

Nama : Afiq Tri Nugraha

NIM : L200180080

Kelas : C

PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

MODUL 3

3.2 Array dan Array Dua Dimensi

- Latihan 3.1

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (tags/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.
>>> #Latihan 3.1
>>> A = [[2,3], [5,7]]
>>> A[0][1]
3
>>> A[1][1]
7
>>>
```

- Latihan 3.2

```
>>> #Latihan 3.2
>>> B = [[0 for j in range(3)] for i in range(3)]
>>> B
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
>>> #Membuat list kuadrat bilangan dari 0 sampai 6
>>> [x**2 for x in range(0,7)]
[0, 1, 4, 9, 16, 25, 36]
>>> #Membuat list yang berisi tuple pasangan bilangan dan kuadratnya, dari 0 sampai 6
>>> [(x,x**2) for x in range(7)]
[(0, 0), (1, 1), (2, 4), (3, 9), (4, 16), (5, 25), (6, 36)]
>>> #Membuat list kuadrat bilangan genap antara 0 sampai 15
>>> [x**2 for x in range(15) if x%2==0]
[0, 4, 16, 36, 64, 100, 144, 196]
>>> #Membuat list sepanjang 5 elemen yang berisi bilangan 3
>>> [3 for i in range(5)]
[3, 3, 3, 3, 3]
>>> #Membuat list sepanjang tiga elemen yang berisi list sepanjang 3 elemen angka 0
>>> [[0 for j in range(3)] for i in range(3)]
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
>>> #Membuat matriks identitas 3x3
>>> [[1 if j==i else 0 for j in range(3)] for i in range(3)]
[[1, 0, 0], [0, 1, 0], [0, 0, 1]]
>>> #Membuat matriks identitas 3x3
>>> [[1 if j==i else 0 for j in range(3)] for i in range(3)]
[[1, 0, 0], [0, 1, 0], [0, 0, 1]]
>>> #Membuat list yang berisi huruf vokal suatu string
>>> d = "Yogyakarta dan Surakarta"
>>> [x for x in d if x in "aeiouAEIOU"]
['o', 'a', 'a', 'a', 'a', 'u', 'a', 'a', 'a']
>>>
>>> #Membuat list bilangan prima dari 20 sampai 50
>>> [x for x in range(20,52) if apkarasPrima(x)]
[23, 29, 31, 37, 41, 43, 47]
>>>
```

3.3 Linked Structures

- Linked List

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

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

```

>>> a = Node(11)
>>> b = Node(12)
>>> c = Node(18)
>>> d = Node(30)
>>> e = Node(45)
>>> a.next = b
>>> b.next = c
>>> c.next = d
>>> d.next = e
>>> print(a.data)
11
>>> print(e.next.data)
52
>>> print(e.next.next.data)
18
>>> print(c.next.data)
30
>>> print(d.next.data)
45
>>> a.kunjungi(a)
11
12
18
30
45
>>> c.kunjungi(c)
18
30
45
>>> e.kunjungi(e)
45
>>> b.kunjungi(c)
18
30
45
>>> c.kunjungi(b)
12
18
30
45
>>>

```

- Advanced Linked List

```

#Advanced Linked List
class DNode(object):
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None

>>> a = DNode(11)
>>> b = DNode(12)
>>> c = DNode(18)
>>> d = DNode(30)
>>> e = DNode(45)
>>> print(d.data)
30
>>> a.next = b
>>> b.next = c
>>> print(a.next.data)
12
>>> e.prev = d
>>> d.prev = c
>>> c.prev = b
>>> b.prev = a
>>> print(d.prev.data)
18
>>> print(b.prev.data)
11
>>> print(e.prev.prev.data)
18
>>>

```

3.4 Soal-Soal Untuk Mahasiswa

1. Terkait array dua dimensi, kita akan membuat tipe data sebuah matrix yang berisi angka-angka. Untuk itu buatlah fungsi-fungsi:
 - Memastikan bahwa isi dan ukuran matrix-nya konsisten (karena tiap anggota dari list luar nya bisa saja mempunyai ukuran yang berbeda-beda daan bahkan bisa saja berbeda tipe)
 - Untuk mengambil ukuran matrix-nya
 - Untuk menjumlahkan dua matrix (pastikan ukurannya sesuai)
 - Untuk mengalikan dua matrix (pastikan ukurannya sesuai)
 - Untuk menghitung determinan sebuah matrix bujursangkar

Jawab:

```
L200180080_Algostruk_Modul 3.py - D:\AFIQ\Semester 4\Laporan praktikum\ASD\L200180080_Algostruk_Modul 3\L200180080_Algostruk_Modul 3.py (3.7.0)
File Edit Format Run Options Window Help

#NO 1

#Memastikan isi dan ukuran matrix
def cekMatrix(matrix):
    """memastikan type data Integer"""
    jum = len(matrix)
    hasil = ""
    for x in matrix:
        for i in x:
            assert isinstance(i, int), "Harus Integer"
    return True

#Mengambil ukuran matrix
def Ukuran(matrix):
    """Mengambil ukuran matriks"""
    return "Ukuran Matrix = "+str(len(matrix))+" x "+str(len(matrix[0]))

#Menjumlahkan dua matrix
def Jumlah(matrix1, matrix2):
    """Penjumlahan 2 Matrix"""
    if Ukuran(matrix1) == Ukuran(matrix2):
        for x in range(0, len(matrix1)):
            for y in range(0, len(matrix1[0])):
                print(matrix1[x][y] + matrix2[x][y], ' '),
            print()
    else:
        print("Matriks Tidak Sesuai")

#Mengalikan dua matrix
def Kali(matrix1, matrix2):
    """Perkalian 2 Matrix"""
    mat3 = []
    if Ukuran(matrix1) == Ukuran(matrix2):
        for x in range(0, len(matrix1)):
            row = []
            for y in range(0, len(matrix1[0])):
                total = 0
                for z in range(0, len(matrix1)):
                    total = total + (matrix1[x][z] * matrix2[z][y])
                row.append(total)
            mat3.append(row)

        for x in range(0, len(mat3)):
            for y in range(0, len(mat3[0])):

if Ukuran(matrix1) == Ukuran(matrix2):
    for x in range(0, len(matrix1)):
        for y in range(0, len(matrix1[0])):
            print(matrix1[x][y] + matrix2[x][y], ' '),
            print()
    else:
        print("Matriks Tidak Sesuai")

#Mengalikan dua matrix
def Kali(matrix1, matrix2):
    """Perkalian 2 Matrix"""
    mat3 = []
    if Ukuran(matrix1) == Ukuran(matrix2):
        for x in range(0, len(matrix1)):
            row = []
            for y in range(0, len(matrix1[0])):
                total = 0
                for z in range(0, len(matrix1)):
                    total = total + (matrix1[x][z] * matrix2[z][y])
                row.append(total)
            mat3.append(row)

        for x in range(0, len(mat3)):
            for y in range(0, len(mat3[0])):
                print(mat3[x][y], ' ')
            print()
    else:
        print("Matriks Tidak Sesuai")

#Menghitung determinan sebuah matrix
def determinan(matrix):
    """Menghitung Determinan Matrix"""
    if len(matrix) == len(matrix[0]):
        bil = [x for x in range(len(matrix))]
        jum = 0
        for i in range(len(matrix)):
            total = 1
            for x in range(len(matrix)):
                total *= matrix[x][bil[x]]
            bil += [bil.pop(0)]
            jum += total
        bil2 = [x for x in range(len(matrix))]
        bil.reverse()
        jum2 = 0

Ln: 128 Col: 35
```

```
L200180080_Algostruk_Modul 3.py - D:\AFIQ\Semester 4\Laporan praktikum\ASD\L200180080_Algostruk_Modul 3\L200180080_Algostruk_Modul 3.py (3.7.0)
File Edit Format Run Options Window Help

if Ukuran(matrix1) == Ukuran(matrix2):
    for x in range(0, len(matrix1)):
        for y in range(0, len(matrix1[0])):
            print(matrix1[x][y] + matrix2[x][y], ' '),
            print()
    else:
        print("Matriks Tidak Sesuai")

#Mengalikan dua matrix
def Kali(matrix1, matrix2):
    """Perkalian 2 Matrix"""
    mat3 = []
    if Ukuran(matrix1) == Ukuran(matrix2):
        for x in range(0, len(matrix1)):
            row = []
            for y in range(0, len(matrix1[0])):
                total = 0
                for z in range(0, len(matrix1)):
                    total = total + (matrix1[x][z] * matrix2[z][y])
                row.append(total)
            mat3.append(row)

        for x in range(0, len(mat3)):
            for y in range(0, len(mat3[0])):
                print(mat3[x][y], ' ')
            print()
    else:
        print("Matriks Tidak Sesuai")

#Menghitung determinan sebuah matrix
def determinan(matrix):
    """Menghitung Determinan Matrix"""
    if len(matrix) == len(matrix[0]):
        bil = [x for x in range(len(matrix))]
        jum = 0
        for i in range(len(matrix)):
            total = 1
            for x in range(len(matrix)):
                total *= matrix[x][bil[x]]
            bil += [bil.pop(0)]
            jum += total
        bil2 = [x for x in range(len(matrix))]
        bil.reverse()
        jum2 = 0

Ln: 128 Col: 35
```

```
L200180080_Algostruk_Modul 3.py - D:\AFIQ\Semester 4\Laporan praktikum\ASD\L200180080_Algostruk_Modul 3\L200180080_Algostruk_Modul 3.py (3.7.0)
File Edit Format Run Options Window Help

for x in range(0, len(mat3)):
    for y in range(0, len(mat3[0])):
        print(mat3[x][y], ' ')
    print()
else:
    print("Matriks Tidak Sesuai")

#Menghitung determinan sebuah matrix
def determinan(matrix):
    """Menghitung Determinan Matrix"""
    if len(matrix) == len(matrix[0]):
        bil = [x for x in range(len(matrix))]
        jum = 0
        for i in range(len(matrix)):
            total = 1
            for x in range(len(matrix)):
                total *= matrix[x][bil[x]]
            bil += [bil.pop(0)]
            jum += total
        bil2 = [x for x in range(len(matrix))]
        bil.reverse()
        jum2 = 0
        for i in range(len(matrix)):
            total2 = 1
            for x in range(len(matrix)):
                total2 *= matrix[x][bil2[x]]
            bil2 += [bil2.pop()]
            jum2 += total2
        print(total-total2)
        return ""
    else:
        print("Matriks Harus Bujursangkar")

#Mengecek program no 1
m1 = [[2,3],[4,5]]
m2 = [[10,20],[5,6]]

print("Nomer 1")
print(cekMatrix(m1))
print(Ukuran(m1))
Jumlah(m1,m2)
Kali(m1,m2)
print(determinan(m1))
```

#NO 1

#Memastikan isi dan ukuran matrix

def cekMatrix(matrix):

"""memastikan type data Integer"""

jum = len(matrix)

hasil = ""

for x in matrix:

for i in x:

assert isinstance(i, int), "Harus Integer"

return True

#Mengambil ukuran matrix

def Ukuran(matrix):

"""Mengambil ukuran matriks"""

return("Ukuran Matrix = "+str(len(matrix))+ " x "+str(len(matrix[0])))

#Menjumlahkan dua matrix

def Jumlah(matrix1,matrix2):

"""Penjumlahan 2 Matrix"""

```

if Ukuran(matrix1) == Ukuran(matrix2):
    for x in range(0, len(matrix1)):
        for y in range(0, len(matrix1[0])):
            print(matrix1[x][y] + matrix2[x][y], ' '),
        print()
else:
    print("Matriks Tidak Sesuai")

```

#Mengalikan dua matrix

```

def Kali(matrix1, matrix2):
    """Perkalian 2 Matrix"""
    mat3 = []
    if Ukuran(matrix1) == Ukuran(matrix2):
        for x in range(0, len(matrix1)):
            row = []
            for y in range(0, len(matrix1[0])):
                total = 0
                for z in range(0, len(matrix1)):
                    total = total + (matrix1[x][z] * matrix2[z][y])
                row.append(total)
            mat3.append(row)

        for x in range(0, len(mat3)):
            for y in range(0, len(mat3[0])):
                print(mat3[x][y], ' ')
            print()
    else:
        print("Matriks Tidak Sesuai")

```

#Menghitung determinan sebuah matrix

```

def determinan(matrix):
    """Menghitung Determinan Matrix"""
    if len(matrix) == len(matrix[0]):
        bil = [x for x in range(len(matrix))]

```

```

    jum = 0
    for i in range(len(matrix)):
        total = 1
        for x in range(len(matrix)):
            total *= matrix[x][bil[x]]
        bil += [bil.pop(0)]
        jum += total
    bil2 = [x for x in range(len(matrix))]
    bil.reverse()
    jum2 = 0
    for i in range(len(matrix)):
        total2 = 1
        for x in range(len(matrix)):
            total2 *= matrix[x][bil2[x]]
        bil2 += [bil2.pop()]
        jum2 += total2
    print(total-total2)
    return ""
else:
    print("Matriks Harus Bujursangkar")

```

#Mengecek program no 1

```
m1 = [[2,3],[4,5]]
```

```
m2 = [[10,20],[5,6]]
```

```
print("Nomer 1")
```

```
print(cekMatrix(m1))
```

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

```
Jumlah(m1,m2)
```

```
Kali(m1,m2)
```

```
print(determinan(m1))
```

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (vs3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: D:\AFIQ\Semester 4\Laporan praktikum\ASD\L200180080_Algostruk_Modul 3\L200180080_Algostruk_Modul 3.py
Nomer 1
True
Ukuran Matrix = 2 x 2
12
23
9
11
35
58
65
110
2
>>>
```

2. Terkait matrix dan list comprehension, buatlah (dengan memanfaatkan list comprehension) fungsi-fungsi:

- Untuk membangkitkan matrix berisi nol semua dengan diberikan ukurannya. Pemanggilan: `buatNol(m,n)` dan `buatNol(m)`. Pemanggilan dengan cara terakhir akan memberikan matrix bujursangkar ukuran $m \times n$
- Untuk membangkitkan matrix identitas dengan diberikan ukurannya. Pemanggilan: `buatIdentitas(m)`

Jawab:

```
##NO 2
#Membangkitkan matrix berisi nol semua
def buatNol(m, n):
    """Menggunakan dua input"""
    matrix = [[0 for x in range(m)] for i in range(n)]
    print(matrix)

def buatNol2(m):
    """Menggunakan satu input"""
    n = m
    matrix = [[0 for x in range(m)] for i in range(n)]
    print(matrix)

#Membangkitkan matrix identitas
def buatIdentitas(m):
    n = m
    matrix = [[1 if j == i else 0 for j in range(m)] for i in range(n)]
    print(matrix)

#Mengecek program no 2
print("Nomer 2")
buatNol(3,3)
buatNol2(3)
buatIdentitas(4)
print("\n")
```

###NO 2

#Membangkitkan matrix berisi nol semua

def buatNol(m, n):

"""Menggunakan dua input"""

matrix = [[0 for x in range(m)] for i in range(n)]

```
print(matrix)
```

```
def buatNol2(m):
```

```
    """Menggunakan satu input"""
```

```
    n = m
```

```
    matrix = [[0 for x in range(m)] for i in range(n)]
```

```
    print(matrix)
```

```
#Membangkitkan matrix identitas
```

```
def buatIdentitas(m):
```

```
    n = m
```

```
    matrix = [[1 if j == i else 0 for j in range(m)]for i in range(n)]
```

```
    print(matrix)
```

```
#Mengecek program no 2
```

```
print("Nomer 2")
```

```
buatNol(3,3)
```

```
buatNol2(3)
```

```
buatIdentitas(4)
```

```
print("\n")
```

```
Nomer 2
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
[[1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], [0, 0, 0, 1]]

>>> |
```

3. Terkait linked list, buatlah fungsi untuk:

- Mencari data yang isinya tertentu: cari(head,yang_dicari)
- Menambah suatu simpul di awal: tambahDepan(head)
- Menambah suatu simpul di akhir: tambahAkhir(head)
- Menyisipkan suatu simpul dimana saja: tambah(head,posisi)
- Menghapus suatu simpul di awal, di akhir, atau dimana saja: hapus(posisi)

Jawab:


```
#NO 3
print("Nomer 3")
class Node(object):
    def __init__(self, data, next=None):
        self.data = data
        self.next = next

def MakeNode(list):
    a = Node(list[0])
    if len(list) > 1:
        b = a
        for i in range(1, len(list)):
            b.next = Node(list[i])
            b = b.next
    return a

def kunjungi(head):
    curNode = head
    while curNode != None:
        print(curNode.data)
        curNode = curNode.next

def cari(head, yang_dicari):
    temp = head
    while temp != None:
        if temp.data == yang_dicari:
            return temp
        temp = temp.next
    return Node(None)

def tambahDepan(head):
    temp = Node("tambah depan", head)
    return temp

def tambahAkhir(head):
    temp = head
    while temp.next != None:
        temp = temp.next
    temp.next = Node("tambah akhir")
    return head

def tambah(head, posisi):
    """ Menambahkan simpul sebelum posisi """
    temp = head
```

Ln: 201 Col: 18

```
def tambah(head, posisi):
    """ Menambahkan simpul sebelum posisi """
    temp = head
    while temp != None:
        if temp.next.data == posisi:
            temp_belakang = temp.next
            temp.next = Node("tambah tengah", temp_belakang)
            return head
        temp = temp.next
    return None

def hapus(head, posisi):
    temp = head
    while temp != None:
        if temp.next.data == posisi:
            temp_belakang = temp.next.next
            temp.next = temp_belakang
            return head
        temp = temp.next
    return None

a = MakeNode(["Afik", "Tri", "Nugraha", "Annisa", "Nugraheni"])

print(a.data)
c = cari(a, "Afik")
print(c.next.data)

print()
kunjungi(a)

print()
a = tambahDepan(a)
kunjungi(a)

print()
a = tambahAkhir(a)
kunjungi(a)

print()
a = tambah(a, "Afik")
kunjungi(a)

print()
```

Ln: 201 Col: 18

```
"L200180080_Algostruk_Modul 3.py - D:\AFIQ\Semester 4\Laporan praktikum\ASD\L200180080_Algostruk_Modul 3\L200180080_Algostruk_Modul 3.py (3.7.0)"
File Edit Format Run Options Window Help

    if temp.next.data == posisi:
        temp.belakang = temp.next
        temp.next = Node("tambah tengah", temp.belakang)
        return head
    temp = temp.next
    return None

def hapus(head, posisi):
    temp = head
    while temp != None:
        if temp.next.data == posisi:
            temp.belakang = temp.next.next
            temp.next = temp.belakang
            return head
        temp = temp.next
    return None

a = MakeNode(["Afiq", "Tri", "Nugraha", "Annisa", "Nugraheni"])

print(a.data)
c = cari(a, "Afiq")
print(c.next.data)

print()
kunjungi(a)

print()
a = tambahDepan(a)
kunjungi(a)

print()
a = tambahAkhir(a)
kunjungi(a)

print()
a = tambah(a, "Afiq")
kunjungi(a)

print()
a = hapus(a, "Afiq")
kunjungi(a)
print("\n")
```

Berikut adalah program yang saya buat:

#NO 3

print("Nomer 3")

class Node(object):

 def __init__(self, data, next=None):

 self.data = data

 self.next = next

def MakeNode(list):

 a = Node(list[0])

 if len(list) > 1:

 b = a

 for i in range(1, len(list)):

 b.next = Node(list[i])

 b = b.next

 return a

def kunjungi(head):

 curNode = head

 while curNode != None:

 print(curNode.data)

```
curNode = curNode.next
```

```
def cari(head, yang_dicari):  
    temp = head  
    while temp != None :  
        if temp.data == yang_dicari:  
            return temp  
        temp = temp.next  
    return Node(None)
```

```
def tambahDepan(head):  
    temp = Node("tambah depan", head)  
    return temp
```

```
def tambahAkhir(head):  
    temp = head  
    while temp.next != None:  
        temp = temp.next  
    temp.next = Node("tambah akhir")  
    return head
```

```
def tambah(head, posisi):  
    """ Menambahkan simpul sebelum posisi """  
    temp = head  
    while temp != None:  
        if temp.next.data == posisi:  
            temp_belakang = temp.next  
            temp.next = Node("tambah tengah", temp_belakang)  
            return head  
        temp = temp.next  
    return None
```

```
def hapus(head, posisi):  
    temp = head
```

```
while temp != None:
    if temp.next.data == posisi:
        temp_belakang = temp.next.next
        temp.next = temp_belakang
        return head
    temp = temp.next
return None
```

```
a = MakeNode(["Afiq", "Tri", "Nugraha", "Annisa", "Nugraheni"])
```

```
print(a.data)
c = cari(a, "Afiq")
print(c.next.data)
```

```
print()
kunjungi(a)
```

```
print()
a = tambahDepan(a)
kunjungi(a)
```

```
print()
a = tambahAkhir(a)
kunjungi(a)
```

```
print()
a = tambah(a, " Afiq ")
kunjungi(a)
```

```
print()
a = hapus(a, " Afiq ")
kunjungi(a)
print("\n")
```

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help

Nomer 3
Afiq
Tri

Afiq
Tri
Nugraha
Annisa
Nugraheni

tambah depan
Afiq
Tri
Nugraha
Annisa
Nugraheni

tambah depan
Afiq
Tri
Nugraha
Annisa
Nugraheni
tambah akhir

tambah depan
tambah tengah
Afiq
Tri
Nugraha
Annisa
Nugraheni
tambah akhir

tambah depan
tambah tengah
Tri
Nugraha
Annisa
Nugraheni
tambah akhir

>>> |
```

4. Terkait doubly linked list, buatlah fungsi untuk:

- Mengunjungi dan mencetak data tiap simpul dari depan dan dari belakang
- Menambah suatu simpul di awal
- Menambah suatu simpul di akhir

Jawab:

```
"L200180080_Algostruk_Modul 3.py - D:\AFIQ\Semester 4\Laporan praktikum\ASD\L200180080_Algostruk_Modul 3\L200180080_Algostruk_Modul 3.py (3.7.0)"
File Edit Format Run Options Window Help

#NO 4
print("Nomor 4")
class DNode(object):
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None

def massDNodeCreator(list):
    a = DNode(list[0])
    p = a
    for i in list[1:]:
        p.next = DNode(i)
        p.next.prev = p
        p = p.next
    return a

def tambahSimpulAwal(head, data):
    data = DNode(data)
    data.next = head
    data.next.prev = data
    return data

def tambahSimpulAkhir(head, data):
    data = DNode(data)
    temp = head
    while temp.next != None:
        temp = temp.next
    temp.next = data
    return head

list = ["e", "f", "g", "h"]
a = massDNodeCreator(list)
print(a.next.next.next.prev.prev.data)

a = tambahSimpulAwal(a, "awal")
print(a.next.prev.data)

a = tambahSimpulAkhir(a, "akhir")
print(a.next.next.next.next.next.data)
```

Berikut adalah program yang saya buat:

```
#NO 4
```

```
print("Nomor 4")
```

```

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

def massDNodeCreator(list):
    a = DNode(list[0])
    p = a
    for i in list[1:]:
        p.next = DNode(i)
        p.next.prev = p
        p = p.next
    return a

def tambahSimpulAwal(head, data):
    data = DNode(data)
    data.next = head
    data.next.prev = data
    return data

def tambahSimpulAkhir(head, data):
    data = DNode(data)
    temp = head
    while temp.next != None:
        temp = temp.next
    temp.next = data
    return head

list = ["e", "f", "g", "h"]
a = massDNodeCreator(list)
print(a.next.next.next.prev.prev.data)

a = tambahSimpulAwal(a, "awal")

```

```
print(a.next.prev.data)
```

```
a = tambahSimpulAkhir(a, "akhir")
```

```
print(a.next.next.next.next.next.data)
```

```
Nomor 4  
f  
awal  
akhir  
>>>
```