

Nama : Irfananda Irsyad A  
NIM : L200180096  
Kelas : D

## Modul 9

### Nomor 1

a.  $n = 10$

Level minimum:  $\log_2 10 + 1 = 4$

Level maksimum:  $10 - 1 = 9$

b.  $n = 35$

Level minimum:  $\log_2 35 + 1 = 6$

Level maksimum:  $35 - 1 = 34$

c.  $n = 76$

Level minimum:  $\log_2 76 + 1 = 7$

Level maksimum:  $76 - 1 = 75$

d.  $n = 345$

Level minimum:  $\log_2 345 + 1 = 9$

Level maksimum:  $345 - 1 = 344$

### Nomor 2

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

### Nomor 3

a.

$h = 3$

$2^3 - 1 = 7$

b.

$$h = 4$$

$$2^4 - 1 = 15$$

c.

$$h = 5$$

$$2^5 - 1 = 31$$

d.

$$h = 6$$

$$2^6 - 1 = 63$$

#### **Nomor 4**

##### **a. Sifat sebuah pohon**

- Pohon biner penuh: pohon a dan pohon c
- Pohon biner sempurna: pohon b
- Pohon biner komplet: pohon d dan pohon e

##### **b. Ukuran**

- Pohon a: 7
- Pohon b: 15
- Pohon c: 14
- Pohon d: 7
- Pohon e: 11

##### **c. Ketinggian**

- Pohon a: 4
- Pohon b: 4
- Pohon c: 8
- Pohon d: 4
- Pohon e: 4

##### **d. lebar**

- Pohon a: 2
- Pohon b: 8
- Pohon c: 2
- Pohon d: 3
- Pohon e: 5

#### **Nomor 5**

##### **a. Urutan pengunjungan**

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

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

c. 14, 78, 52, 83, 2, 23

d. 17 dan 9

e. Jalur tujuan menuju simpul

- 83: 14, 78, 52
- 39: 14, 78
- 4: 14, 2, 23
- 9: 14, 78, 52, 83

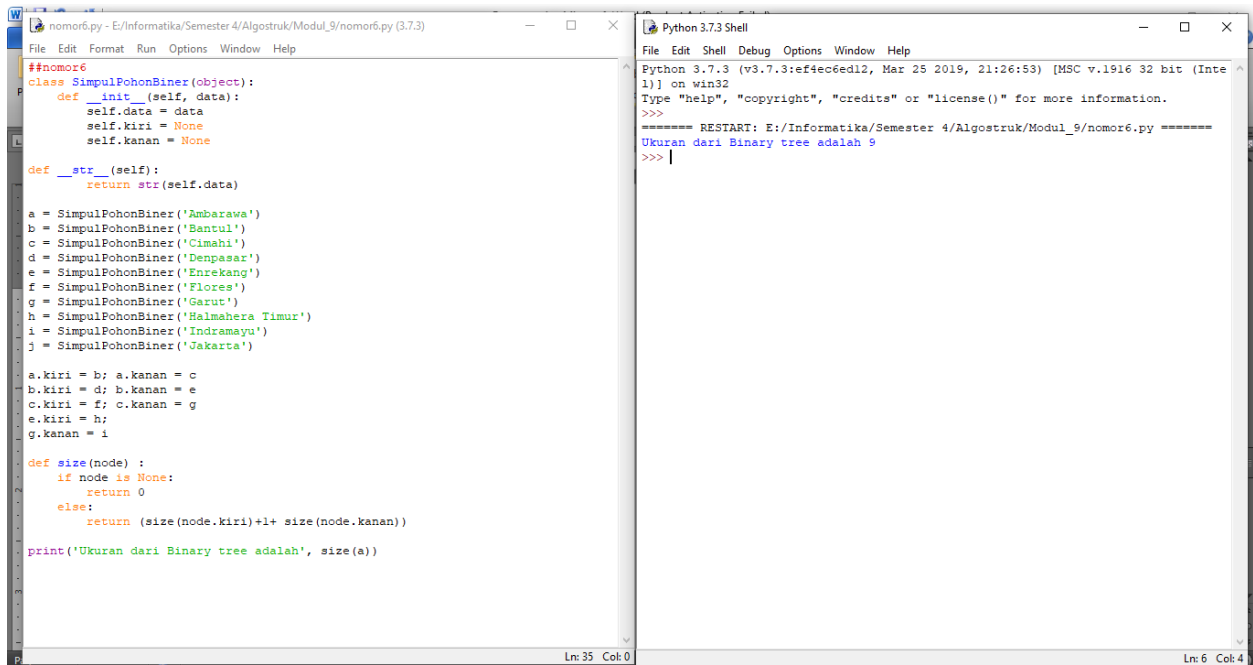
f. Menentukan silsilah

- Keturunan: 83, 41, 17, 9
- Leluhurnya: 14, 78
- Saudaranya: 39

g. Menentukan kedalaman

- 78: 1
- 41: 3
- 60: 2
- 19: 3

## Nomor 6



```
##nomor6
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

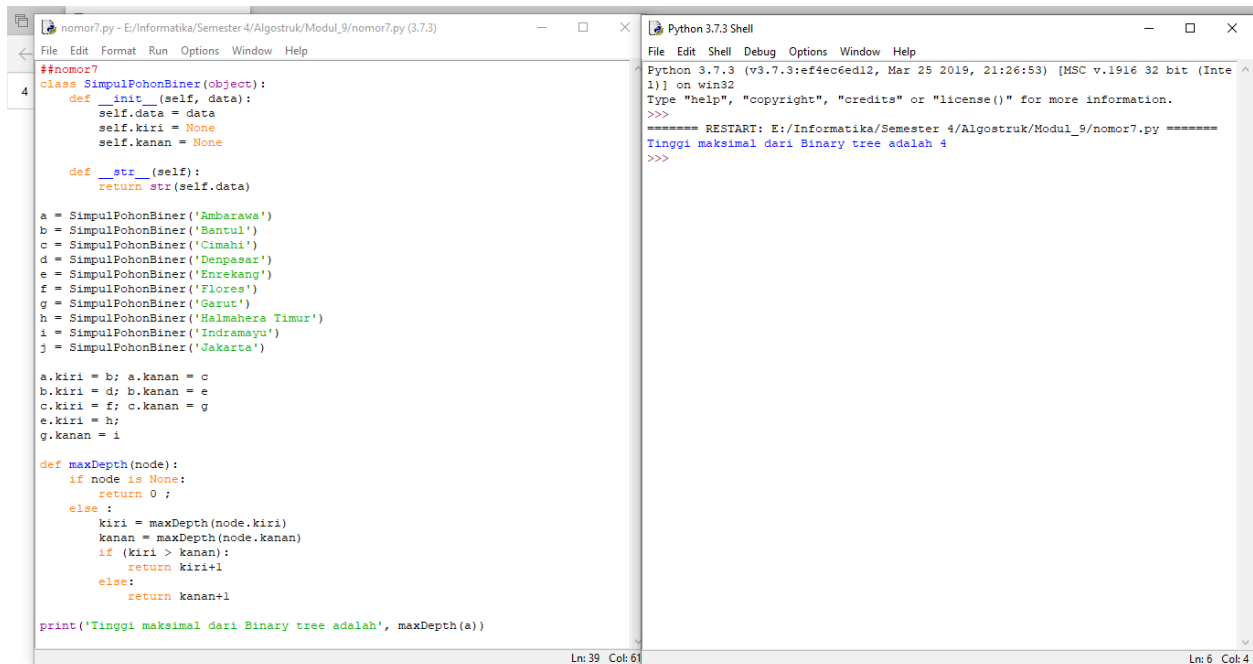
def size(node):
    if node is None:
        return 0
    else:
        return (size(node.kiri)+1+ size(node.kanan))

print('Ukuran dari Binary tree adalah', size(a))
```

Python 3.7.3 Shell

```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:/Informatika/Semester 4/Algostruk/Modul_9/nomor6.py =====
Ukuran dari Binary tree adalah 9
>>>
```

## Nomor 7



```
##nomor7
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 maxDepth(node):
    if node is None:
        return 0
    else:
        kiri = maxDepth(node.kiri)
        kanan = maxDepth(node.kanan)
        if (kiri > kanan):
            return kiri+1
        else:
            return kanan+1

print('Tinggi maksimal dari Binary tree adalah', maxDepth(a))
```

Python 3.7.3 Shell

```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:/Informatika/Semester 4/Algostruk/Modul_9/nomor7.py =====
Tinggi maksimal dari Binary tree adalah 4
>>>
```

## Nomor 8

```

E:\Informatika\Semester 4\Algostruk\Modul_9\nomor8.py (3.7.3)*
File Edit Format Run Options Window Help
##nomor8
class simpulbiner(object):
    def __init__(self, data):
        self.data=data
        self.kiri=None
        self.kanan=None

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

A=simpulbiner('Ambarawa')
B=simpulbiner('Bantul')
C=simpulbiner('Cimahi')
D=simpulbiner('Denpasar')
E=simpulbiner('Enrekang')
H=simpulbiner('Halmahera Timur')

A.kiri=B; A.kanan=C
B.kiri=D; B.kanan=E
D.kiri=H;

datalist=[A.data, B.data, C.data, D.data, E.data, H.data]
level=[]

def preorder(sub):
    if sub is not None:
        print(sub.data)
        preorder(sub.kiri)
        preorder(sub.kanan)

def inorder(sub):
    if sub is not None:
        inorder(sub.kiri)
        print(sub.data)
        inorder(sub.kanan)

def postorder(sub):
    if sub is not None:
        postorder(sub.kiri)
        postorder(sub.kanan)

Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel
)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:/Informatika/Semester 4/Algostruk/Modul_9/nomor8.py =====
Ambarawa , Level 0
Bantul , Level 1
Cimahi , Level 1
Denpasar , Level 2
Enrekang , Level 2
Halmahera Timur , Level 3
=====preorder=====
Ambarawa
Bantul
Denpasar
Halmahera Timur
Enrekang
Cimahi
=====inorder=====
Halmahera Timur
Denpasar
Bantul
Enrekang
Ambarawa
Cimahi
=====postorder=====
Halmahera Timur
Denpasar
Enrekang
Bantul
Cimahi
Ambarawa
>>> |
Ln: 1 Col: 2
Ln: 32 Col: 4
08:44 PM
11-May-20
```

