

LAPORAN TUGAS

PRAKTIKUM ALGORITMA & STRUKTUR DATA

MODUL 3

Nama : Muhammad Ridwan NurFarizi
NIM : L200180020
Kelas : A

1. A.

The screenshot shows two windows side-by-side. The left window is the Python 3.7.6 Shell, displaying a command-line session where the user defines four variables (m1, m2, m3, m4) and checks their truthiness. The right window is a code editor for a file named T1A.py, containing Python code to validate matrix inputs.

```
Python 3.7.6 (tags/v3.7.6:43364a7ae0, Dec 19 2019, 00:42:30) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=====
RESTART: C:/Users/LABSI-10/Documents/Lat5.py =====
>>>
=====
RESTART: C:/Users/LABSI-10/Documents/T1A.py =====
m1 = True
m2 = True
m3 = False
m4 = False
>>>
=====
RESTART: C:/Users/LABSI-10/Documents/T1A.py =====
m1 = True
m2 = True
m3 = False
m4 = False
>>>
```

```
def cekMatrix(matrix) :
    panjang = len(matrix)
    hasil = True
    for x in matrix :
        lebar = len(x)
        if lebar != panjang:
            hasil = False
            break
        for i in x:
            if type(i) != int:
                hasil = False
                break
    return hasil

m1 = [[2,3],[4,5]]
m2 = [[10,20],[5,6]]
m3 = [[4,8,3],[2,"g",4],[3,6,9]]
m4 = [[6,2,7],[2,8]]
```

```
print("m1 =", cekMatrix(m1))
print("m2 =", cekMatrix(m2))
print("m3 =", cekMatrix(m3))
print("m4 =", cekMatrix(m4))
```

B.

The screenshot shows two windows side-by-side. The left window is titled "Python 3.7.6 Shell" and displays a Python session. The right window is titled "T1B.py - C:/Users/LABSI-10/Documents/T1B.py (3.7.6)" and shows the source code for a script named T1B.py. Both windows have status bars at the bottom indicating line and column numbers.

```
Python 3.7.6 (tags/v3.7.6:43364a7ae0, Dec 19 2019, 00:42:30) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ===== RESTART: C:/Users/LABSI-10/Documents/Lat5.py =====
>>> ===== RESTART: C:/Users/LABSI-10/Documents/T1A.py =====
m1 = True
m2 = True
m3 = False
m4 = False
>>> ===== RESTART: C:/Users/LABSI-10/Documents/T1B.py =====
m1 = True
m2 = True
m3 = False
m4 = False
>>> ===== RESTART: C:/Users/LABSI-10/Documents/T1B.py =====
Traceback (most recent call last):
  File "C:/Users/LABSI-10/Documents/T1B.py", line 34, in <module>
    print(Ukuran(m1))
  File "C:/Users/LABSI-10/Documents/T1B.py", line 29, in Ukuran
    return("Ukuran Matrix = "+ str(len(matrix))+ " x "+ str(len(matrix[0])))
TypeError: 'list' object is not callable
>>> ===== RESTART: C:/Users/LABSI-10/Documents/T1B.py =====
Ukuran Matrix = 2 x 2
Ukuran Matrix = 2 x 3
>>>
```

```
#1A
def cekMatrix(matrix) :
    # panjang = len(matrix)
    # hasil = True
    # for x in matrix :
    #     lebar = len(x)
    #     if lebar != panjang:
    #         hasil = False
    #         break
    #
    #     for i in x:
    #         if type(i) != int:
    #             hasil = False
    #             break
    #     return hasil

#m1 = [[2,3],[4,5]]
#m2 = [[10,20],[5,6]]
#m3 = [[4,8,3],[2,"8",4],[3,6,9]]
#m4 = [[6,2,7],[2,8]]

#print("m1 =", cekMatrix(m1))
#print("m2 =", cekMatrix(m2))
#print("m3 =", cekMatrix(m3))
#print("m4 =", cekMatrix(m4))

#1B
def Ukuran(matrix):
    return("Ukuran Matrix = "+ str(len(matrix))+ " x "+ str(len(matrix[0])))

m1 = [[2,3],[4,5]]
m2 = [[10,20,7],[5,6,3]]

print(Ukuran(m1))
print(Ukuran(m2))
```

C.

The screenshot shows a Windows desktop environment. In the foreground, there is a game window for "FINAL FANTASY XIV" with a character in a blue robe. Behind it, there is a code editor window titled "T1C.py" showing Python code. To the left, there is another code editor window titled "Python 3.8.2 Shell" showing a different Python session. The desktop background features a colorful, abstract pattern. The taskbar at the bottom shows various icons for applications like File Explorer, Task View, and system utilities.

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:ea7b3a059, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ===== RESTART: C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/T1C.py =====
nomer 1c
ukuran sama
[[1,2,3],[11,12,13]]
ukuran beda
>>>
```

```
T1C.py - C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/T1C.py
File Edit Format Run Options Window Help
k = [[4,7],[4,4]]
l = [[5,6],[5,5]]
m = [[3,4,5],[3,4,5]]
n = [[3,24],[32,5],[31,51]]
o = [[2,3,3],[17,2,22]]
p = [[5,9,20],[1,2,3],[3,4,5]]

def jumlah(n,m):
    x,y = 0,0
    for i in range(len(n)):
        for j in range(len(m)):
            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")
    print("nomer 1c")
jumlah(k,l)
jumlah(k,n)
```

D.

The image shows a Windows desktop environment. In the foreground, there are two open windows for Python 3.8.2 Shell. The left window displays a code snippet where a list 'm' is multiplied by a scalar '3'. The right window displays a more complex multiplication operation involving lists 'n' and 'm'. In the background, a Final Fantasy XIV character is visible in a game window, standing in a dark, atmospheric setting. The desktop background features a blue and purple abstract design. Various icons are visible on the taskbar and desktop, including Microsoft Edge, File Explorer, and several pinned application icons.

E.

A screenshot of a Windows desktop environment. In the center, there are two windows for the Python 3.8.2 Shell. The left window shows a command-line session where a user attempts to calculate the determinant of a matrix, but receives an error message: "tidak bisa dihitung determinannya, karena bukan matriks bujursangkar". The right window shows a code editor with a script named 'determinanHitung.py' containing a function 'determinanHitung' that calculates the determinant of a square matrix using cofactor expansion. Below the code editor is a game window for Final Fantasy X-2, showing a character in a dark, atmospheric setting.

```
Python 3.8.2 (tags/v3.8.2:cb79fbc, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

> RESTART: C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/_3TIE.py
3/TIE.py
nomer 1
13
-6
200
200
tidak bisa dihitung determinannya, karena bukan matriks bujursangkar
>>>
```

```
z = 0
for i in range(len(A)):
    if len(A[i]) == x:
        z+=1
if(z == 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 k in range(1, len(A)):
        As = []
        As.append(A[0])
        As.append(A[k])
        height = len(As)
        for f in range(1, height):
            As[f] = As[f][0:f] + As[f][f+1:]
        sign = (-1)**(f+k % 2)
        sub_det = determinanHitung(As)
        total += sign * A[0][f] * sub_det
    else:
        return "tidak bisa dihitung determinannya, karena bukan matriks bujur"
else:
    return "tidak bisa dihitung determinannya, karena bukan matriks bujursangkar"
return total
```

```
Q = [[3,1],[2,5]]
x = [[1,2,3,1],[3,3,1],[2,1,2]]
y = [[1,-2,0,0],[3,2,-3,1],[4,0,5,1],[2,3,-1,4]]
t = [[10,23,45,12,13],[1,2,3,4,5],[1,2,3,4,6],[1,2,3,4,8],[1,4,5,6,10]]

print("nomer 1")
print(determinanHitung(q))
print(determinanHitung(x))
print(determinanHitung(y))
print(determinanHitung(t))
print(determinanHitung(n))
print(determinanHitung(o))
```

```
Lm 12 Col 4
```

```
TIE.py - C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/TIE...
File Edit Format Run Options Window Help
def buatMol(n,x,m):
    if(m==1):
        mnn
        print("Membuat matriks 0 dengan ordo "+str(n)+"x"+str(m))
        print([[0 for x in range(m)] for y in range(n)])
    elif(m==2):
        buatMol(3)
        buatMol(5,3)
```

```
Lm 41 Col 12
```

2. A.

```
Python 3.8.2 (tags/v3.8.2:cb79fbc, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

> RESTART: C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/_3TIA.py
3/TIA.py
nomer 2
Hambuat matriks 0 dengan ordo 3x3
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
Hambuat matriks 0 dengan ordo 5x3
[[0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0]]
>>>
```

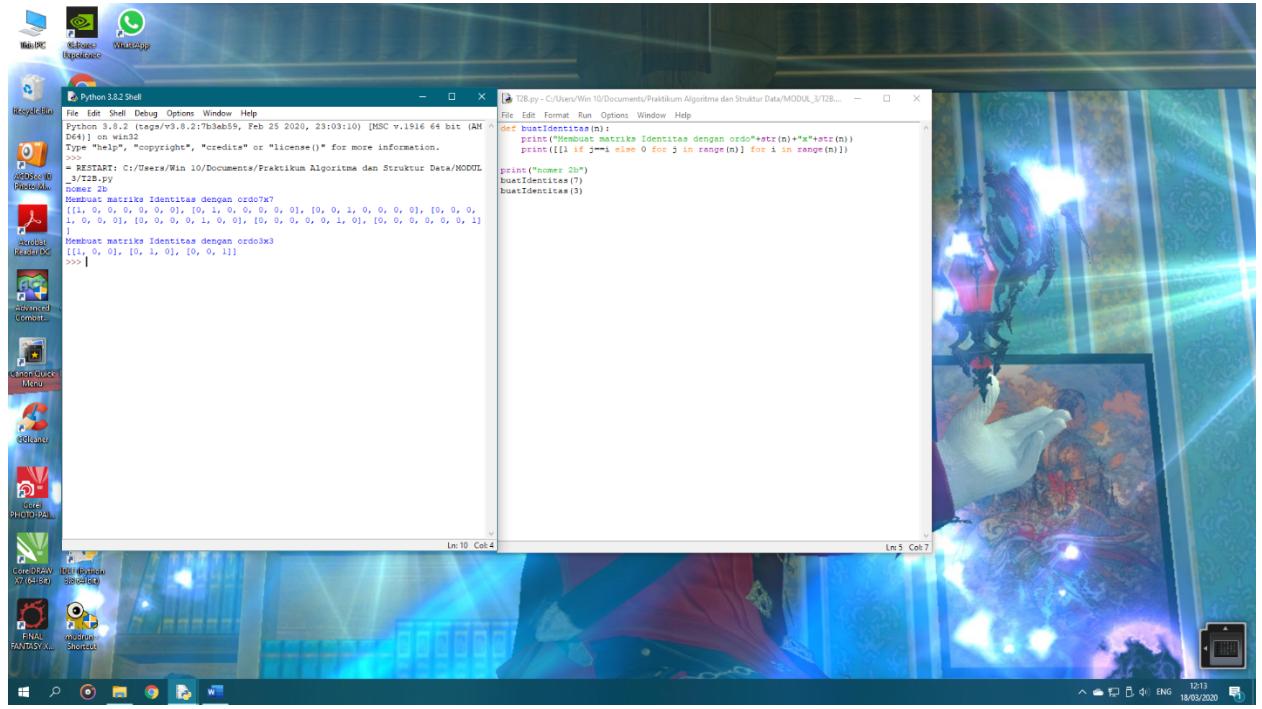
```
def buatMol(n,x,m):
    if(m==1):
        mnn
        print("Membuat matriks 0 dengan ordo "+str(n)+"x"+str(m))
        print([[0 for x in range(m)] for y in range(n)])
    elif(m==2):
        buatMol(3)
        buatMol(5,3)
```

```
Lm 10 Col 4
```

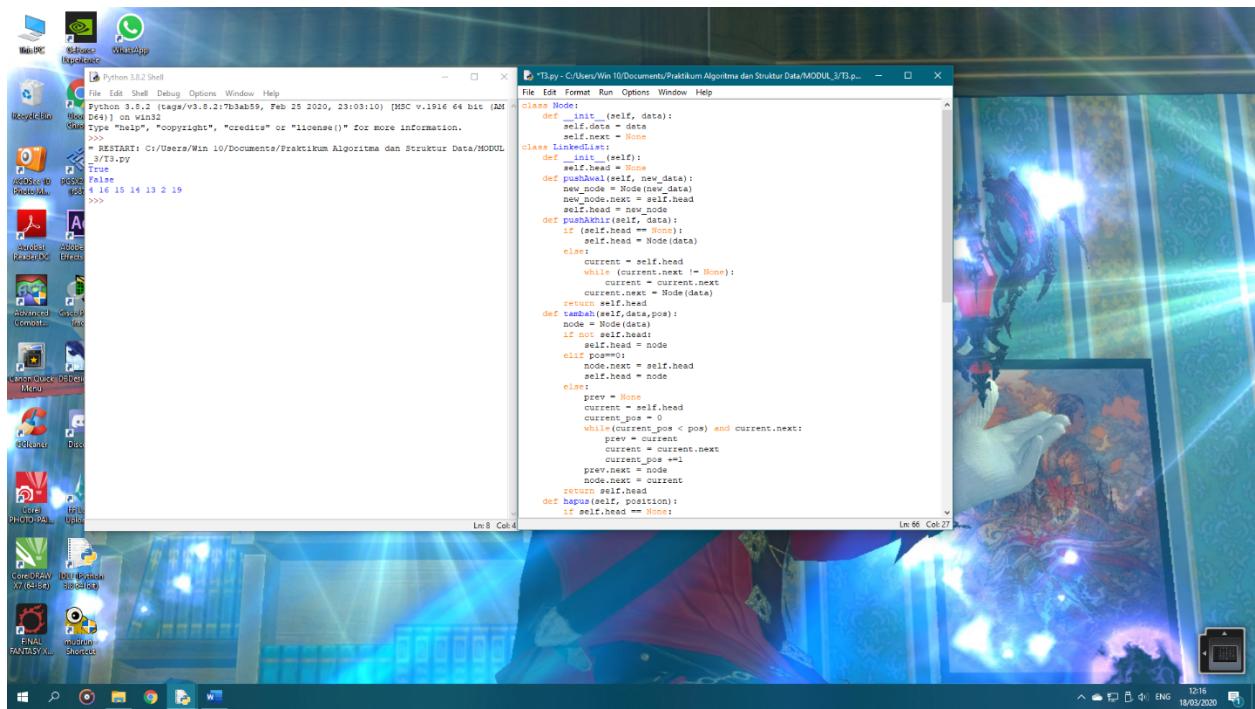
```
TIA.py - C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/TIA...
File Edit Format Run Options Window Help
def buatMol(n,x,m):
    if(m==1):
        mnn
        print("Membuat matriks 0 dengan ordo "+str(n)+"x"+str(m))
        print([[0 for x in range(m)] for y in range(n)])
    elif(m==2):
        buatMol(3)
        buatMol(5,3)
```

```
Lm 6 Col 7
```

B.



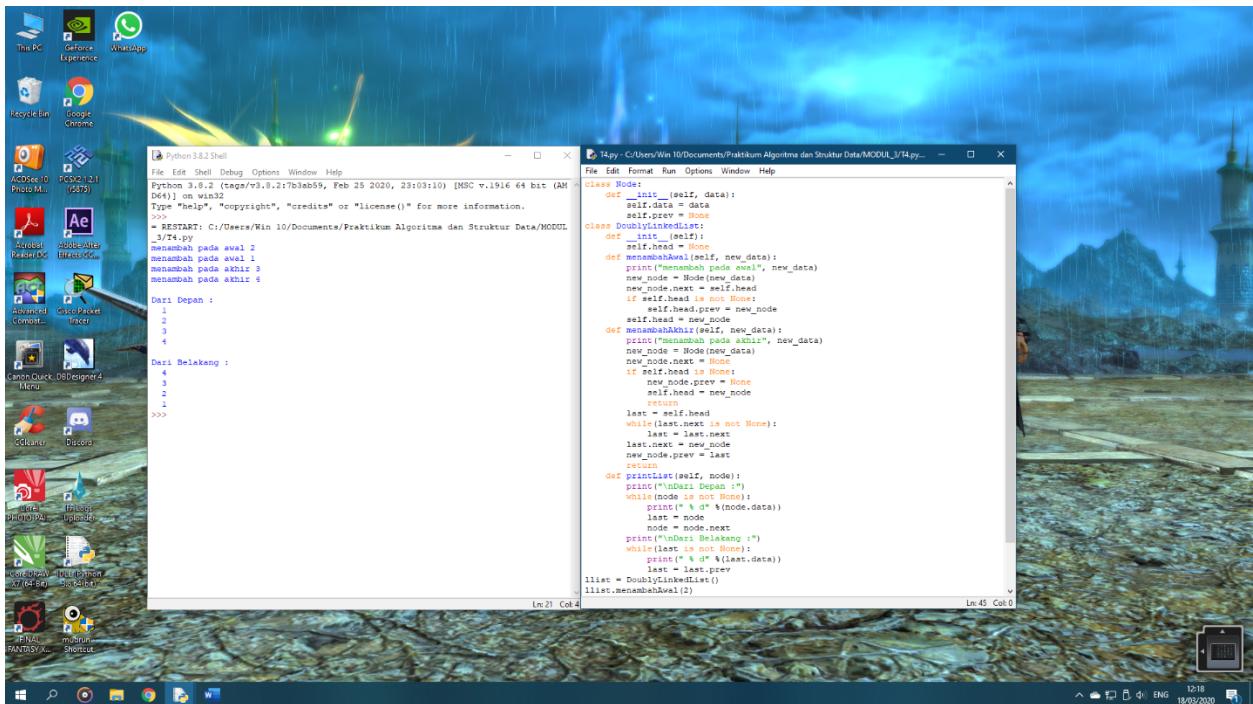
3.



```
File Edit Format Run Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 23:03:10) [MSC v.1914 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/T3.py
3/T3.PY
False
4 16 15 14 13 2 19
Lm: 8 Col: 4
```

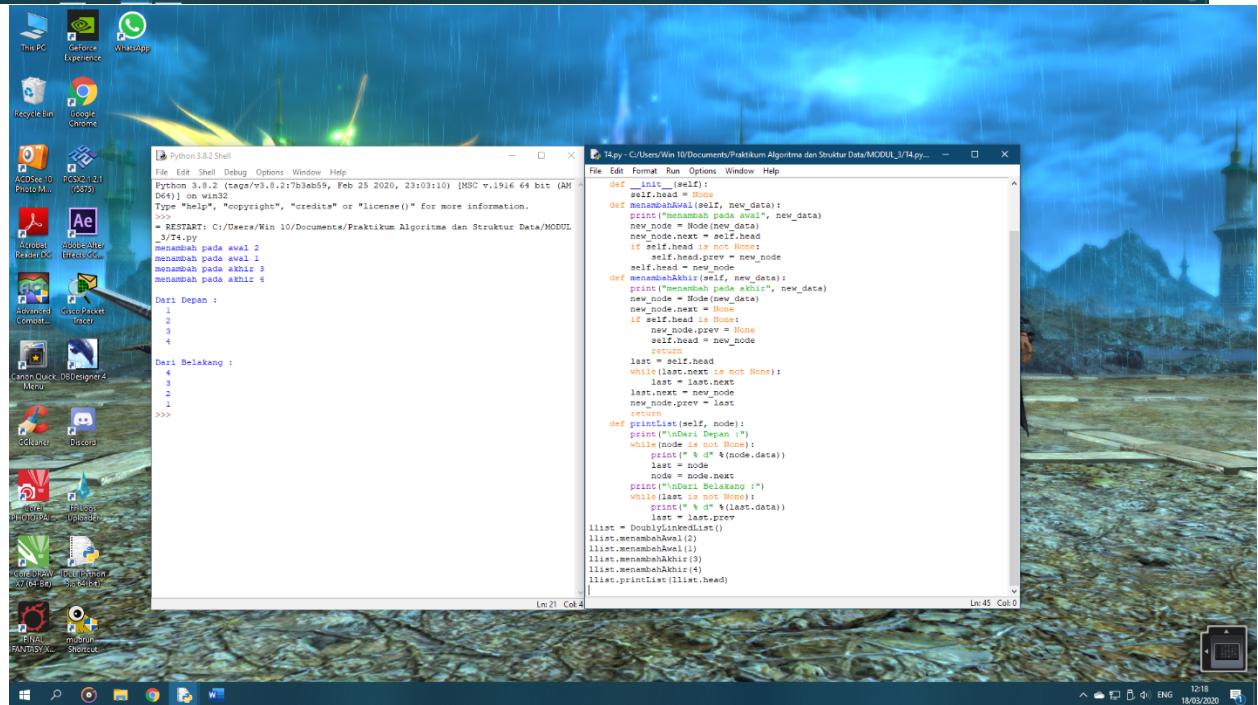
```
*T3.py - C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/T3.p...
File Edit Format Run Options Window Help
def hasap(self, position):
    if self.head == None:
        return False
    temp = self.head
    if position == 0:
        self.head = temp.next
        temp = None
        return True
    for i in range(position - 1):
        temp = temp.next
        if temp is None:
            break
    if temp is None:
        return False
    if temp.next is None:
        return False
    next = temp.next.next
    temp.next = next
    temp.next = 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.pushHval(13)
llist.pushHval(14)
llist.pushHval(15)
llist.pushHval(16)
llist.pushHval(17)
llist.pushHval(18)
llist.pushHval(19)
llist.tambah(2,5)
print(llist.cari(14))
print(llist.cari(17))
llist.display()
Lm: 8 Col: 27
```

4.



```
File Edit Shell Debug Options Window Help
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:cb73bb59, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
> RESTART: C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/74.py
menambah pada awal 2
menambah pada awal 1
menambah pada akhir 3
menambah pada akhir 8
Dari Depan :
2
3
4
5
6
7
8
Dari Belakang :
8
7
6
5
4
3
2
1
>>>
```

```
File Edit Format Run Options Window Help
4.py - C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/74.py...
File Edit Format Run Options Window Help
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None
class DoublyLinkedList:
    def __init__(self):
        self.head = None
    def menambahAwal(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 menambahAkhir(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):
        print("\nDari Depan :")
        while(self.head is not None):
            print("%d" % (self.head.data))
            self.head = self.head.next
        print("\nDari Belakang :")
        while(last is not None):
            print("%d" % (last.data))
            last = last.prev
    llist = DoublyLinkedList()
    llist.menambahAwal(2)
    llist.menambahAwal(1)
```



```
File Edit Shell Debug Options Window Help
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:cb73bb59, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
> RESTART: C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/74.py
menambah pada awal 2
menambah pada awal 1
menambah pada akhir 3
menambah pada akhir 4
Dari Depan :
2
3
4
5
6
7
8
Dari Belakang :
8
7
6
5
4
3
2
1
>>>
```

```
File Edit Format Run Options Window Help
4.py - C:/Users/Win 10/Documents/Praktikum Algoritma dan Struktur Data/MODUL_3/74.py...
File Edit Format Run Options Window Help
def __init__(self):
    self.head = None
def menambahAwal(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 menambahAkhir(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
    else:
        last = self.head
        while(last.next is not None):
            last = last.next
        last.next = new_node
        new_node.prev = last
    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):
        print("\nDari Depan :")
        while(self.head is not None):
            print("%d" % (self.head.data))
            self.head = self.head.next
        print("\nDari Belakang :")
        while(last is not None):
            print("%d" % (last.data))
            last = last.prev
    llist = DoublyLinkedList()
    llist.menambahAwal(2)
    llist.menambahAwal(1)
    llist.menambahAkhir(3)
    llist.menambahAkhir(4)
    llist.printlist(llist.head)
```