Muhammad Fadhil Bariz Ardanto
L200180200 / Kelas G

Modul 3

Tugas

No.1

1.py - C:\Users\asus\Desktop\Coolyeah\01_Algortima Struktur Data\PRAKTIKUM\pertemuan3\1.py (...

File Edit Format Run Options Window Help

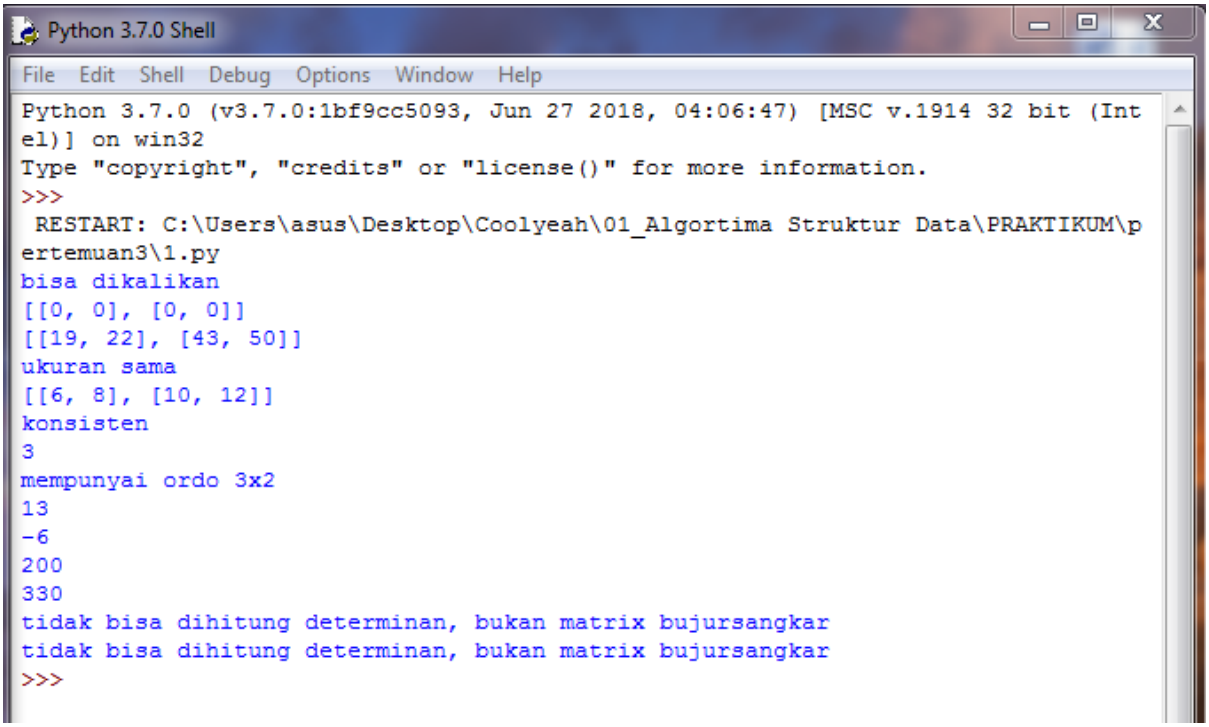
```
a = [[1,2],[3,4]]
b = [[5,6],[7,8]]
c = [[12,3,"y"],[12,33,4]]
d = [[3,4],[2,4],[1,5]]
e = [[5,6,7],[7,8,9]]
f = [[2,3],[4,5,6],[7,8,9]]
#No.1a
def cekKonsisten(n):
    x = len(n[0])
    z = 0
    for i in range(len(n)):
        if (len(n[i]) == x):
            z += 1

    if (z == len(n)):
        print("konsisten")
    else:
        print("tidak konsisten")
#No.1b
def ordo(n):
    x,y = 0,0
    for i in range(len(n)):
        x+=1
        y = len(n[i])
    print(len(n))
    print("mempunyai ordo "+str(x)+"x"+str(y))
```

```python
#No.1c
def jumlah(n,m):
    x,y = 0,0
    for i in range(len(n)):
        x+=1
        y = len(n[i])
    xy = [[0 for j in range(x)] for i in range(y)]

    z = 0
    if(len(n)==len(m)):
        for i in range(len(n)):
            if(len(n[i]) == len(m[i])):
                z+=1
    if(z==len(n) and z==len(m)):
        print("ukuran sama")
        for i in range(len(n)):
            for j in range(len(n[i])):
                xy[i][j] = n[i][j] + m[i][j]
        print(xy)
    else:
        print("ukuran beda")
```

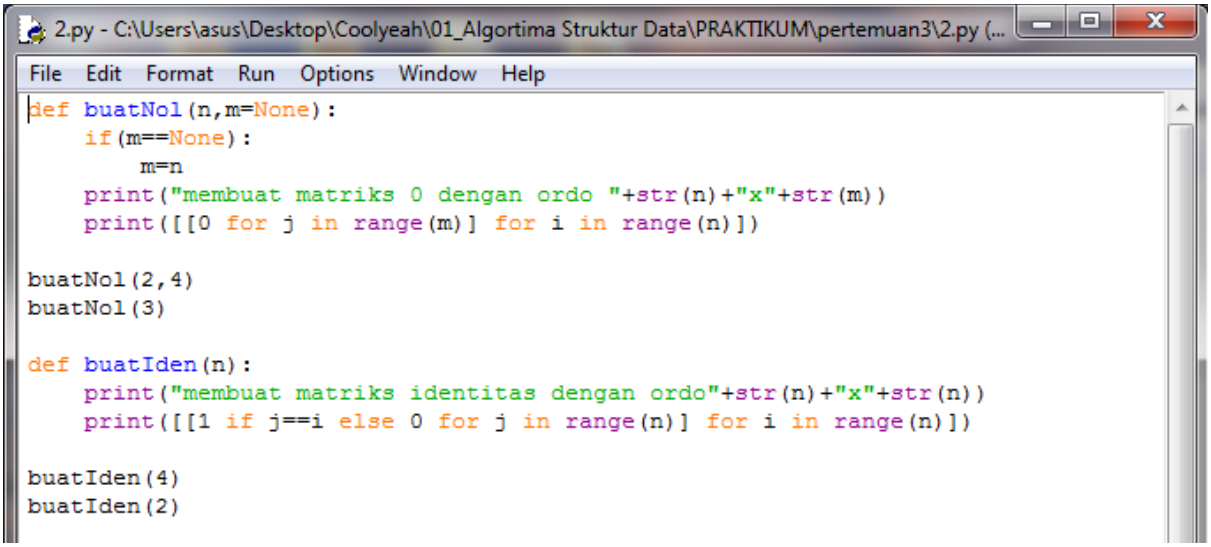
```python
#No.1d
def kaliMatrix(n,m):
    aa = 0
    x,y = 0,0
    for i in range(len(n)):
        x+=1
        y = len(n[i])
    v,w = 0,0
    for i in range(len(m)):
        v+=1
        w = len(m[i])

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

    else:
        print("tidak memenuhi syarat")
```

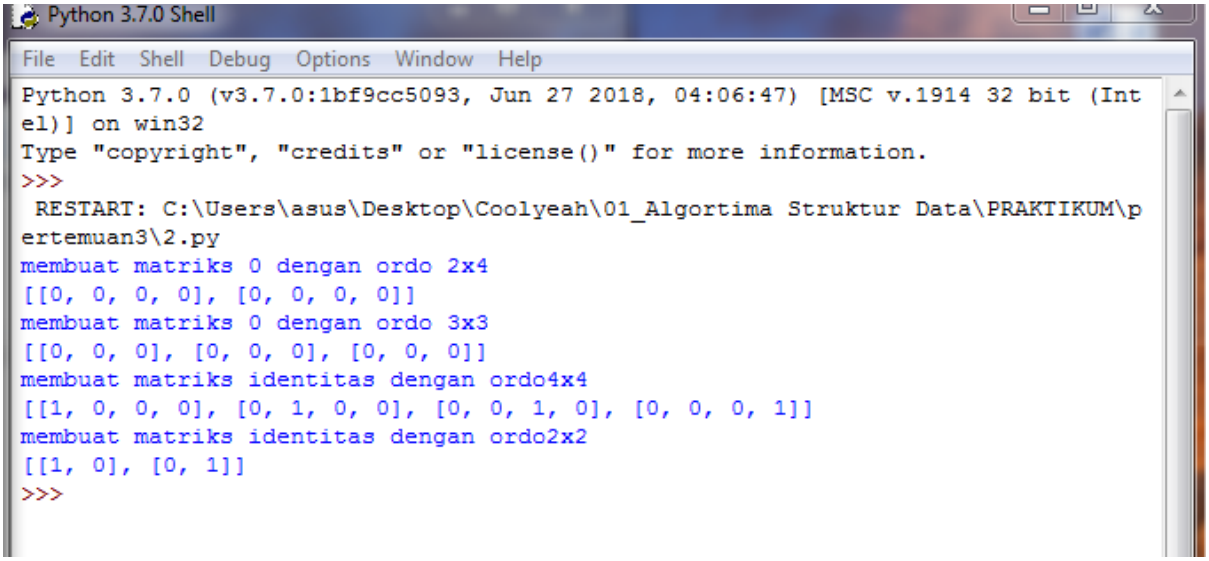
```python
#No.1e
def determBujursangkar(A, total=0):
    x = len(A[0])
    z = 0
    for i in range(len(A)):
        if (len(A[i]) == x):
            z+=1
    if(z == len(A)):
        if(x==len(A)):
            indices = list(range(len(A)))
            if len(A) == 2 and len(A[0]) == 2:
                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 = determBujursangkar(As)
                total += sign * A[0][fc] * sub_det
        else:
            return "tidak bisa dihitung determinan, bukan matrix bujursangkar"
    else:
        return "tidak bisa dihitung determinan, bukan matrix bujursangkar"
    return total
```

```python
z = [[3,1],[2,5]]
x = [[1,2,1],[3,3,1],[2,1,2]]
v = [[1,-2,0,0],
     [3,2,-3,1],
     [4,0,5,1],
     [2,3,-1,4]]
r = [[10,23,45,12,13],
     [1,2,3,4,5],
     [1,2,3,4,6],
     [4,2,3,4,8],
     [1,4,5,6,10]]



kaliMatrix(a,b)
jumlah(a,b)
cekKonsisten(a)
ordo(d)
print(determBujursangkar(z))
print(determBujursangkar(x))
print(determBujursangkar(v))
print(determBujursangkar(r))
print(determBujursangkar(d))
print(determBujursangkar(e))
```

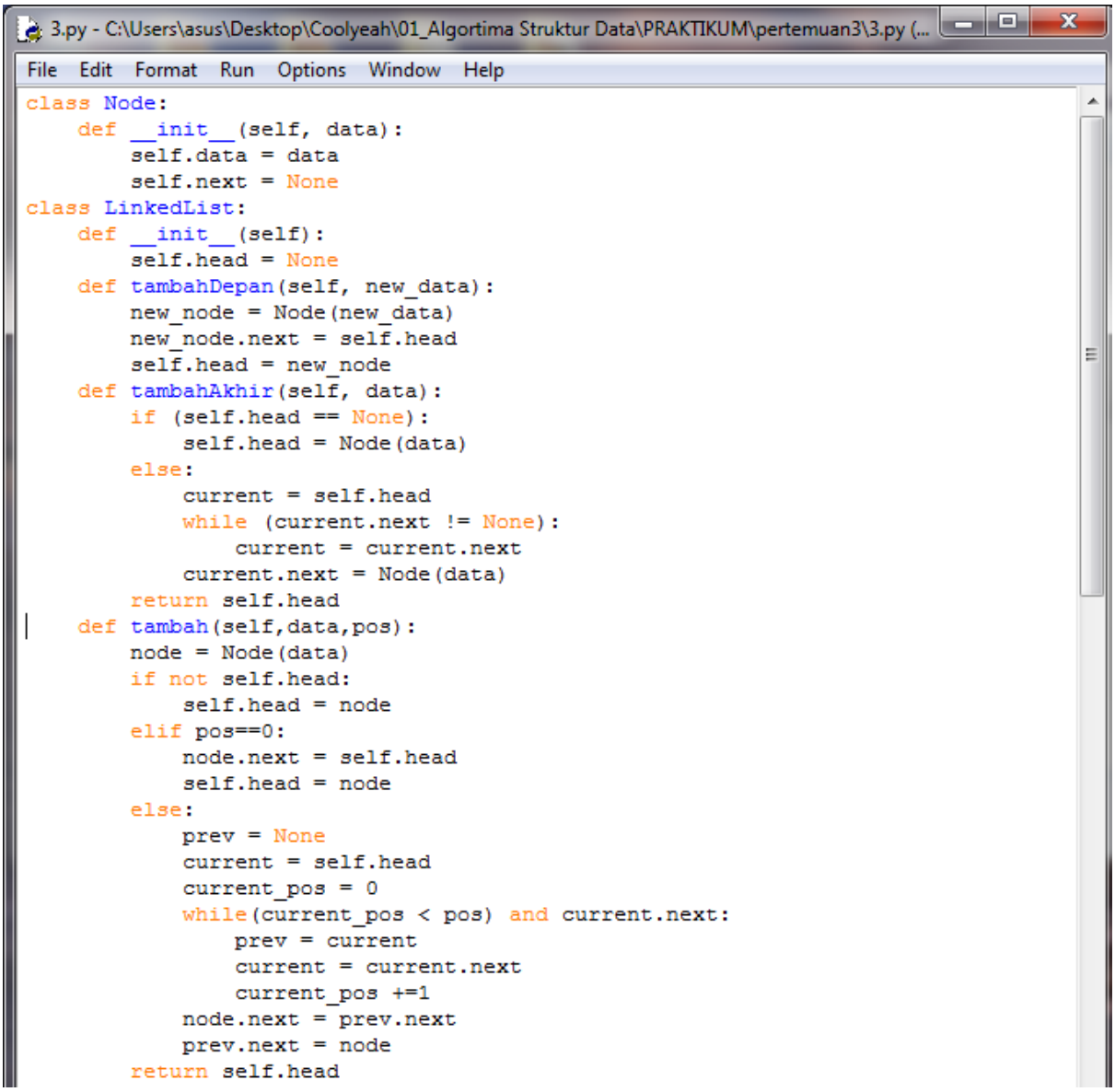

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\asus\Desktop\Coolyeah\01_Algortima Struktur Data\PRAKTIKUM\pertemuan3\1.py
bisa dikalikan
[[0, 0], [0, 0]]
[[19, 22], [43, 50]]
ukuran sama
[[6, 8], [10, 12]]
konsisten
3
mempunyai ordo 3x2
13
-6
200
330
tidak bisa dihitung determinan, bukan matrix bujursangkar
tidak bisa dihitung determinan, bukan matrix bujursangkar
>>>

No.2


2.py - C:\Users\asus\Desktop\Coolyeah\01_Algortima Struktur Data\PRAKTIKUM\pertemuan3\2.py (...
File Edit Format Run Options Window Help
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)])
buatNol(2,4)
buatNol(3)
def buatIden(n):
print("membuat matriks identitas dengan ordo"+str(n)+"x"+str(n))
print([[1 if j==i else 0 for j in range(n)] for i in range(n)])
buatIden(4)
buatIden(2)

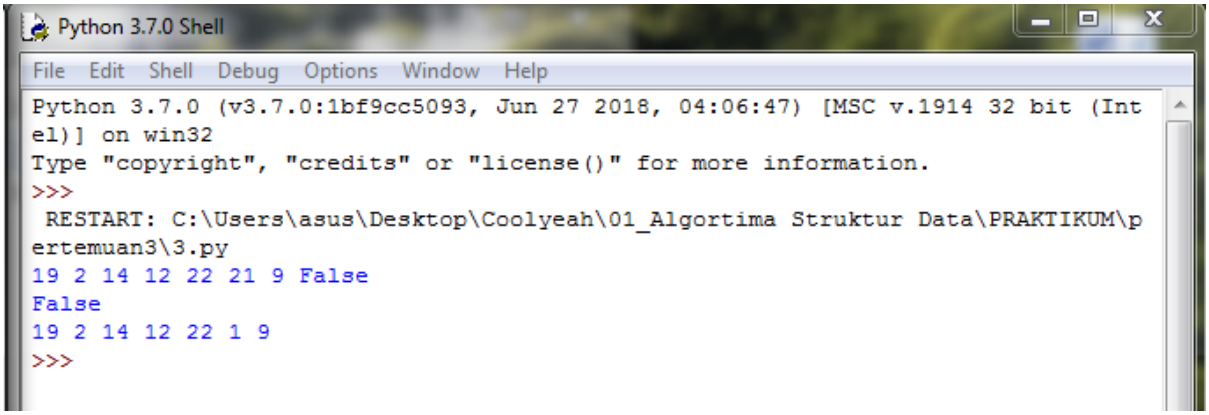

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\asus\Desktop\Coolyeah\01_Algortima Struktur Data\PRAKTIKUM\pertemuan3\2.py
membuat matriks 0 dengan ordo 2x4
[[0, 0, 0, 0], [0, 0, 0, 0]]
membuat matriks 0 dengan ordo 3x3
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
membuat matriks identitas dengan ordo4x4
[[1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], [0, 0, 0, 1]]
membuat matriks identitas dengan ordo2x2
[[1, 0], [0, 1]]
>>>

No.3


3.py - C:\Users\asus\Desktop\Coolyeah\01_Algortima Struktur Data\PRAKTIKUM\pertemuan3\3.py (...
File Edit Format Run Options Window Help
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
node.next = prev.next
prev.next = node
return self.head

```
    def hapusNode(self, position):
        if self.head == None:
            return
        temp = self.head
        if position == 0:
            self.head = temp.next
            temp = None
            return
        for i in range(position ):
            prev = temp
            temp = temp.next
            if temp is None:
                break
        if temp is None:
            return
        if temp.next is None:
            return
        prev.next = temp.next
        temp= None

    def cari(self, x):
        current = self.head
        while current != None:
            if current.data == x:
                return "True"
            current = current.next
        return "False"
    def display(self):
        current = self.head
        while current is not None:
            print(current.data, end = ' ')
            current = current.next
```

```
llist = LinkedList()
llist.tambahDepan(21)
llist.tambahDepan(22)
llist.tambahDepan(12)
llist.tambahDepan(14)
llist.tambahDepan(2)
llist.tambahDepan(19)
llist.tambahAkhir(9)
llist.display()
llist.hapusNode(5)
llist.tambah(1,5)
print(llist.cari(21))
print(llist.cari(29))
llist.display()
```



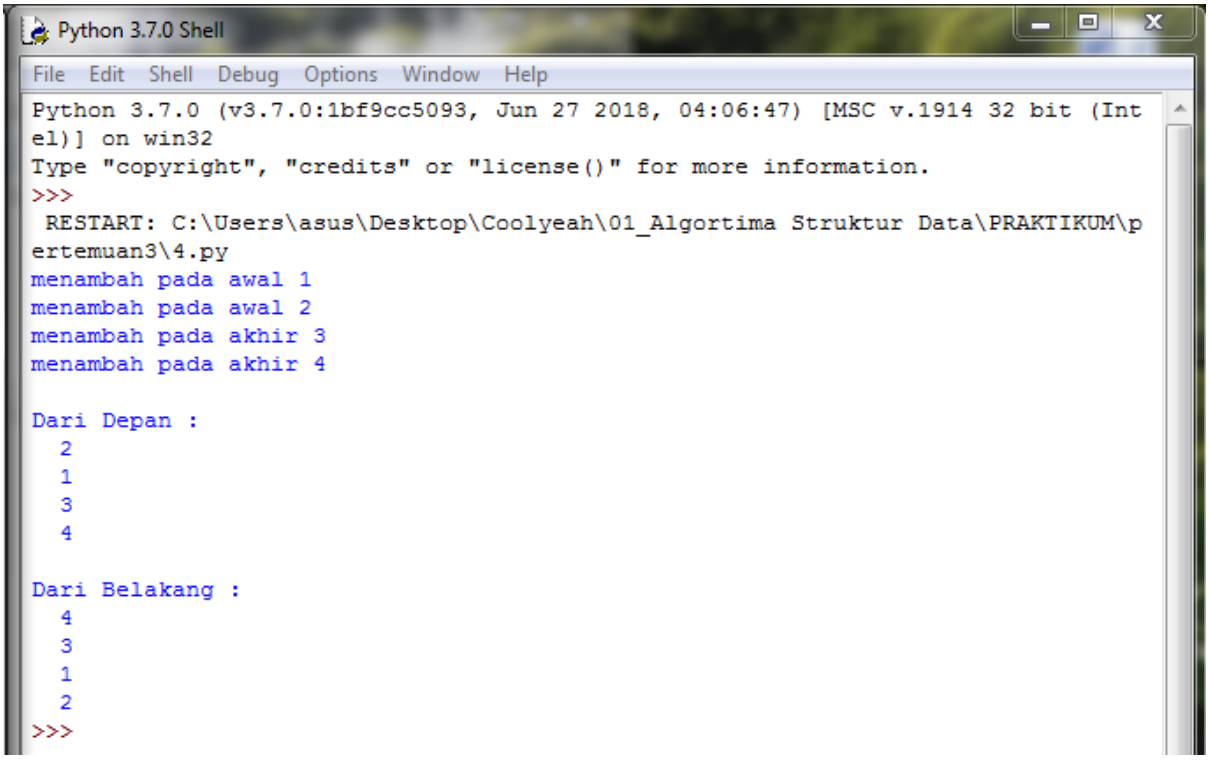
No.4

4.py - C:\Users\asus\Desktop\Coolyeah\01_Algortima Struktur Data\PRAKTIKUM\pertemuan3\4.py (...

File Edit Format Run Options Window Help

```
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
```

```
llist = DoublyLinkedList()
llist.awal(1)
llist.awal(2)
llist.akhir(3)
llist.akhir(4)
llist.printList(llist.head)
```


Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\asus\Desktop\Coolyeah\01_Algortima Struktur Data\PRAKTIKUM\pertemuan3\4.py
menambah pada awal 1
menambah pada awal 2
menambah pada akhir 3
menambah pada akhir 4
Dari Depan :
2
1
3
4
Dari Belakang :
4
3
1
2
>>>