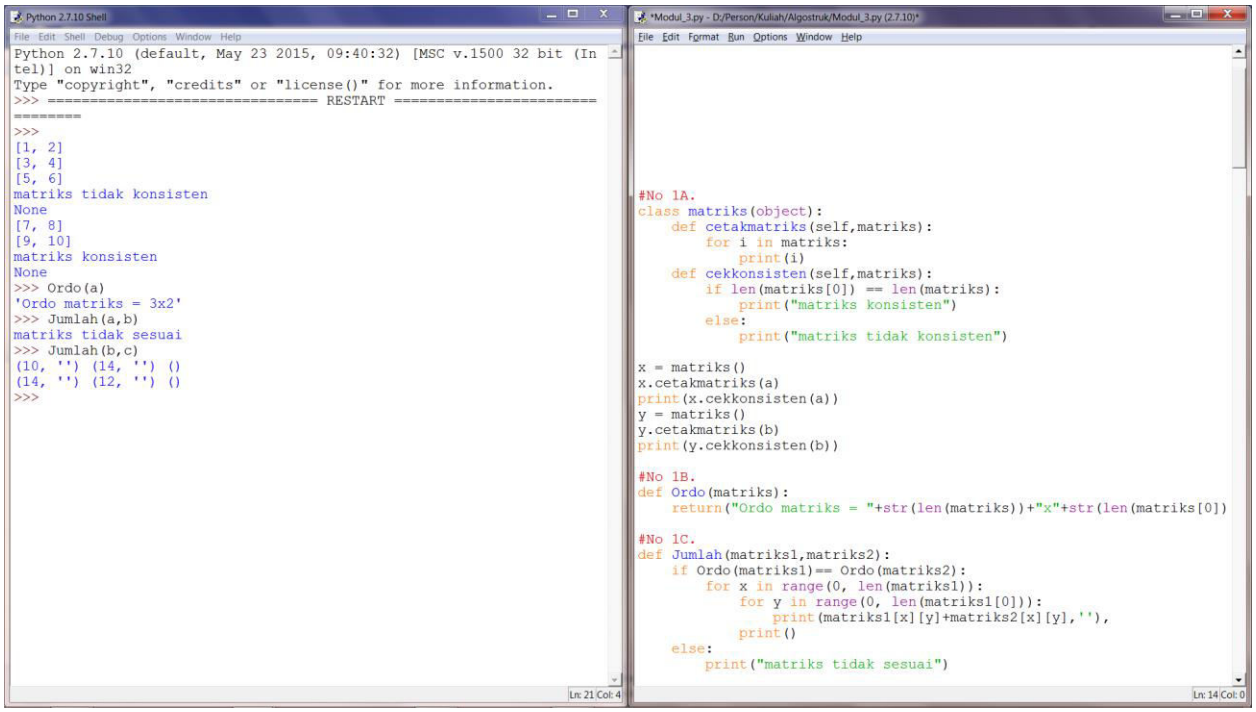
Nama : Fandhitya Giovani
Nim : L200180118
Kelas : E

Tugas

Modul 3

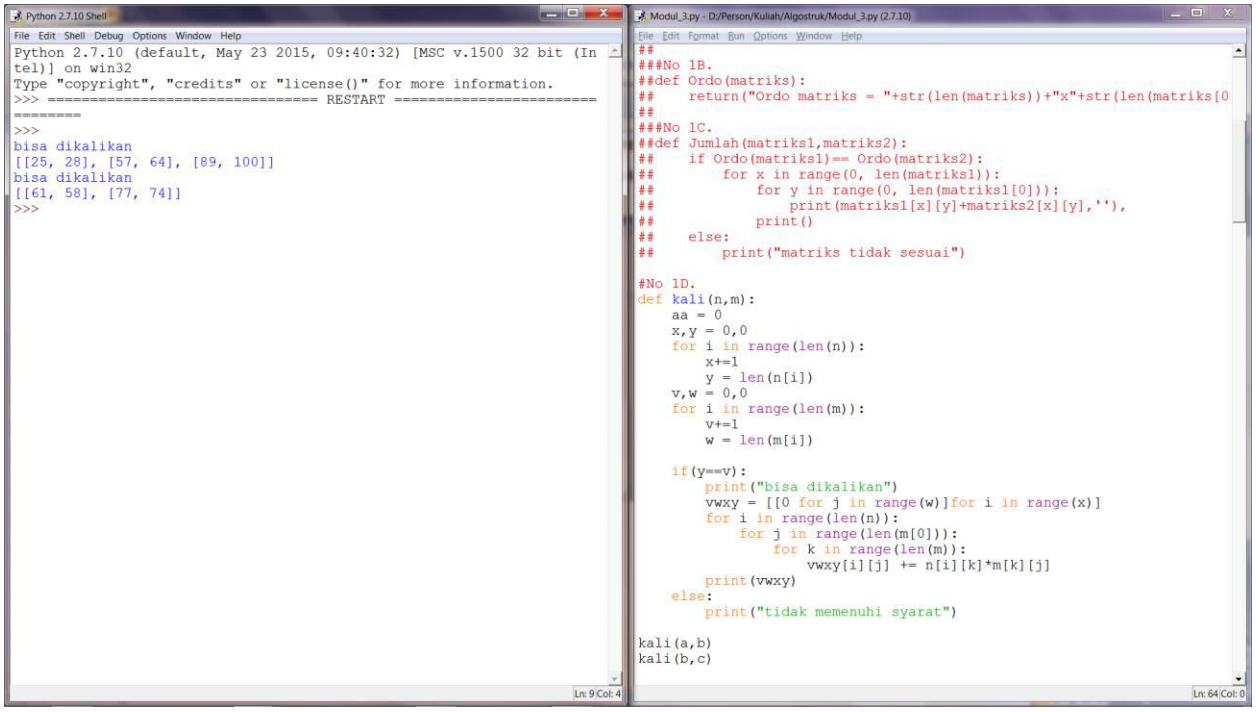
Nomer 1A, 1B, 1C



```
Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (In
tel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
=========
>>>
[1, 2]
[3, 4]
[5, 6]
matriks tidak konsisten
None
[7, 8]
[9, 10]
matriks konsisten
None
>>> Ordo(a)
'Ordo matriks = 3x2'
>>> Jumlah(a,b)
matriks tidak sesuai
>>> Jumlah(b,c)
(10, '') (14, '') ()
(14, '') (12, '') ()
>>>
```

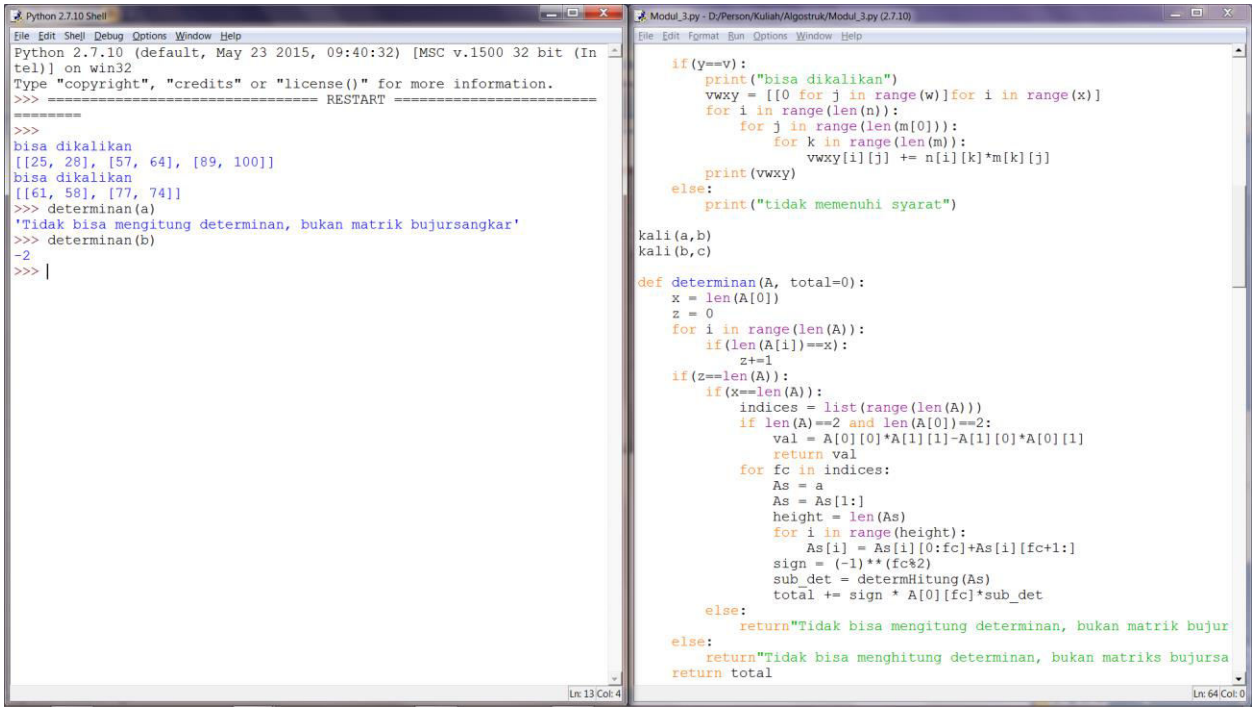
```
*Modul_3.py - D:/Person/Kuliah/Algostruk/Modul_3.py (2.7.10)*
File Edit Format Run Options Window Help
#No 1A.
class matriks(object):
    def cetakmatriks(self,matriks):
        for i in matriks:
            print(i)
    def cekkonsisten(self,matriks):
        if len(matriks[0]) == len(matriks):
            print("matriks konsisten")
        else:
            print("matriks tidak konsisten")

x = matriks()
x.cetakmatriks(a)
print(x.cekkonsisten(a))
y = matriks()
y.cetakmatriks(b)
print(y.cekkonsisten(b))

#No 1B.
def Ordo(matriks):
    return("Ordo matriks = "+str(len(matriks))+"x"+str(len(matriks[0])

#No 1C.
def Jumlah(matriks1,matriks2):
    if Ordo(matriks1)== Ordo(matriks2):
        for x in range(0, len(matriks1)):
            for y in range(0, len(matriks1[0])):
                print(matriks1[x][y]+matriks2[x][y],''),
            print()
    else:
        print("matriks tidak sesuai")
```

Nomer 1D



```
Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (In
tel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
========
>>>
bisa dikalikan
[[25, 28], [57, 64], [89, 100]]
bisa dikalikan
[[61, 58], [77, 74]]
>>>
```

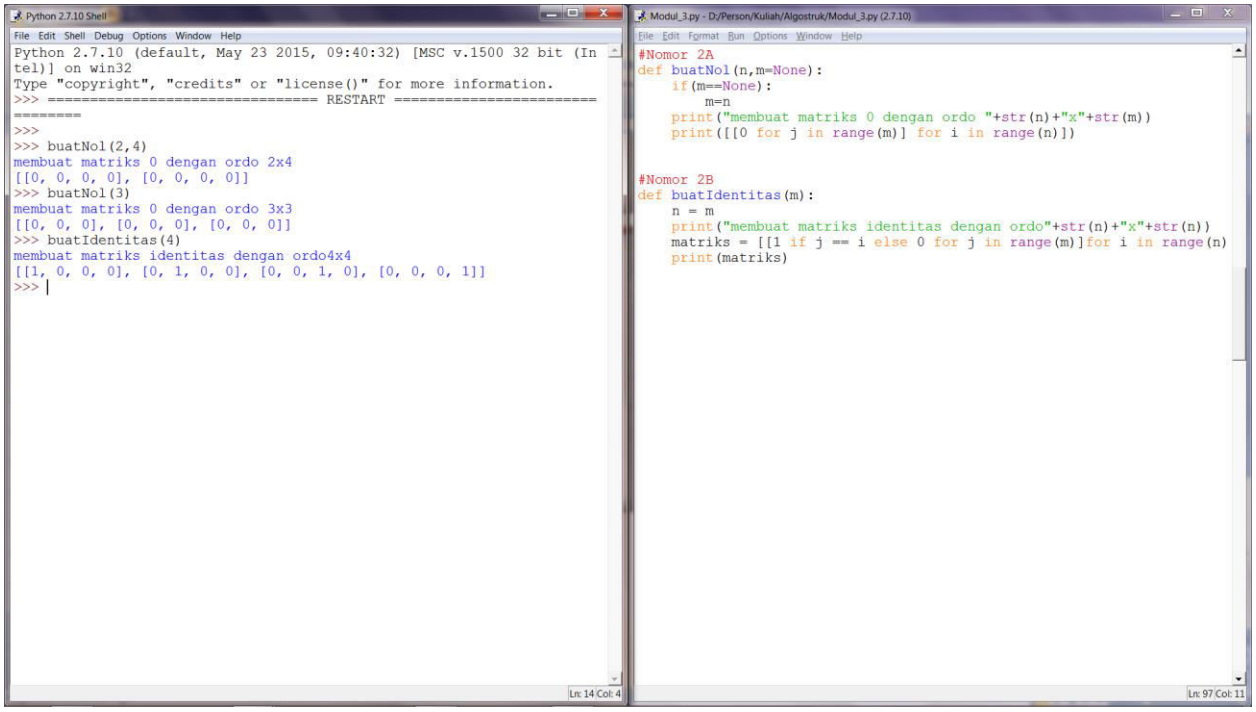
```
Modul_3.py - D:/Person/Kuliah/Algostruk/Modul_3.py (2.7.10)
File Edit Format Run Options Window Help
##
###No 1B.
##def Ordo(matriks):
##    return("Ordo matriks = "+str(len(matriks))+"x"+str(len(matriks[0
##
###No 1C.
##def Jumlah(matriks1,matriks2):
##    if Ordo(matriks1)== Ordo(matriks2):
##        for x in range(0, len(matriks1)):
##            for y in range(0, len(matriks1[0])):
##                print(matriks1[x][y]+matriks2[x][y],''),
##            print()
##    else:
##        print("matriks tidak sesuai")

#No 1D.
def kali(n,m):
    aa = 0
    x,y = 0,0
    for i in range(len(n)):
        x+=1
        y = len(n[i])
    v,w = 0,0
    for i in range(len(m)):
        v+=1
        w = len(m[i])

    if(y==v):
        print("bisa dikalikan")
        vwxy = [[0 for j in range(w)]for i in range(x)]
        for i in range(len(n)):
            for j in range(len(m[0])):
                for k in range(len(m)):
                    vwxy[i][j] += n[i][k]*m[k][j]
        print(vwxy)
    else:
        print("tidak memenuhi syarat")

kali(a,b)
kali(b,c)
```



```
Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (In
tel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
========
>>>
bisa dikalikan
[[25, 28], [57, 64], [89, 100]]
bisa dikalikan
[[61, 58], [77, 74]]
>>> determinan(a)
'Tidak bisa mengitung determinan, bukan matrik bujursangkar'
>>> determinan(b)
-2
>>>
```

```
Modul_3.py - D:/Person/Kuliah/Algostruk/Modul_3.py (2.7.10)
File Edit Format Run Options Window Help
    if(y==v):
        print("bisa dikalikan")
        vwxy = [[0 for j in range(w)]for i in range(x)]
        for i in range(len(n)):
            for j in range(len(m[0])):
                for k in range(len(m)):
                    vwxy[i][j] += n[i][k]*m[k][j]
        print(vwxy)
    else:
        print("tidak memenuhi syarat")

kali(a,b)
kali(b,c)

def determinan(A, total=0):
    x = len(A[0])
    z = 0
    for i in range(len(A)):
        if(len(A[i])==x):
            z+=1
    if(z==len(A)):
        if(x==len(A)):
            indices = list(range(len(A)))
            if len(A)==2 and len(A[0])==2:
                val = A[0][0]*A[1][1]-A[1][0]*A[0][1]
                return val
            for fc in indices:
                As = a
                As = As[1:]
                height = len(As)
                for i in range(height):
                    As[i] = As[i][0:fc]+As[i][fc+1:]
                sign = (-1)**(fc%2)
                sub_det = determHitung(As)
                total += sign * A[0][fc]*sub_det
        else:
            return"Tidak bisa mengitung determinan, bukan matrik bujur
    else:
        return"Tidak bisa menghitung determinan, bukan matriks bujursa
    return total
```

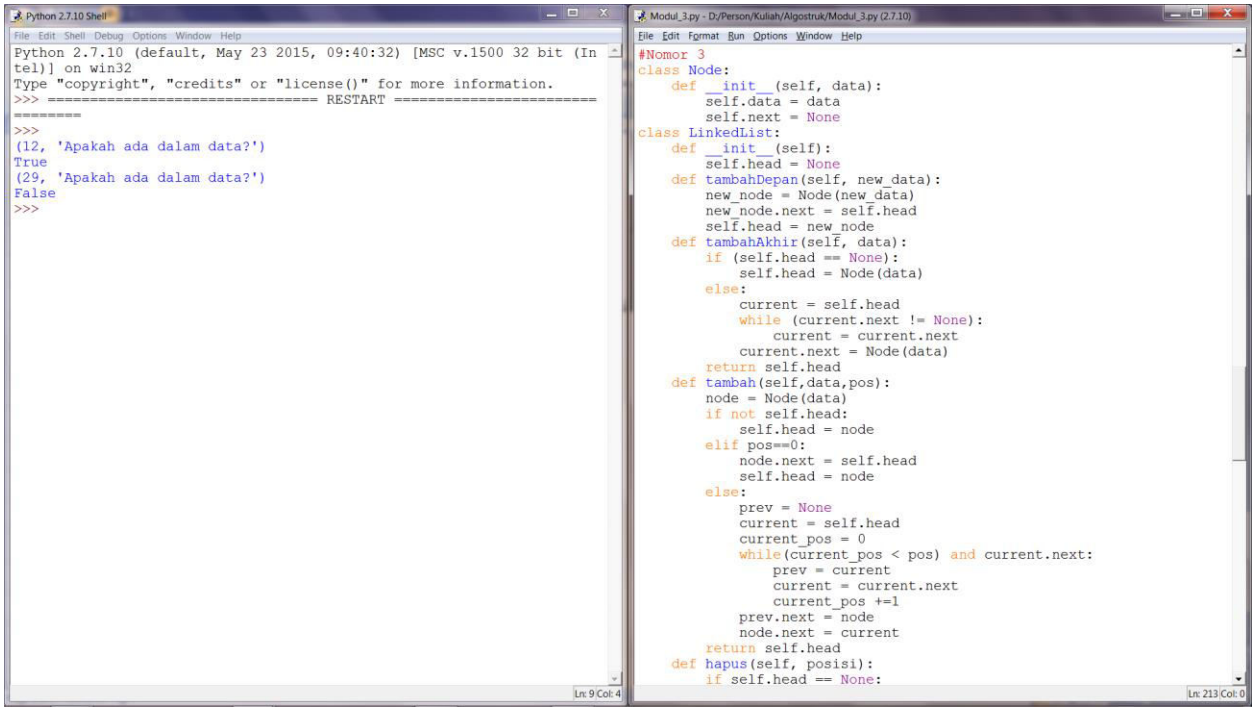
Nomer 2A, 2B



```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
>>> buatNol(2,4)
membuat matriks 0 dengan ordo 2x4
[[0, 0, 0, 0], [0, 0, 0, 0]]
>>> buatNol(3)
membuat matriks 0 dengan ordo 3x3
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
>>> buatIdentitas(4)
membuat matriks identitas dengan ordo4x4
[[1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], [0, 0, 0, 1]]
>>>
```

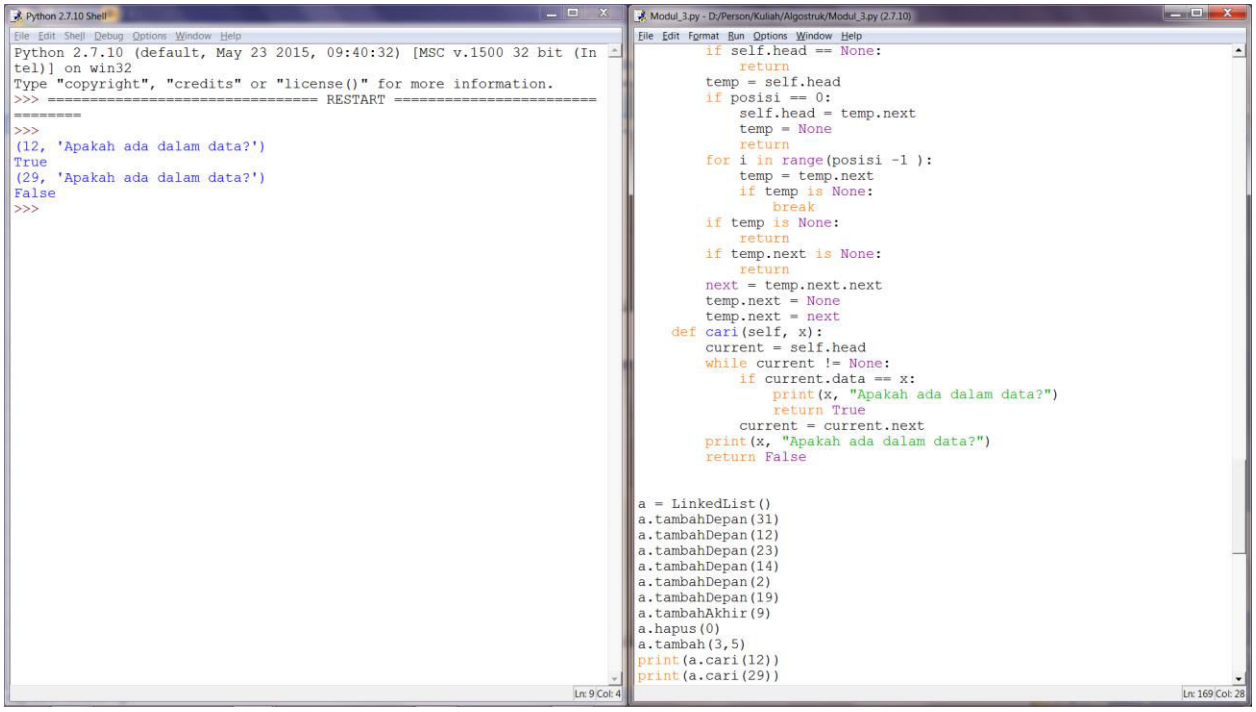
```
#Nomor 2A
def buatNol(n,m=None):
    if(m==None):
        m=n
    print("membuat matriks 0 dengan ordo "+str(n)+"x"+str(m))
    print([[0 for j in range(m)] for i in range(n)])


#Nomor 2B
def buatIdentitas(m):
    n = m
    print("membuat matriks identitas dengan ordo"+str(n)+"x"+str(n))
    matriks = [[1 if j == i else 0 for j in range(m)]for i in range(n)
    print(matriks)
```

Nomer 3



```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
(12, 'Apakah ada dalam data?')
True
(29, 'Apakah ada dalam data?')
False
>>>
```

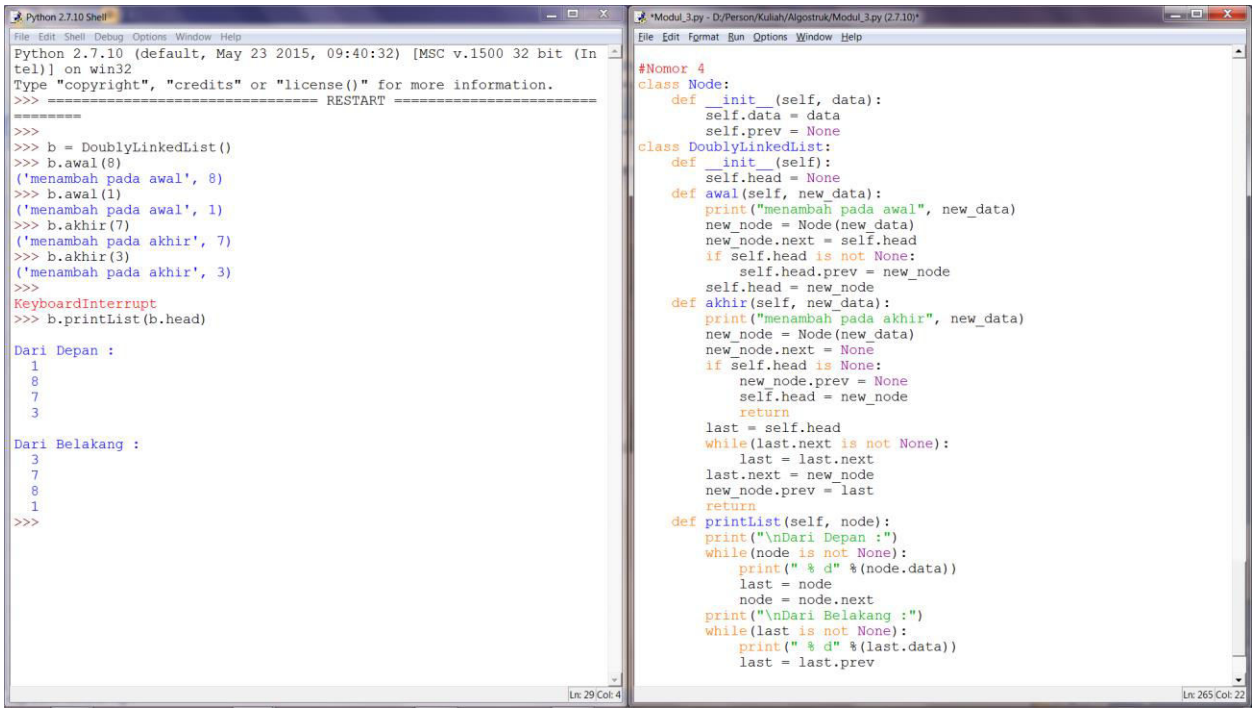
```
#Nomor 3
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None
class LinkedList:
    def __init__(self):
        self.head = None
    def tambahDepan(self, new_data):
        new_node = Node(new_data)
        new_node.next = self.head
        self.head = new_node
    def tambahAkhir(self, data):
        if (self.head == None):
            self.head = Node(data)
        else:
            current = self.head
            while (current.next != None):
                current = current.next
            current.next = Node(data)
        return self.head
    def tambah(self,data,pos):
        node = Node(data)
        if not self.head:
            self.head = node
        elif pos==0:
            node.next = self.head
            self.head = node
        else:
            prev = None
            current = self.head
            current_pos = 0
            while(current_pos < pos) and current.next:
                prev = current
                current = current.next
                current_pos +=1
            prev.next = node
            node.next = current
        return self.head
    def hapus(self, posisi):
        if self.head == None:
```



```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
(12, 'Apakah ada dalam data?')
True
(29, 'Apakah ada dalam data?')
False
>>>
```

```
        if self.head == None:
            return
        temp = self.head
        if posisi == 0:
            self.head = temp.next
            temp = None
            return
        for i in range(posisi -1 ):
            temp = temp.next
            if temp is None:
                break
        if temp is None:
            return
        if temp.next is None:
            return
        next = temp.next.next
        temp.next = None
        temp.next = next
    def cari(self, x):
        current = self.head
        while current != None:
            if current.data == x:
                print(x, "Apakah ada dalam data?")
                return True
            current = current.next
        print(x, "Apakah ada dalam data?")
        return False


a = LinkedList()
a.tambahDepan(31)
a.tambahDepan(12)
a.tambahDepan(23)
a.tambahDepan(14)
a.tambahDepan(2)
a.tambahDepan(19)
a.tambahAkhir(9)
a.hapus(0)
a.tambah(3,5)
print(a.cari(12))
print(a.cari(29))
```

Nomer 4.



```
Python 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
>>> b = DoublyLinkedList()
>>> b.awal(8)
('menambah pada awal', 8)
>>> b.awal(1)
('menambah pada awal', 1)
>>> b.akhir(7)
('menambah pada akhir', 7)
>>> b.akhir(3)
('menambah pada akhir', 3)
>>>
KeyboardInterrupt
>>> b.printList(b.head)

Dari Depan :
  1
  8
  7
  3

Dari Belakang :
  3
  7
  8
  1
>>>
```

```
#Nomor 4
class Node:
    def __init__(self, data):
        self.data = data
        self.prev = None
class DoublyLinkedList:
    def __init__(self):
        self.head = None
    def awal(self, new_data):
        print("menambah pada awal", new_data)
        new_node = Node(new_data)
        new_node.next = self.head
        if self.head is not None:
            self.head.prev = new_node
        self.head = new_node
    def akhir(self, new_data):
        print("menambah pada akhir", new_data)
        new_node = Node(new_data)
        new_node.next = None
        if self.head is None:
            new_node.prev = None
            self.head = new_node
            return
        last = self.head
        while(last.next is not None):
            last = last.next
        last.next = new_node
        new_node.prev = last
        return
    def printList(self, node):
        print("\nDari Depan :")
        while(node is not None):
            print(" % d" %(node.data))
            last = node
            node = node.next
        print("\nDari Belakang :")
        while(last is not None):
            print(" % d" %(last.data))
            last = last.prev
```