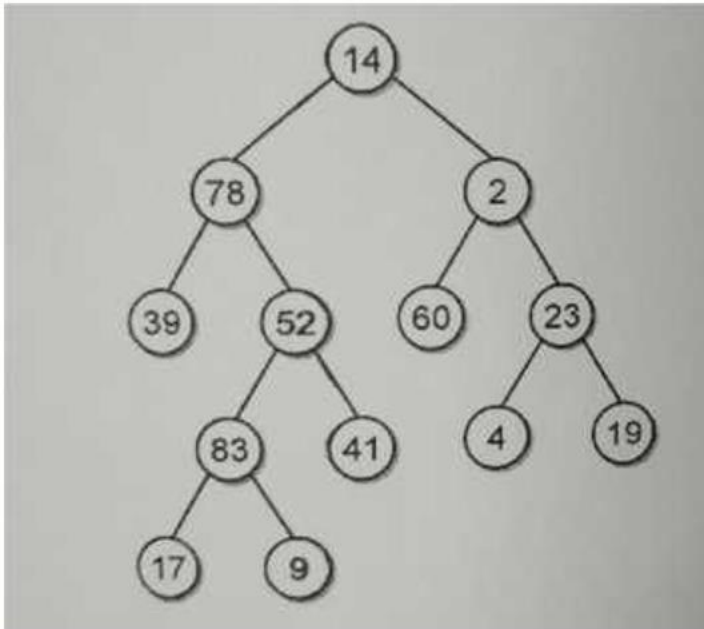


a. Tunjukkan semua properti struktural yang berlaku pada tiap-tiap pohon di atas :
 penuh , sempurna, komplet. Ingat bahwa sebuah pohon biner bisa saja bersifat penuh sekaligus sempurna dan sebagainya.

- 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 = 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
 - i. Keturunannya = 83, 41, 17, 9
 - ii. Leluhurnya = 14, 78

- iii. Saudaranya = 39
- g. Tentukan kedalaman dari tiap-tiap simpul ini :
 - i. 78 = 1
 - ii. 41 = 3
 - iii. 60 = 2
 - iv. 19 = 3

Soal – soal pemrograman

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

```
Python 3.4.0: no6.py - D:/semester 4/prak asd/no6.py
File Edit Format Run Options Windows Help

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

    def __str__(self):
        return str(self.data)

A=simpulpohonbiner('Ambarawa')
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')
J=simpulpohonbiner('Jakarta')

A.kiri=B; A.kanan=C
B.kiri=D; B.kanan=E
C.kiri=F; C.kanan=G
E.kiri=H
G.kanan=I

def ukuranpohon(akar):
    if akar is None:
        return 0
    else:
        return (ukuranpohon(akar.kiri)+ 1 + ukuranpohon(akar.kanan))

print('Ukuran dari Binary Tree adalah', ukuranpohon(A))

>>> ===== RESTART =====
>>>
Ukuran dari Binary Tree adalah 9
```

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

```
*Python 3.4.0: no7.py - D:/semester 4/prak asd/no7.py*
File Edit Format Run Options Windows Help

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

A=simpulpohonbiner('Ambarawa')
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')
J=simpulpohonbiner('Jakarta')

A.kiri=B; A.kanan=C
B.kiri=D; B.kanan=E
C.kiri=F; C.kanan=G
E.kiri=H
G.kanan=I

def tinggipohon(akar):
    if akar is None:
        return 0 ;

    else :
        lDepth = tinggipohon(akar.kiri)
        rDepth = tinggipohon(akar.kanan)

        if (lDepth > rDepth):
            return lDepth+1
        else:
            return rDepth+1

print('Tinggi maksimal dari Binary Tree adalah', tinggipohon(A))

>>> ===== RESTART =====
>>>
Tinggi maksimal dari Binary Tree adalah 4
>>> ===== RESTART =====
```

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.

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

    def __str__(self):
        return str(self.data)

A=simpulpohonbiner('Ambarawa')
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')
J=simpulpohonbiner('Jakarta')

A.kiri=B; A.kanan=C
B.kiri=D; B.kanan=E
C.kiri=F; C.kanan=G
E.kiri=H
G.kanan=I

datalist=[A.data, B.data, C.data, D.data, E.data, F.data, G.data, H.data, I.data, J.data]
level=[]

def traverse(root):
    lvlist=[]
    current_level = [root]
    lv=0
    while current_level:
        next_level = list()
        for n in current_level:
            if n.kiri:
                next_level.append(n.kiri)
                level.append(lv+1)
            if n.kanan:
                next_level.append(n.kanan)
                level.append(lv+1)
            current_level = next_level

        lv+=1
        lvlist.append(lv)
    return lvlist

def cetakdatadanlevel(root):
    traverse(A)
    print(root.data, ', Level 0')
    for i in range(len(level)):
        print(datalist[i+1], ', Level', level[i])

cetakdatadanlevel(A)
```

```
>>> ===== RESIARI =====
>>>
Ambarawa , Level 0
Bantul , Level 1
Cimahi , Level 1
Denpasar , Level 2
Enrekang , Level 2
Flores , Level 2
Garut , Level 2
Halmahera Timur , Level 3
Indramayu , Level 3
```