LAPORAN PRAKTIKUM
ALGORITMA DAN STRUKTUR DATA
MODUL 3

DISUSUN OLEH :

NIM	**L200184040**
NAMA	**AQSHAL FATWA IBRAHIM**
KELAS	**A**

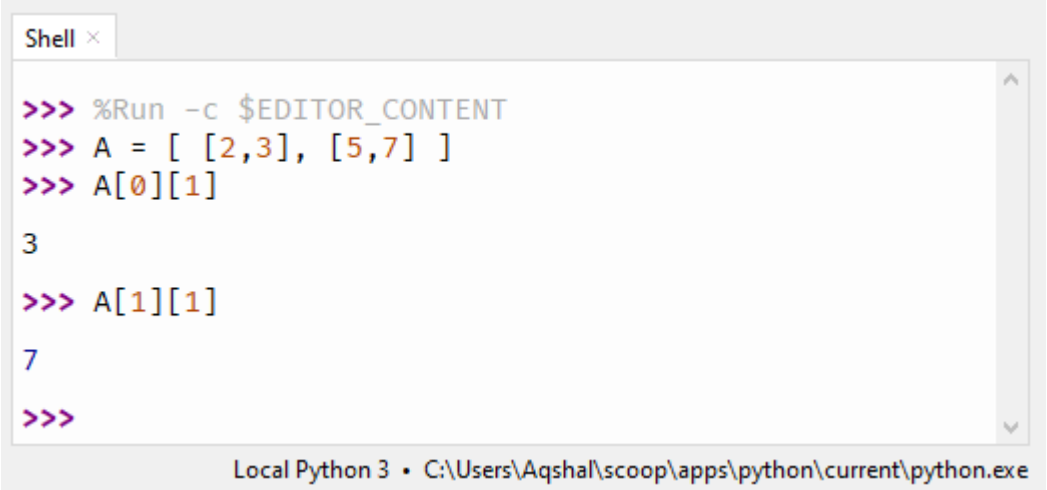
PROGRAM STUDI INFORMATIKA

FAKULTAS KOMUNIKASI DAN INFORMATIKA

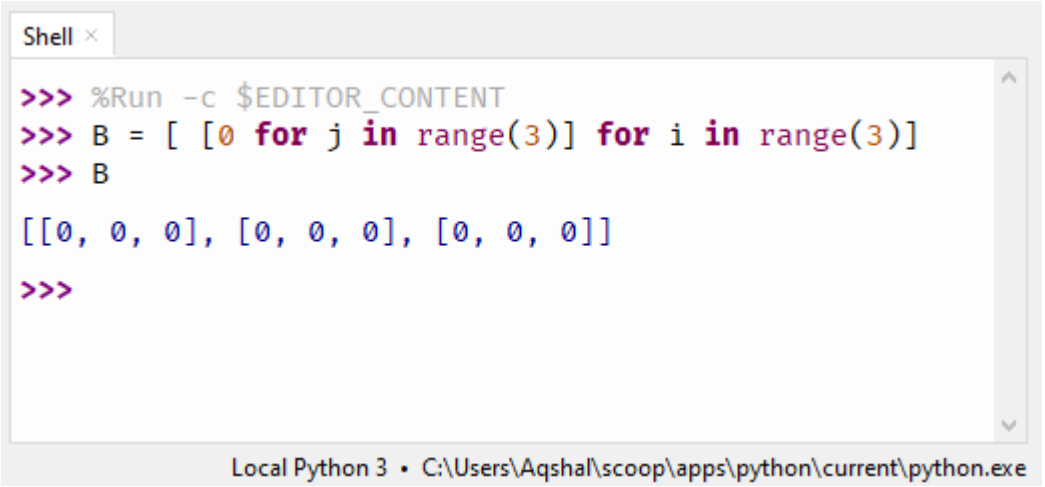
UNIVERSITAS MUHAMMADIYAH SURAKARTA

LATIHAN

3.1

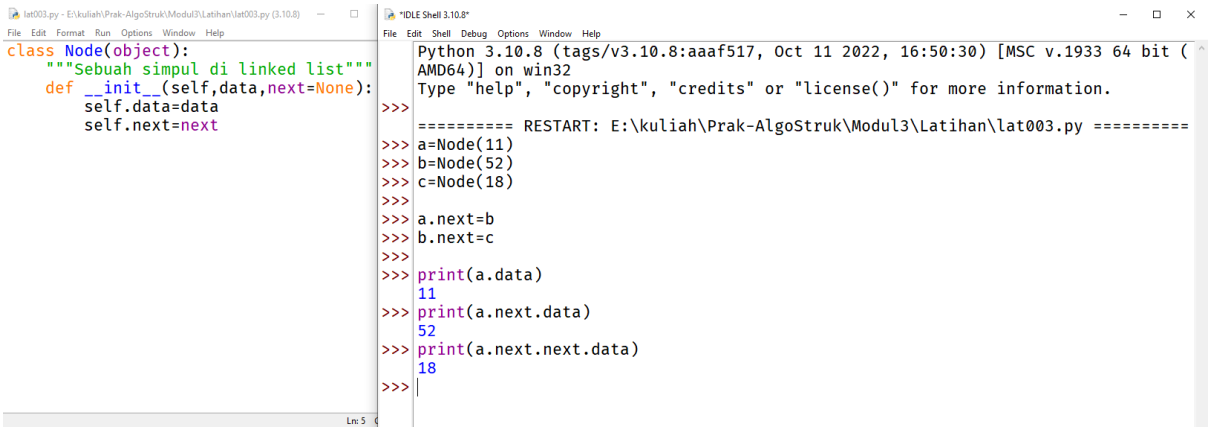


3.2



3.3

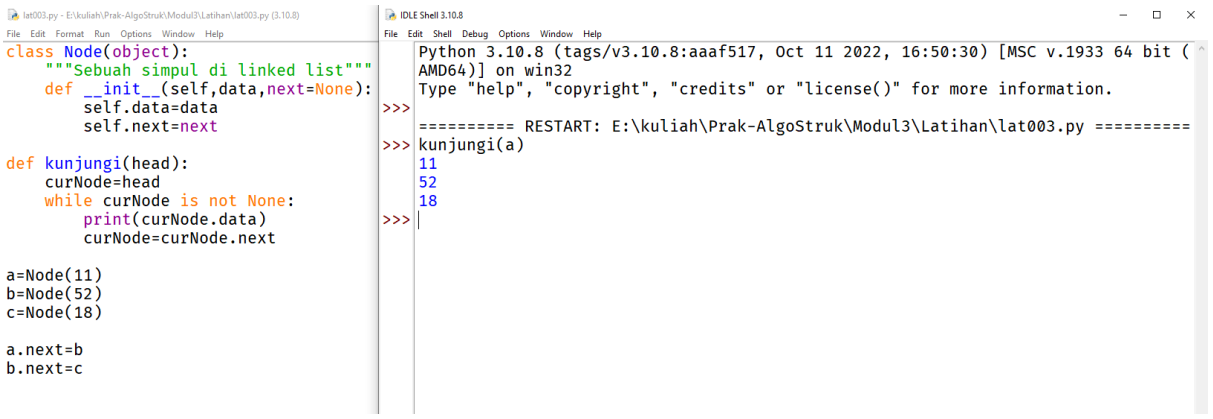
Linked List



```
class Node(object):
    """Sebuah simpul di linked list"""
    def __init__(self,data,next=None):
        self.data=data
        self.next=next
```

```
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: E:\kuliah\Prak-AlgoStruk\Modul3\Latihan\lat003.py ==========
>>> a=Node(11)
>>> b=Node(52)
>>> c=Node(18)
>>>
>>> a.next=b
>>> b.next=c
>>>
>>> print(a.data)
11
>>> print(a.next.data)
52
>>> print(a.next.next.data)
18
>>>
```

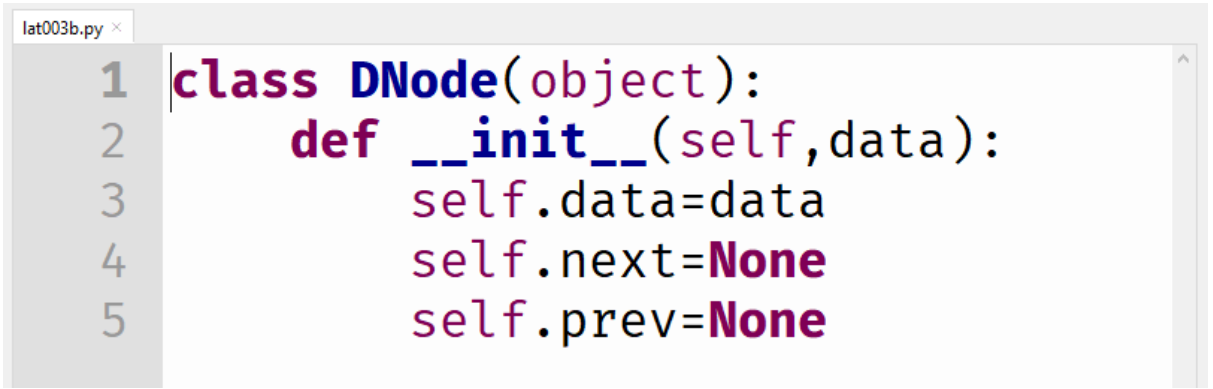
Mengunjungi Setiap Simpul dari Depan



```
class Node(object):
    """Sebuah simpul di linked list"""
    def __init__(self,data,next=None):
        self.data=data
        self.next=next

def kunjungi(head):
    curNode=head
    while curNode is not None:
        print(curNode.data)
        curNode=curNode.next

a=Node(11)
b=Node(52)
c=Node(18)

a.next=b
b.next=c
```

```
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: E:\kuliah\Prak-AlgoStruk\Modul3\Latihan\lat003.py ==========
>>> kunjungi(a)
11
52
18
>>>
```

Advanced Linked List



lat003b.py

```
class DNode(object):
    def __init__(self,data):
        self.data=data
        self.next=None
        self.prev=None
```

Soal-soal untuk Mahasiswa

1.

no1.py - E:\kuliah\Prak-AlgoStruk\Modul3\Soal\no1.py (3.10.8)

File Edit Format Run Options Window Help

```
# 1.
def apakahKonsisten(matrix):
    if len(matrix) == 0:
        return True
    else:
        for i in range(len(matrix)):
            if len(matrix[i]) != len(matrix[0]):
                return False
        return True

# 2.
def ukuran(matrix):
    if apakahKonsisten(matrix):
        return (len(matrix), len(matrix[0]))
    else:
        return False

# 3.
def jumlah(matrix1, matrix2):
    if ukuran(matrix1) == ukuran(matrix2):
        hasil = []
        for i in range(len(matrix1)):
            hasil.append([])
            for j in range(len(matrix1[i])):
                hasil[i].append(matrix1[i][j] + matrix2[i][j])
        return hasil
    else:
        return False

# 4.
def kali(matrix1, matrix2):
    if ukuran(matrix1)[1] == ukuran(matrix2)[0]:
        hasil = []
        for i in range(len(matrix1)):
            hasil.append([])
            for j in range(len(matrix2[0])):
                hasil[i].append(0)
                for k in range(len(matrix1[i])):
                    hasil[i][j] += matrix1[i][k] * matrix2[k][j]
        return hasil
    else:
        return False

# 5.
def determinan(matrix):
    if ukuran(matrix)[0] == ukuran(matrix)[1]:
        if len(matrix) == 1:
            return matrix[0][0]
        else:
            hasil = 0
            for i in range(len(matrix)):
                hasil += matrix[0][i] * (-1)**(i) * determinan(submatrix(matrix, 0, i))
            return hasil
    else:
        return False

```

Ln: 44 Col: 4

2.

no2.py - E:\kuliah\Prak-AlgoStruk\Modul3\Soal\no2.py (3.10.8)

File Edit Format Run Options Window Help

```
# 1.
def buatNol(m, n=None):
    if n == None:
        n = m
    return [[0 for j in range(n)] for i in range(m)]

# 2.
def buatIdentitas(n):
    return [[1 if j == i else 0 for j in range(n)] for i in range(n)]
```

Ln: 9 Col: 69

3.

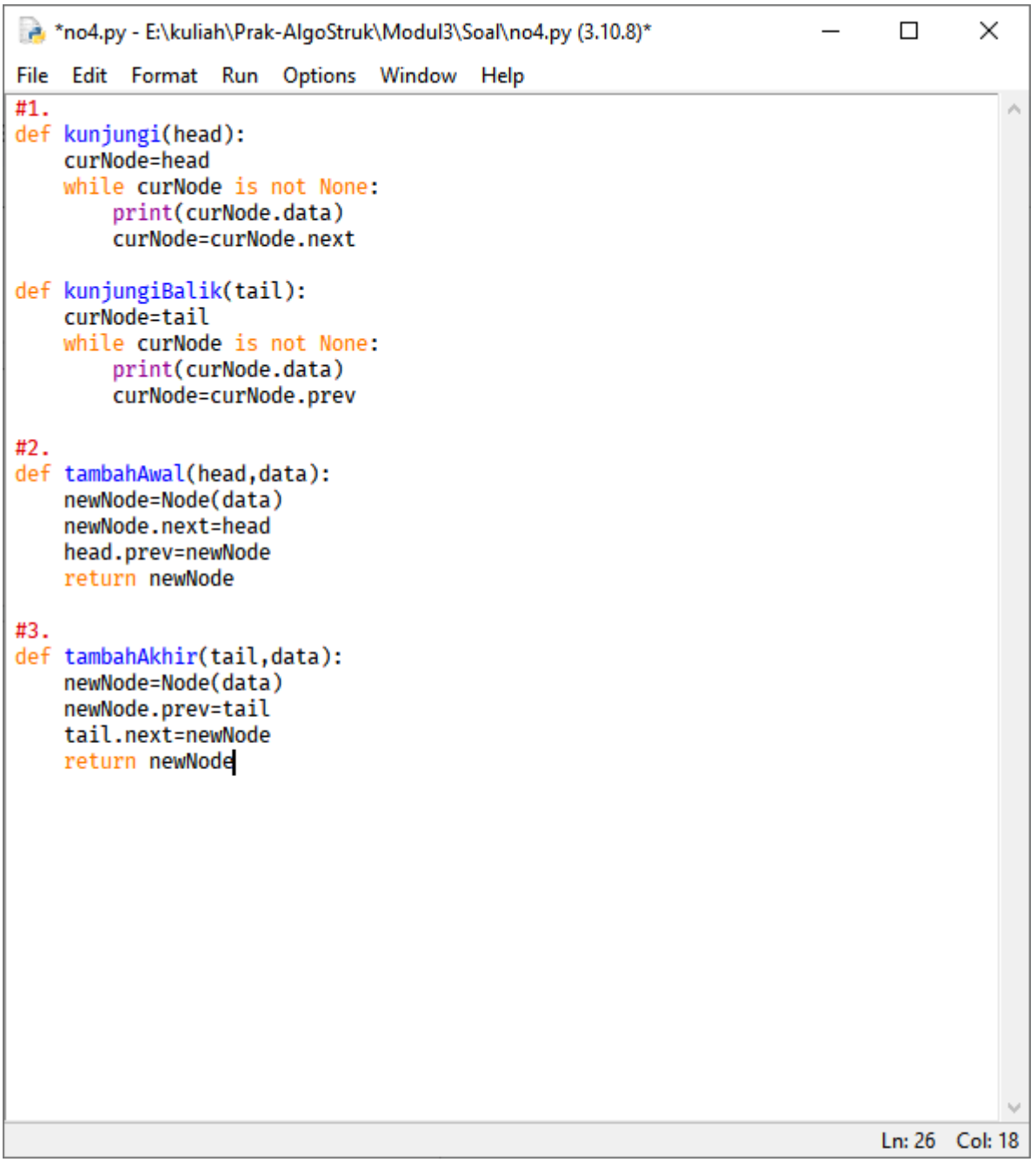
no3.py - E:\kuliah\Prak-AlgoStruk\Modul3\Soal\no3.py (3.10.8)

File Edit Format Run Options Window Help

```
# 1.
def cari(head,yang_dicari):
    curNode=head
    while curNode is not None:
        if curNode.data==yang_dicari:
            return True
        curNode=curNode.next
    return False

# 2.
def tambahDepan(head,data):
    newNode=Node(data)
    newNode.next=head
    return newNode

# 3.
def tambahAkhir(head,data):
    newNode=Node(data)
    curNode=head
    while curNode.next is not None:
        curNode=curNode.next
    curNode.next=newNode
    return head

# 4.
def tambah(head,posisi,data):
    newNode=Node(data)
    curNode=head
    while curNode.data!=posisi:
        curNode=curNode.next
    newNode.next=curNode.next
    curNode.next=newNode
    return head

# 5.
def hapus(head,posisi):
    curNode=head
    if curNode.data==posisi:
        head=curNode.next
        return head
    while curNode.next.data!=posisi:
        curNode=curNode.next
    curNode.next=curNode.next.next
    return head
```

Ln: 44 Col: 15

4.


no4.py - E:\kuliah\Prak-AlgoStruk\Modul3\Soal\no4.py (3.10.8)
File Edit Format Run Options Window Help
#1.
def kunjungi(head):
curNode=head
while curNode is not None:
print(curNode.data)
curNode=curNode.next
def kunjungiBalik(tail):
curNode=tail
while curNode is not None:
print(curNode.data)
curNode=curNode.prev
#2.
def tambahAwal(head,data):
newNode=Node(data)
newNode.next=head
head.prev=newNode
return newNode
#3.
def tambahAkhir(tail,data):
newNode=Node(data)
newNode.prev=tail
tail.next=newNode
return newNode
Ln: 26 Col: 18