

Modul 3

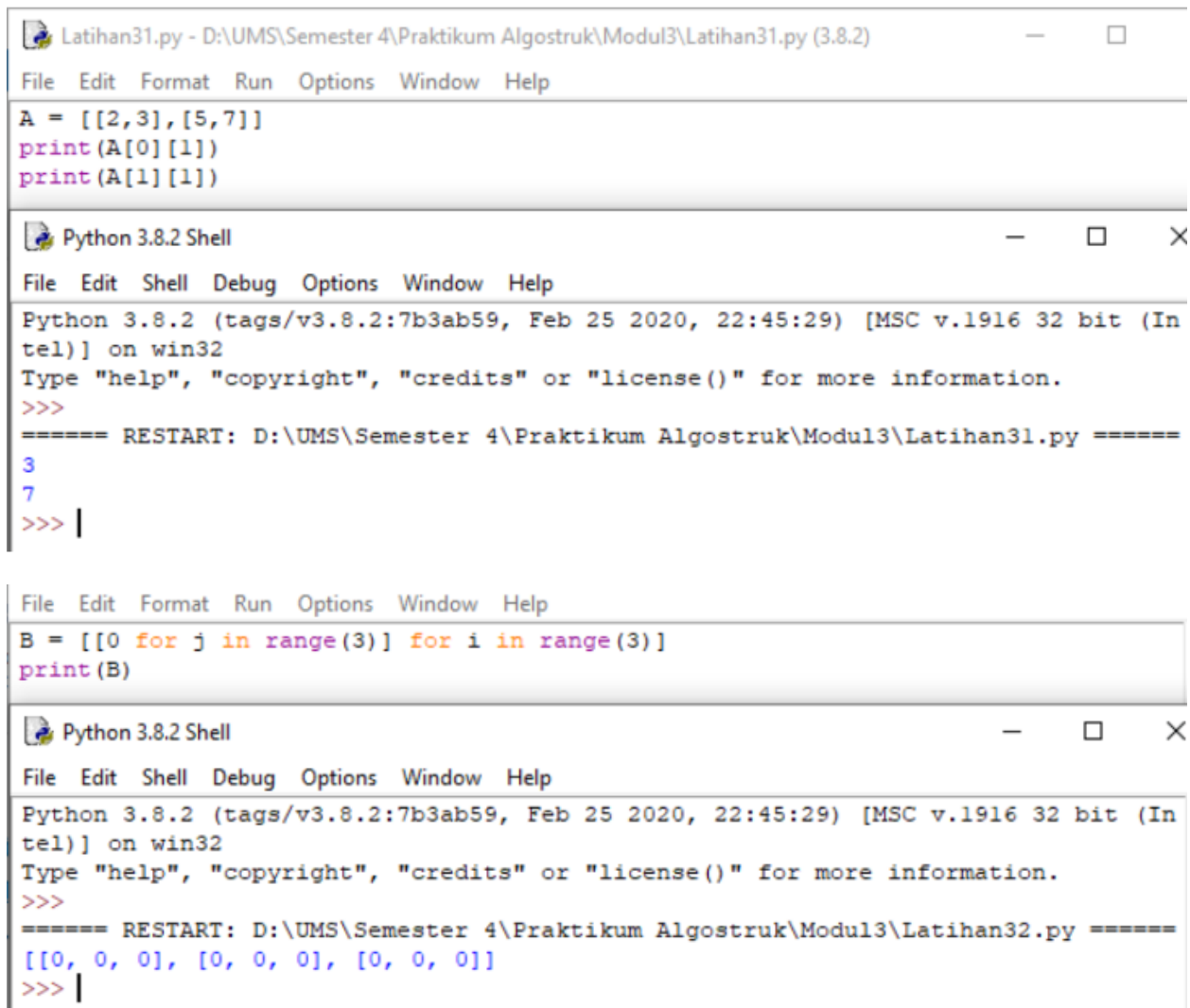
Algoritma dan Struktur Data

Nama : Damar Fatika Sari

NIM : L200180126

Kelas : E

Kegiatan Praktikum



The image shows two screenshots of a Python IDE. The first screenshot shows a script named 'Latihan31.py' with the following code:

```
A = [[2,3],[5,7]]
print(A[0][1])
print(A[1][1])
```

The second screenshot shows the 'Python 3.8.2 Shell' window for the same script, displaying the output:

```
>>>
===== RESTART: D:\UMS\Semester 4\Praktikum Algostruk\Modul3\Latihan31.py =====
3
7
>>> |
```

The third screenshot shows another script with the following code:

```
B = [[0 for j in range(3)] for i in range(3)]
print(B)
```

The fourth screenshot shows the 'Python 3.8.2 Shell' window for this script, displaying the output:

```
>>>
===== RESTART: D:\UMS\Semester 4\Praktikum Algostruk\Modul3\Latihan32.py =====
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
>>> |
```

<pre> File Edit Format Run Options Window Help class Node(object): """Sebuah simpul di linked list""" def __init__(self, data, next=None): self.data = data self.next = next a = Node(11) b = Node(52) c = Node(18) a.next = b b.next = c print(a.data) print(a.next.data) print(a.next.next.data) </pre>	<pre> File Edit Shell Debug Options Window Help Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, tel)] on win32 Type "help", "copyright", "credits" or "license() >>> ===== RESTART: D:\UMS\Semester 4\Praktikum Algc 11 52 18 >>> </pre>
--	--

<pre> File Edit Format Run Options Window Help 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) </pre>	<pre> File Edit Shell Debug Options Window Help Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45: tel)] on win32 Type "help", "copyright", "credits" or "license()" for >>> ===== RESTART: D:\UMS\Semester 4\Praktikum Algostruk\ 11 52 18 >>> </pre>
--	--

<pre> File Edit Format Run Options Window Help class DNode(object): def __init__(self,data): self.data = data self.next = None self.prev = None a = DNode(11) b = DNode(52) c = DNode(18) a.next = b b.next = c c.prev = b b.prev = a print(a.data) print(b.next.data) print(a.next.data) print(c.prev.data) </pre>	<pre> File Edit Shell Debug Options Window Help Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45: tel)] on win32 Type "help", "copyright", "credits" or "license()" for >>> ===== RESTART: D:\UMS\Semester 4\Praktikum Algostruk\ 11 18 52 52 >>> </pre>
--	---

ListComprehension.py - D:\UMS\Semester 4\Praktikum Algostruk\Modul3\ListComprehensi...	Python 3.8.2 Shell
<pre>File Edit Format Run Options Window Help A = [x**2 for x in range(0,7)] print(A) B = [(x,x**2) for x in range(7)] print(B) C = [x**2 for x in range(15) if x%2==0] print(C) D = [3 for i in range(5)] print(D) E = [[0 for j in range(3)] for i in range(3)] print(E) F = [[1 if j ==1 else 0 for j in range (3)] for i in range(3)] print(F) d = "Yogyakarta dan Surakarta" G = [x for x in d if x in "aiueoAIUEO"] print(G) # H = [x for x in range (20,50) if apakahPrima(x)] # print(H)</pre>	<pre>File Edit Shell Debug Options Window Help Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MS tel)] on win32 Type "help", "copyright", "credits" or "license()" for more i >>> == RESTART: D:\UMS\Semester 4\Praktikum Algostruk\Modul3\List [0, 1, 4, 9, 16, 25, 36] [(0, 0), (1, 1), (2, 4), (3, 9), (4, 16), (5, 25), (6, 36)] [0, 4, 16, 36, 64, 100, 144, 196] [3, 3, 3, 3, 3] [[0, 0, 0], [0, 0, 0], [0, 0, 0]] [[0, 1, 0], [0, 1, 0], [0, 1, 0]] ['o', 'a', 'a', 'a', 'a', 'u', 'a', 'a', 'a'] >>> </pre>

Tugas

1.

ListComprehension.py - D:\UMS\Semester 4\Praktikum Algostruk\Modul3\ListComprehensi...	Python 3.8.2 Shell
<pre>File Edit Format Run Options Window Help A = [[1,2],[3,4],[5,6]] B = [[7,8],[9,10]] C = [[3,6],[5,2]] #Nomor 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)) #Nomor 1B def ordo (matriks): return ("Ordo matriks = "+str(len(matriks))+ " x "+str(len(matriks[0])))</pre>	<pre>File Edit Shell Debug Options Window He Python 3.8.2 (tags/v3.8.2:7b3ab59, tel)] on win32 Type "help", "copyright", "credits" >>> ===== RESTART: D:/UMS/Semester 4, [1, 2] [3, 4] [5, 6] matriks tidak konsisten None [7, 8] [9, 10] matriks konsisten None >>> </pre>

```
#Nomor 1B
def ordo(matriks):
    return ("Ordo matriks = "+str(len(matriks))+ " x "+str(len(matriks[0])))
```

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/UMS/Semester 4/Praktikum Algostruk/Modul3/Nomor1.py =====
>>> ordo(A)
'Ordo matriks = 3 x 2'
>>> ordo(B)
'Ordo matriks = 2 x 2'
>>> ordo(C)
'Ordo matriks = 2 x 2'
>>>
```

```
#Nomor 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")
```

```
>>>
===== RESTART: D:/UMS/Sem
>>> Jumlah(A,B)
Matriks tidak sesuai
>>> Jumlah(B,C)
10
14
14
12
>>> |
```

```
#Nomor 1D
def kali(m,n):
    a = 0
    x,y = 0,0
    for i in range(len(m)):
        x += 1
        y = len(m[i])
    v,w = 0,0
    for i in range(len(n)):
        v += 1
        w = len(n[i])

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

kali(A,B)
kali(B,C)
```

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/UMS/Semester 4/Pra
Bisa Dikalikan
[[25, 28], [57, 64], [89, 100]]
Bisa Dikalikan
[[61, 58], [77, 74]]
>>> |
```

```

#Nomor 1E
def determinan(p, total = 0):
    x = len(p[0])
    z = 0
    for i in range(len(p)):
        if (len(p[i]) == x):
            z += 1
    if (z == len(p)):
        if (x == len(p)):
            indices = list(range(len(p)))
            if len(p) == 2 and len(p[0]) == 2:
                val = p[0][0] * p[1][1] - p[1][0] * p[0][1]
                return val
            for fc in indices:
                pq = p
                pq = pq[1:]
                height = len(pq)
                for i in range(height):
                    pq[i] = pq[i][0:fc] + pq[i][fc+1:]
                sign = (-1) ** (fc % 2)
                sub_det = determinanHitung(pq)
                total += sign * A[0][fc] * sub_det
            else:
                return "Tidak bisa dihitung, bukan matriks bujur sangkar"
        else:
            return "Tidak bisa dihitung, bukan matriks bujur sangkar"
    return total

```

```

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:4
tel)] on win32
Type "help", "copyright", "credits" or "license()" f
>>>
===== RESTART: D:\UMS\Semester 4\Praktikum Algost.
>>> determinan(A)
'Tidak bisa dihitung, bukan matriks bujur sangkar'
>>> determinan(B)
-2
>>> determinan(C)
-24
>>> |

```

2.

```
File Edit Format Run Options Window Help
```

```

#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)])

```

```
Python 3.8.2 Shell
```

```
File Edit Shell Debug Options Window Help
```

```

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (In
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/UMS/Semester 4/Praktikum Algostruk/Modul3/Nomor2.py =====
>>> buatNol(2,4)
Membuat matriks 0 dengan ordo 2 x 4
[[0, 0, 0, 0], [0, 0, 0, 0]]
>>> buatNol(3)
Membuat matriks 0 dengan ordo 3 x 3
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
>>> |

```

```
#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)
```

Python 3.8.2 Shell

File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/UMS/Semester 4/Praktikum Algostruk/Modul3/Nomor2.py =====
>>> buatIdentitas(4)
Membuat matriks identitas dengan ordo 4 x 4
[[1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], [0, 0, 0, 1]]
>>> buatIdentitas(2)
Membuat matriks identitas dengan ordo 2 x 2
[[1, 0], [0, 1]]

3.

File Edit Format Run Options Window Help

```
#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:
            return
        if posisi == 0:
            temp = self.head
            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
    def display(self):
        current = self.head
        while current is not None:
            print(current.data, end = ' ')
            current = current.next

A = LinkedList()
A.tambahDepan(31)
A.tambahDepan(12)
A.tambahDepan(23)
A.tambahAkhir(19)
A.hapus(0)
```



```

===== RESTART: D:\UMS\Semester 4\Praktikum Algostruk\Modul3\Nomor3.py =====
12 Apakah ada dalam data?
True
90 Apakah ada dalam data?
False
12 31 3 19
>>> |

```

Ln: 10 Col:

```

A = LinkedList()
A.tambahDepan(31)
A.tambahDepan(12)
A.tambahDepan(23)
A.tambahAkhir(19)
A.hapus(0)
A.tambah(3,5)
print(A.cari(12))
print(A.cari(90))
A.display()

```

4.

```

#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))

```

File Edit Format Run Options Window Help

```

d = DoublyLinkedList()
d.awal(8)
d.awal(1)
d.akhir(7)
d.akhir(3)
d.printList(d.head)

```

Python 3.8.2 Shell

File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020) on win32

Type "help", "copyright", "credits" or "license()" for more

>>>

```

===== RESTART: D:/UMS/Semester 4/Praktikum
Menambah pada awal 8
Menambah pada awal 1
Menambah pada akhir 7
Menambah pada akhir 3

```

Dari depan :

```

1
8
7
3

```

Dari belakang :

```

3
7
8
1

```

>>> |