

Nama : Fatwa Dwi Hidayat

Nim : L200180176

Kelas : G

LAPORAN PRAKTIKUM MODUL 4

ALGORITMA DAN STRUKTUR DATA

1. Membuat suatu fungsi pencarian yang, alih alih mengembalikan True/False

```
1.py - C:\Users\asus\OneDrive\Desktop\1.py (3.8.2)
File Edit Format Run Options Window Help

class Mhs(object):
    def __init__(self, nama, nim, kota, us):
        self.nama = nama
        self.nim = nim
        self.kotaTinggal = kota
        self.uangSaku = us

m0 = Mhs("Aldy", 175, "jakarta", 240000)
m1 = Mhs("Fatwa", 111, "bandung", 230000)
m2 = Mhs("Fakhar", 123, "Surakarta", 250000)
m3 = Mhs("Erdi", 122, "Surakarta", 235000)
m4 = Mhs("Hanan", 133, "papua", 240000)
m5 = Mhs("Rizki", 999, "kendari", 250000)
m6 = Mhs("iqbal", 908, "Riau", 245000)
m7 = Mhs("ijul", 678, "padang", 245000)
m8 = Mhs("fikri", 456, "Sorong", 245000)
m9 = Mhs("wafiq", 000, "sumba", 270000)
m10 = Mhs("kevin", 127, "wonogiri", 265000)

Daftar = [m0, m1, m2, m3, m4, m5, m6, m7, m8, m9, m10]

def cariKotaTinggal(list, target):
    a = []
    for i in list:
        if i.kotaTinggal == target:
            a.append(list.index(i))
    return a

a = cariKotaTinggal(Daftar, "Surakarta")
b = cariKotaTinggal(Daftar, "Magelang")
print(a)
print(b)

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) on win32
Type "help", "copyright", "credits" or "license()" for more
>>>
===== RESTART: C:\Users\asus\OneDrive\Desktop\1.py (3.8.2)
>>> [2, 3]
>>> []
>>> |
```

2. Membuat fungsi untuk menemukan uang saku yang terkecil

```
2.py - C:\Users\asus\OneDrive\Desktop\2.py (3.8.2)
File Edit Format Run Options Window Help

class Mhs(object):
    def __init__(self, nama, nim, kota, us):
        self.nama = nama
        self.nim = nim
        self.kotaTinggal = kota
        self.uangSaku = us

m0 = Mhs("Aldy", 175, "jakarta", 240000)
m1 = Mhs("Fatwa", 111, "bandung", 230000)
m2 = Mhs("Fakhar", 123, "Surakarta", 250000)
m3 = Mhs("Erdi", 122, "Surakarta", 235000)
m4 = Mhs("Hanan", 133, "papua", 240000)
m5 = Mhs("Rizki", 999, "kendari", 250000)
m6 = Mhs("iqbal", 908, "Riau", 245000)
m7 = Mhs("ijul", 678, "padang", 245000)
m8 = Mhs("fikri", 456, "Sorong", 245000)
m9 = Mhs("wafiq", 000, "sumba", 270000)
m10 = Mhs("kevin", 127, "wonogiri", 265000)

Daftar = [m0, m1, m2, m3, m4, m5, m6, m7, m8, m9, m10]

def cariUangSakuTerkecil(list):
    temp = list[0].uangSaku
    for i in list[1:]:
        if i.uangSaku < temp:
            temp = i.uangSaku
            nama = i.nama
    return nama, temp

a = cariUangSakuTerkecil(Daftar)
print(a)
```

```
Python 3.8.2 Shell
File Edit Shell Debug Optic
Python 3.8.2 (tags/v3.8.2:
tel)] on win32
Type "help", "copyright
>>>
===== REST
('Fatwa', 230000)
>>> |
```

3. Mengubah program nomor 2, jika ada lebih dari satu mahasiswa yang uang sakunya terkecil, semua objek mahasiswa itu dikembalikan

```
3.py - C:\Users\asus\OneDrive\Desktop\3.py (3.8.2)
File Edit Format Run Options Window Help

class Mhs(object):
    def __init__(self, nama, nim, kota, us):
        self.nama = nama
        self.nim = nim
        self.kotaTinggal = kota
        self.uangSaku = us

m0 = Mhs("Aldy", 175, "jakarta", 240000)
m1 = Mhs("Fatwa", 111, "bandung", 230000)
m2 = Mhs("Fakhar", 123, "Surakarta", 250000)
m3 = Mhs("Erdi", 122, "Surakarta", 235000)
m4 = Mhs("Hanan", 133, "papua", 240000)
m5 = Mhs("Rizki", 999, "kendari", 250000)
m6 = Mhs("iqbal", 908, "Riau", 245000)
m7 = Mhs("ijul", 678, "padang", 245000)
m8 = Mhs("fikri", 456, "Sorong", 245000)
m9 = Mhs("wafiq", 000, "sumba", 270000)
m10 = Mhs("kevin", 127, "wonogiri", 265000)

Daftar = [m0, m1, m2, m3, m4, m5, m6, m7, m8, m9, m10]

def cariUangSakuTerkecilObject(n):
    baru = n[0].uangSaku
    list = []
    for i in range(len(n)):
        if (n[i].uangSaku==baru):
            list.append(n[i].nama)
        elif (n[i].uangSaku<baru):
            baru = n[i].uangSaku
            list = []
            list.append(n[i].nama)
    return list

a = cariUangSakuTerkecilObject(Daftar)
print(a)
```

```
Python 3.8.2 Shell
File Edit Shell Debug Optic
Python 3.8.2 (tags/v3.8.2:
tel)] on win32
Type "help", "copyright
>>>
=====
['Fatwa']
>>> |
```

4. Membuat fungsi untuk mengembalikan semua objek mahasiswa yang uang sakunya kurang dari 250000

```
4.py - C:\Users\asus\OneDrive\Desktop\4.py (3.8.2)
File Edit Format Run Options Window Help

class Mhs(object):
    def __init__(self, nama, nim, kota, us):
        self.nama = nama
        self.nim = nim
        self.kotaTinggal = kota
        self.uangSaku = us

m0 = Mhs("Aldy", 175, "jakarta", 240000)
m1 = Mhs("Fatwa", 111, "bandung", 230000)
m2 = Mhs("Fakhar", 123, "Surakarta", 250000)
m3 = Mhs("Erdi", 122, "Surakarta", 235000)
m4 = Mhs("Hanan", 133, "papua", 240000)
m5 = Mhs("Rizki", 999, "kendari", 250000)
m6 = Mhs("iqbal", 908, "Riau", 245000)
m7 = Mhs("ijul", 678, "padang", 245000)
m8 = Mhs("fikri", 456, "Sorong", 245000)
m9 = Mhs("wafiq", 000, "sumba", 270000)
m10 = Mhs("kevin", 127, "wonogiri", 265000)

Daftar = [m0, m1, m2, m3, m4, m5, m6, m7, m8, m9, m10]

def cariUangSakuKurang250k(list):
    temp = []
    for i in list:
        if i.uangSaku < 250000:
            temp.append(i)
    return temp

a = cariUangSakuKurang250k(Daftar)
for i in a:
    print(i.nama)
```

```
Python 3.8
File Edit She
Python 3.8
tel)] on w
Type "help
>>>
=====
Aldy
Fatwa
Erdi
Hanan
iqbal
ijul
fikri
>>> |
```

5. Membuat suatu program untuk mencari suatu item di sebuah linked list

```
5.py - C:\Users\asus\OneDrive\Desktop\5.py (3.8.2)
File Edit Format Run Options Window Help

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

    def cariLinkedList(self, dicari):
        curNode = self
        while curNode is not None:
            if curNode.next != None:
                if curNode.data != dicari:
                    curNode = curNode.next
            else:
                print ("Data ", dicari, "ada dalam linked list")
                break
        elif curNode.next == None:
            print ("Data ", dicari, "tidak ada dalam linked list")
            break

a = node(12)
menu = a
a.next = node(34)
a = a.next
a.next = node(10)
a = a.next
a.next = node(45)

menu.cariLinkedList(10)
menu.cariLinkedList(110)
```

```
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 3
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more informatio
>>>
===== RESTART: C:\Users\asus\OneDrive\Desktop\5.py =====
Data 10 ada dalam linked list
Data 110 tidak ada dalam linked list
>>> |
```

6. Binary search, mengubah fungsi binSe agar mengembalikan index lokasi elemen yang ditemukan, kalau tidak ketemu kembali False

```
6.py - C:\Users\asus\OneDrive\Desktop\6.py (3.8.2)
File Edit Format Run Options Window Help
def binary (kumpulan,target):
    low = 0
    high = len(kumpulan) -1

    while low <= high:
        mid = (high + low) // 2
        if kumpulan [mid] == target:
            return "target berada di index " + str(mid)
            break

        elif target < kumpulan [mid]:
            high = mid - 1

        else:
            low = mid + 1

    return False

listnya = [23, 51, 78, 10, 62, 145, 410]
target1 = 56
target2 = 62

print("\nlistnya adalah ",listnya)
print("nilai target adalah ", target1)
print(binary(listnya, target1))

print("\nlistnya adalah ",listnya)
print("nilai target adalah ", target2)
print(binary(listnya, target2))

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
tel)] on win32
Type "help", "copyright", "credits" or "license()" for m
>>>
===== RESTART: C:\Users\asus\OneDrive\Desкто
listnya adalah [23, 51, 78, 10, 62, 145, 410]
nilai target adalah 56
False

listnya adalah [23, 51, 78, 10, 62, 145, 410]
nilai target adalah 62
target berada di index 4
>>> |
```

7. Seperti no 6 tapi, mencari angka 6 pada list [2,3,5,6,6,6,8,9,9,10,11,12,13,13,14] akan mengembalikan [3,4,5]. Karena sudah urut, "tinggal melihat kiri dan kanannya"

```
7.py - C:\Users\asus\OneDrive\Desktop\7.py (3.8.2)
File Edit Format Run Options Window Help
def binary (kumpulan,target):
    low = 0
    high = len(kumpulan) -1
    listku = []

    while low <= high:
        if kumpulan [low] == target:
            listku.append(low)
            low += 1

        else:
            low += 1

    return listku

s = [2,3,5,6,6,6,8,9,9,10,11,12,13,14]
dicari = 6
print("Posisi data ", dicari, " pada list ", s, "adalah ",binary(s, dicari))

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 win3
2
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\asus\OneDrive\Desktop\7.py =====
Posisi data 6 pada list [2, 3, 5, 6, 6, 6, 8, 9, 9, 10, 11, 12, 13, 14] adalah [3, 4, 5]
>>> |
```


8. Tebak angka modul 1, menjelaskan pola nya

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: C:\Users\asus\OneDrive\Desktop\8.py =====

Soal :

Pada permainan tebak angka yang sudah kamu buat di Modul 1 (soal nomer 12, halaman 16) kalau angka yang harus ditebak berada diantara 1-100, seharusnya maksimal jumlah tebakan adalah 7. Kalau antara 1-1000, maksimal jumlah tebakan adalah 10. Mengapa seperti itu? Bagaimana polanya

Jawab :

Ada dua kemungkinan pola yang bisa digunakan.

Misalkan, angka yang akan ditebak adalah 70.

-POLA PERTAMA-

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

SIMULASI

tebakan ke-1 : 50 (mengambil nilai tengah) Jawaban = "Lebih dari Itu"

tebakan ke-2 : 75 (dari 50 + 25) Jawaban = "Kurang dari Itu"

tebakan ke-3 : 62 (dari 50 + 12) Jawaban = "Lebih dari Itu"

tebakan ke-4 : 68 (dari 62 + 6) Jawaban = "Lebih dari Itu"

tebakan ke-5 : 71 (dari 68 + 3) Jawaban = "Kurang dari Itu"

tebakan ke-6 : 69 (dari 68 + 1) Jawaban = "Lebih dari Itu"

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 ditebak adalah 68

Tebakan ke-1 : 64 dijawab lebih dari itu

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

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

Tebakan ke-4 : 72(dari 64 + 8) dijawab "Kurang dari itu"

Tebakan ke-5 : 68(dari 64 + 4) dijawab "Lebih dari itu"

Tebakan ke-6 : 70(dari 68 + 2) dijawab "TEPAT"

>>> |