NAMA : KEVIN AVICENNA WIDIARTO
NIM : L200200183
Modul : 3

Praktikum Algoritma Struktur Data

MODUL 3 ASD

NO 1

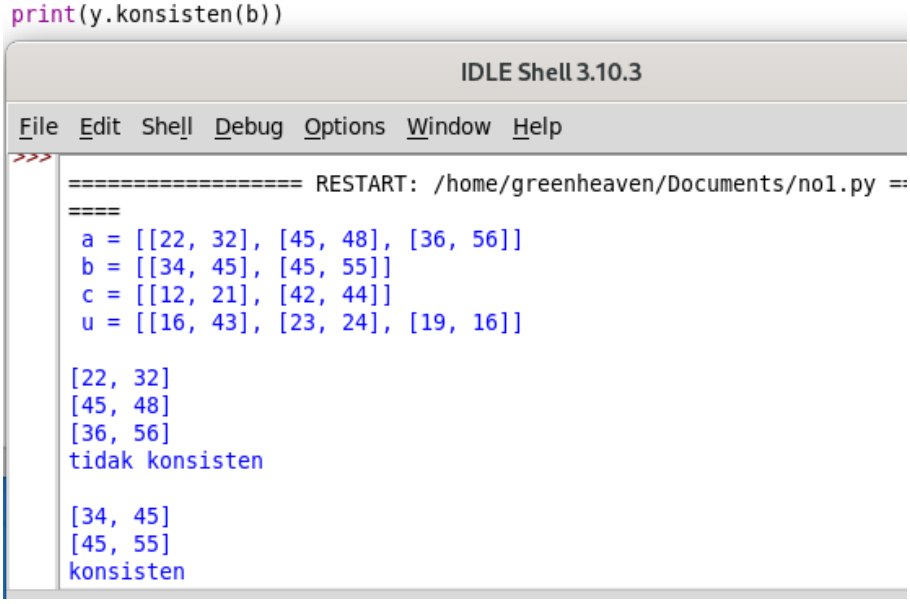
```
a = [[22,32],[45,48],[36,56]]
b = [[34,45],[45,55]]
c = [[12,21],[42,44]]
u = [[16,43],[23,24],[19,16]]
s = [[7,8],[9,10]]

class Matriks(object):
    def BPmatriks(self,m):
        for i in m:
            print(i)

    def konsisten(self,m):
        if len(m[0]) == len(m):
            return "konsisten"
        else:
            return "tidak konsisten"

print(" a =" ,a,"\n","b =" ,b,"\n","c =" ,c,"\n","u =" ,u,"\n")
m = Matriks()
m.BPmatriks(a)
print(m.konsisten(a),"\n")

y = Matriks()
y.BPmatriks(b)
print(y.konsisten(b))
```



NO 1B

no1b.py - /home/greenheaven/Documents/no1b.py (3.10

File Edit Format Run Options Window Help

```
from no1 import c

def Ordo(m):
        return("Ordo = "+str(len(m))+" x "+str(len(m[0])))
```

IDLE Shell 3.10.3

File Edit Shell Debug Options Window Help

```
Python 3.10.3 (main, Mar 18 2022, 00:00:00) [GCC 11.2.1 202
)] on linux
Type "help", "copyright", "credits" or "license()" for more
>>>
================= RESTART: /home/greenheaven/Documents/no1b
>>> print(c)
[[12, 21], [42, 44]]
>>> print(Ordo(c))
Ordo = 2 x 2
>>>
```

NO 1C

no1c.py - /home/greenheaven/Documents/no1c.py (3.10.3)

File Edit Format Run Options Window Help

```
from no1b import Ordo
from no1 import b,c


def Jumlah(m1,m2):
    if Ordo(m1) == Ordo(m2):
        for x in range(0, len(m1)):
            for y in range(0, len(m1[0])):
                print(m1[x][y] + m2[x][y],' '),
            print()
    else:
        print("Tidak tepat")
```

IDLE Shell 3.10.3

File Edit Shell Debug Options Window Help

```
================= RESTART: /home/greenheaven/Documents/no1c.py
>>> print(b,"\n",c)
[[34, 45], [45, 55]]
 [[12, 21], [42, 44]]
>>> Jumlah(b,c)
46
66

87
99

>>>
```

NO 1D

no1d.py - /home/greenheaven/Documents/no1d.py (3.10.3)

File Edit Format Run Options Window Help

```
from no1 import a,b,c,u,s

def kali(a,b):
    x,y = 0,0
    for i in range(len(a)):
        x+=1
        y = len(a[i])
    v,w = 0,0
    for i in range(len(b)):
        v+=1
        w = len(b[i])

    if(y==v):
        print("")
        x = [[0 for j in range(w)] for i in range(x)]
        for i in range(len(a)):
            for j in range(len(b[0])):
                for k in range(len(b)):
                    x[i][j] += a[i][k] * b[k][j]
        print(x)

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

IDLE Shell 3.10.3

File Edit Shell Debug Options Window Help

```
>>> kali(a,b)

[[2188, 2750], [3690, 4665], [3744, 4700]]
>>> kali(b,s)

[[643, 722], [810, 910]]
>>>
```

```
def determinan(Z, total=0):
    x = len([0])
    y = 0
    for i in range(len(Z)):
        if (len(Z[i]) == x):
            y+=1
    if(y == len(Z)):
        if(x==len(Z)):
            indices = list(range(len(Z)))
            if len(Z) == 2 and len(Z[0]) == 2:
                val = Z[0][0] * Z[1][1] - Z[1][0] * Z[0][1]
                return val
            for fc in indices:
                A = Z
                A = A[1:]
                height = len(A)
                for i in range(height):
                    A[i] = A[i][0:fc] + A[i][fc+1:]
                sign = (-1) ** (fc % 2)
                sub_det = determinan(A)
                total += sign * Z[0][fc] * sub_det
        else:
            return "tidak bisa dihitung karena bukan matrix bujursangkar"
    else:
        return "tidak bisa dihitung karena bukan matrix bujursangkar"
    return total
```

IDLE Shell 3.10.3

File Edit Shell Debug Options Window Help

```
>>> determinan(a)
'tidak bisa dihitung karena bukan matrix bujursangkar'
>>>
```

NO 2A

```
def buatNol(x,y=None):
    if(y==None):
        y=x
    print("Matriks 0 dan ordonya "+str(x)+"x"+str(y))
    print([[0 for j in range(y)] for i in range(x)])
```

```
>>>
================= RESTART: /home/greenheaven/Documents/no2a.py
>>> buatNol(3,4)
Matriks 0 dan ordonya 3x4
[[0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]]
>>>
```

NO 2B

```
def buatIdentitas(y):
    x = y
    print("Matriks identitas dan ordonya "+str(x)+"x"+str(x))
    matriks = [[1 if j == i else 0 for j in range(y)]for i in range(x)]
    print(matriks)
```

IDLE Shell 3.10.3

File Edit Shell Debug Options Window Help

```
Python 3.10.3 (main, Mar 18 2022, 00:00:00) [GCC 11.2.1 20220127 (Red Ha
t 11.2.1-9)] on linux
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================= RESTART: /home/greenheaven/Documents/no2b.py =========
========
>>> buatIdentitas(5)
Matriks identitas dan ordonya 5x5
[[1, 0, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 0, 1, 0], [0,
0, 0, 0, 1]]
>>> buatIdentitas(1)
Matriks identitas dan ordonya 1x1
[[1]]
>>>
```

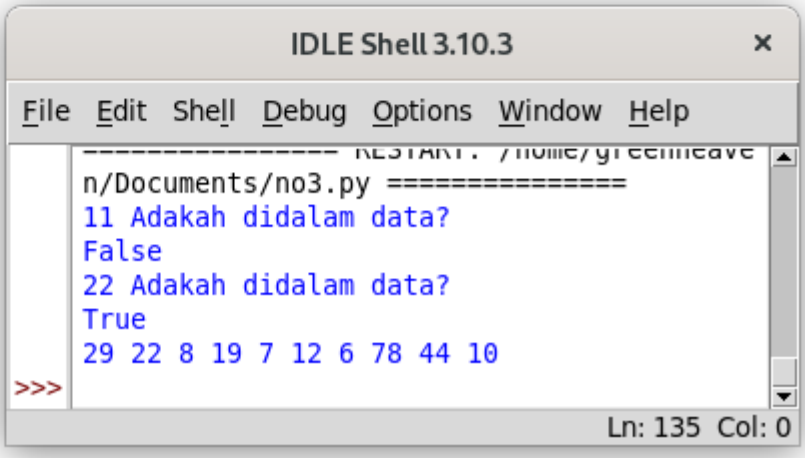
NO 3

```
class Base:
    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 = Base(new_data)
        new_node.next = self.head
        self.head = new_node
    def tambahAkhir(self, data):
        if (self.head == None):
            self.head = Base(data)
        else:
            current = self.head
            while (current.next != None):
                current = current.next
            current.next = Base(data)
        return self.head
    def tambah(self,data,pos):
        node = Base(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
        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:
```

```
    def hapus(self, posisi):
        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, "Adakah didalam data?")
                return True
            current = current.next
        print(x, "Adakah didalam 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(44)
a.tambahDepan(78)
a.tambahDepan(12)
a.tambahDepan(7)
a.tambahDepan(19)
a.tambahDepan(8)
a.tambahDepan(22)
a.tambahAkhir(10)
a.tambahDepan(29)
a.hapus(29)
a.tambah(6,6)
print(a.cari(11))
print(a.cari(22))
a.display()
```



NO 4

no4.py - /home/greenheaven/Documents/no4.py (3.10.3)

File Edit Format Run Options Window Help

```
class Base:
    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 awal bernilai", new_data)
        new_node = Base(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 akhir bernilai", new_data)
        new_node = Base(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 see(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
```

IDLE Shell 3.10.3

File Edit Shell Debug Options Window Help

```
>>>
========== RESTART: /home/greenheaven/Documents/no4.py ==
>>> A = DoublyLinkedList()
>>> A.awal(100)
Menambah awal bernilai 100
>>> A.awal(22)
Menambah awal bernilai 22
>>> A.akhir(21)
Menambah akhir bernilai 21
>>> A.akhir(87)
Menambah akhir bernilai 87
>>> A.see(A.head)

Dari Depan :
  22
  100
  21
  87

Dari Belakang :
  87
  21
  100
  22
>>>
```