

LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

MODUL 4



Nama : Daffa Putra Alwansyah
NIM : L200190031
Kelas : B

##Latihan

#=====

```
A=[10,51,2,18,4,31,13,5,23,64,29]
```

```
def cariLurus(wadah, target):
```

```
    n = len(wadah)
```

```
    for i in range(n):
```

```
        if wadah[i] == target:
```

```
            return True
```

```
    return False
```

```
>>> A=[10,51,2,18,4,31,13,5,23,64,29]
>>> cariLurus(A,10)
True
>>> cariLurus(A,69)
False
>>>
```

#=====

```
class MhsTIF(object):
```

```
    def __init__(self,nama,umur,tinggal,us):
```

```
        self.nama = nama
```

```
        self.umur = umur
```

```
        self.kotaTinggal = tinggal
```

```
        self.uangSaku = us
```

```
c0 = MhsTIF('Ika', 10, 'Sukoharjo', 240000)
```

```
c1 = MhsTIF('Budi', 51, 'Sragen', 230000)
```

```
c2 = MhsTIF('Ahmad', 2, 'Surakarta', 250000)
```

```
c3 = MhsTIF('Chandra', 18, 'Surakarta', 235000)
```

```
c4 = MhsTIF('Eka', 4, 'Boyolali', 240000)
```

```
c5 = MhsTIF('Fandi', 31, 'Salatiga', 250000)
```

```
c6 = MhsTIF('Deni', 13, 'Klaten', 245000)
```

```
c7 = MhsTIF('Galuh', 5, 'Wonogiri', 245000)
```

```
c8 = MhsTIF('Janto', 23, 'Klaten', 245000)
c9 = MhsTIF('Hasan', 64, 'Karanganyar', 270000)
c10 = MhsTIF('Khalid', 29, 'Purwodadi', 265000)
```

```
Daftar=[c0,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10]
target = 'Klaten'
for i in Daftar:
    if i.kotaTinggal == target:
        print(i.nama + 'tinggal di' + target)
```

```
= RESTART: D:\Kuliah\Semest
Denitinggal diKlaten
Jantotinggal diKlaten
>>> |
```

```
#=====
```

```
def usTerkecil():
    a = Daftar[0].uangSaku
    p = []
    for i in range(len(Daftar)):
        if a > Daftar[i].uangSaku:
            a = Daftar[i].uangSaku

    for i in range(len(Daftar)):
        if Daftar[i].uangSaku == a:
            p.append(Daftar[i].nama)

    return "Uang saku terkecil adalah " + str(p) + "dengan uang saku " + str(a)
```

```
>>> usTerkecil()
"Uang saku terkecil adalah ['Budi']dengan uang saku 230000"
>>> |
```

```
#=====
```

```
def usTerbesar():
    a = Daftar[0].uangSaku
    p = []
    for i in range(len(Daftar)):
        if a < Daftar[i].uangSaku:
            a = Daftar[i].uangSaku
    for i in range(len(Daftar)):
        if Daftar[i].uangSaku == a:
            p.append(Daftar[i].nama)

    return "uang saku terbesar adalah " + str(p) + "dengan uang saku " + str(a)
```

```
>>> usTerbesar()
"uang saku terbesar adalah ['Hasan']dengan uang saku 270000"
>>> |
```

```
#=====
```

```
def usKurang():
    p = []
```

```

for i in range (len(Daftar)):
    if Daftar[i].uangSaku < 250000:
        p.append(Daftar[i].nama)
return "Uang yang kurang dari 250k adalah: " + str(p)

```

```

>>> usKurang()
"Uang yang kurang dari 250k adalah: ['Ika', 'Budi', 'Chandra', 'Eka', 'Deni', 'Galuh', 'Janto']"
>>> |

```

```

#=====

```

```

def usLebih():
    p = []
    for i in range (len(Daftar)):
        if Daftar[i].uangSaku > 250000:
            p.append(Daftar[i].nama)
    return "Uang yang lebih dari 250k adalah: " + str(p)

```

```

>>> usLebih()
"Uang yang lebih dari 250k adalah: ['Hasan', 'Khalid']"
>>> |

```

```

#=====

```

```

p = [2,4,5,10,13,18,23,29,31,51,64]

```

```

def binSe(kumpulan,target):
    low = 0
    high = len(kumpulan)-1

    x=[]
    while low <=high:
        mid =(high + low)//2
        if kumpulan[mid]==target:
            return True
        elif target < kumpulan[mid]:
            high = mid -1
        else:
            low = mid+1
    return False

```

```

>>> binSe(p,2)
True
>>> binSe(p,3333)
False
>>> |

```

```

#=====

```

```

p = [2,4,5,10,13,18,23,29,31,51,64]

```

```

def binSeDex(kumpulan,target):
    low = 0
    high = len(kumpulan)-1
    a = []
    while low <= high:
        if kumpulan [low] == target:
            a.append(low)
            low += 1

```

```

        return a
    else:
        low += 1
    return False

```

```

>>> binSeDex(p, 2)
[0]
>>> binSeDex(p, 5)
[2]
>>> binSeDex(p, 55)
False
>>> |

```

```
#=====
```

=====**TUGAS MAHASISWA**=====

```

class MhsTIF(object):
    def __init__(self,nama,umur,tinggal,us):
        self.nama = nama
        self.umur = umur
        self.kotaTinggal = tinggal
        self.uangSaku = us

```

```

c0 = MhsTIF('Ika', 10, 'Sukoharjo', 240000)
c1 = MhsTIF('Budi', 51, 'Sragen', 230000)
c2 = MhsTIF('Ahmad', 2, 'Surakarta', 250000)
c3 = MhsTIF('Chandra', 18, 'Surakarta', 235000)
c4 = MhsTIF('Eka', 4, 'Boyolali', 240000)
c5 = MhsTIF('Fandi', 31, 'Salatiga', 250000)
c6 = MhsTIF('Deni', 13, 'Klaten', 245000)
c7 = MhsTIF('Galuh', 5, 'Wonogiri', 245000)
c8 = MhsTIF('Janto', 23, 'Klaten', 245000)
c9 = MhsTIF('Hasan', 64, 'Karanganyar', 270000)
c10 = MhsTIF('Khalid', 29, 'Purwodadi', 265000)

```

```
Daftar=[c0,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10]
```

Nomor 1

```

def cari(caritinggal):
    x=[]
    for i in range(len(Daftar)):
        if caritinggal == Daftar[i].kotaTinggal:
            x.append(i)
    print(x)
    if len (x) > 0:
        print(True)
    else:
        print(False)

```

```

>>> cari("Klaten")
[6, 8]
True

```

```
#=====
```

Nomor 2 dan 3

```
def usTerkecil():
    a = Daftar[0].uangSaku
    p = []
    for i in range(len(Daftar)):
        if a > Daftar[i].uangSaku:
            a = Daftar[i].uangSaku

    for i in range(len(Daftar)):
        if Daftar[i].uangSaku == a:
            p.append(Daftar[i].nama)

    return "Uang saku terkecil adalah " + str(p) + " dengan uang saku " + str(a)
```

```
>>> usTerkecil()
"Uang saku terkecil adalah ['Budi'] dengan uang saku 230000"
```

```
#=====
```

Nomor 4

```
def usKurang():
    p = []
    for i in range(len(Daftar)):
        if Daftar[i].uangSaku < 250000:
            p.append(Daftar[i].nama)
    return "Uang yang kurang dari 250k adalah: " + str(p)
```

```
>>> usKurang()
"Uang yang kurang dari 250k adalah: ['Ika', 'Budi', 'Chandra', 'Eka', 'Deni', 'Galuh', 'Janto']"
```

```
def usLebih():
    p = []
    for i in range(len(Daftar)):
        if Daftar[i].uangSaku > 250000:
            p.append(Daftar[i].nama)
    return "Uang yang lebih dari 250k adalah: " + str(p)
```

```
>>> usLebih()
"Uang yang lebih dari 250k adalah: ['Hasan', 'Khalid']"
```

```
#=====
```

Nomor 5

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

    def cari(self, dicari):
```

```

x = self
while x is not None:
    if x.next != None:
        if x.data != dicari:
            x = x.next
        else:
            print ("Data", dicari, "ada dalam Linked List")
            break
    elif x.next == None:
        print ("Data", dicari, "tidak ada dalam Linked List")
        break
a = LinkedList(1)
menu = a
a.next = LinkedList(2)
a = a.next
a.next = LinkedList(3)
a = a.next
a.next = LinkedList(4)
a = a.next
menu.cari(1)
menu.cari(5)

```

```

= RESTART: D:\Kuliah\Semester 4\Prakt
y
Data 1 ada dalam Linked List
Data 5 tidak ada dalam Linked List
>>> |

```

#=====

Nomor 6

```

p = [2,4,5,10,13,18,23,29,31,51,64]
def binSeDex(kumpulan,target):
    low = 0
    high = len(kumpulan)-1
    a = []
    while low <= high:
        if kumpulan [low] == target:
            a.append(low)
            low += 1
            return a
        else:
            low += 1
    return False

```

```

>>> binSeDex (p, 64)
[10]
>>> binSeDex (p, 2)
[0]
>>> binSeDex (p, 33)
False
>>> |

```

#=====

Nomor 7

p = [2, 3, 5, 6, 6, 6, 8, 9, 9, 10, 11, 12, 13, 13, 14]

```
def binSe(kumpulan,target):
    low = 0
    high = len(kumpulan)-1
    a = []
    while low <= high:
        if kumpulan [low] == target:
            a.append(low)
            low += 1
        else:
            low += 1
    return a
```

```
>>> binSe(p,6)
[3, 4, 5]
>>> binSe(p,13)
[12, 13]
>>> |
```

#=====

Nomor 8

```
print(
    ""Terdapat 2 pola.
```

Pertama menggunakan konsep Big-O. Dimana yang dipakai adalah rumus $O(\log n)$ dengan rincian $1 = 1$, $2 = 2$, $4 = 3$, $10 = 4$, $100 = 7$, $1000 = 10$. Di mana log berasal dari pangkat log berbasis 2.

Untuk pola pertama:

```
apabila ingin menebak angka 70
a = nilai tebakan pertama // 2
tebakan selanjutnya = nilai tebakan "lebih dari" + a
*jika hasil tebakan selanjutnya "kurang dari", maka nilai yang dipakai
tetap nilai lebih dari sebelumnya*
a = a // 2
```

tebakan ke 1: 50 (mengambil nilai tengah) jawaban= "lebih dari itu"
tebakan ke 2: 75 (dari 50 + 25) jawaban = "kurang dari itu,ulangi"
tebakan ke 3: 62 (dari 50 + 12) jawaban = "lebih dari itu,ulangi"
tebakan ke 4: 68 (dari 62 + 6) jawaban = "lebih dari itu,ulangi"
tebakan ke 5: 71 (dari 68 + 3) jawaban = "kurang dari itu,ulangi"
tebakan ke 6: 69 (dari 68 + 1) jawaban = "lebih dari itu,ulangi"
tebakan ke 7: antara 71 dan 69 hanya ada 1 angka = 70!!!

Pola kedua menggunakan barisan geometri $S_n = 2^n$

barisan yang terjadi adalah : 2, 4, 8, 16, 32, 64

Misal angka yang akan diebak adalah 68

Tebakan ke-1 : 64 dijawab lebih dari itu

Tebakan ke-2 : 96(dari 64 + 32) dijawab "Kurang dari itu,ulangi"

Tebakan ke-3 : 80(dari 64 + 16) dijawab "Kurang dari itu,ulangi"

Tebakan ke-4 : 72(dari $64 + 8$) dijawab "Kurang dari itu,ulangi"
Tebakan ke-5 : 68(dari $64 + 4$) dijawab "Lebih dari itu,ulangi"
Tebakan ke-6 : 70(dari $68 + 2$) dijawab "TEPAT"
""")