

Nama : Galih Prayoga

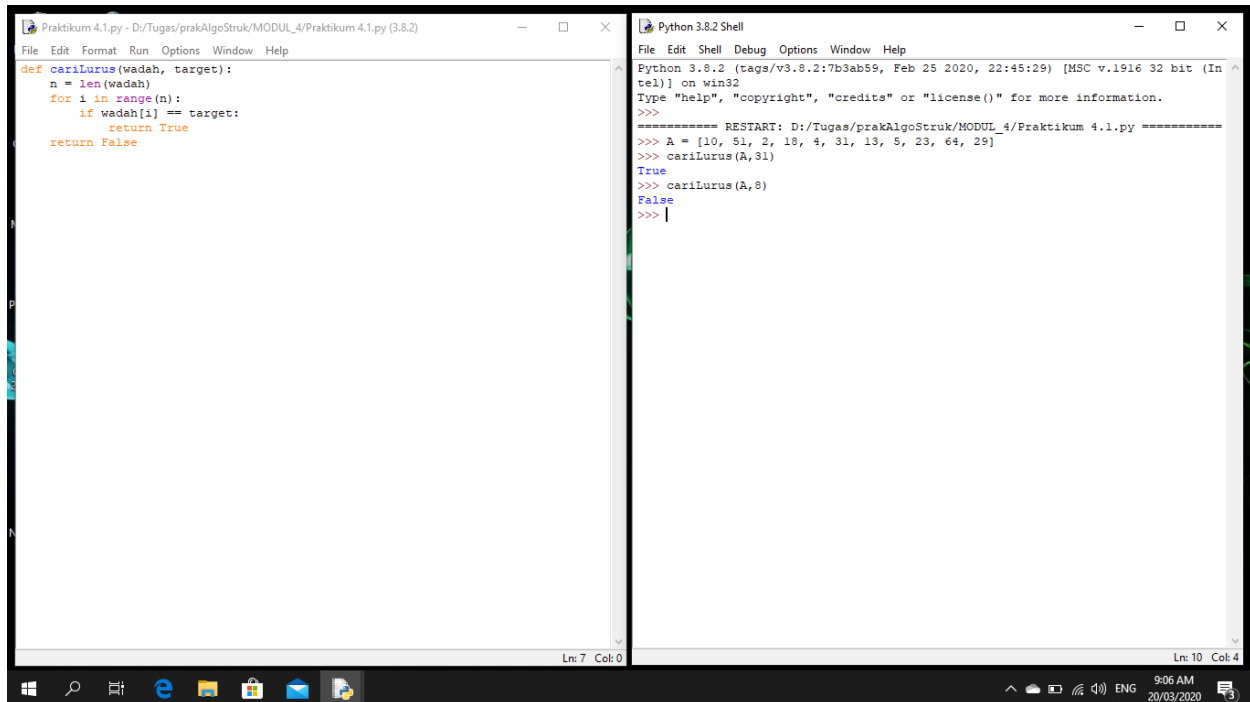
NIM : L200180006

Kelas : A

MODUL 4(PRAKTIKUM DAN TUGAS)

- Praktikum) :

4.1 Linear search

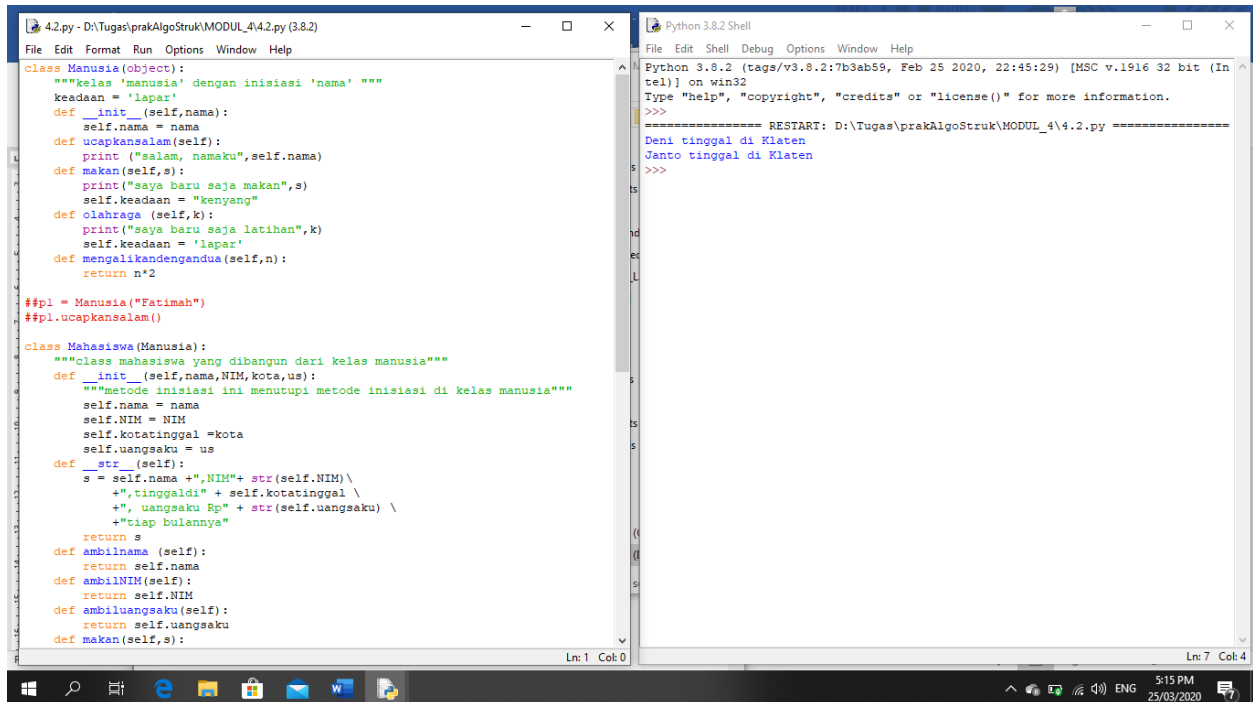


The screenshot displays a Python IDE with two windows. The left window, titled 'Praktikum 4.1.py', contains a function definition for a linear search algorithm. The right window, titled 'Python 3.8.2 Shell', shows the execution of the function with a list of numbers and a target value.

```
def carilurus(wadah, target):  
    n = len(wadah)  
    for i in range(n):  
        if wadah[i] == target:  
            return True  
    return False
```

```
Python 3.8.2 Shell  
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: D:/Tugas/prakAlgoStruk/MODUL_4/Praktikum 4.1.py =====  
>>> A = [10, 51, 2, 18, 4, 31, 13, 5, 23, 64, 29]  
>>> carilurus(A,31)  
True  
>>> carilurus(A,8)  
False  
>>> |
```

4.2 Pencarian lurus untuk objek membuat sendiri



The screenshot shows a Python IDE with two windows. The left window, titled '4.2.py - D:\Tugas\prakAlgoStruk\MODUL_4\4.2.py (3.8.2)', contains the following code:

```
class Manusia(object):
    """Kelas 'manusia' dengan inisiasi 'nama' """
    keadaan = 'lapar'
    def __init__(self, nama):
        self.nama = nama
    def ucapkansalam(self):
        print("salam, namaku", self.nama)
    def makan(self, s):
        print("saya baru saja makan", s)
        self.keadaan = "kenyang"
    def olahraga(self, k):
        print("saya baru saja latihan", k)
        self.keadaan = 'lapar'
    def mengalikandengandua(self, n):
        return n*2

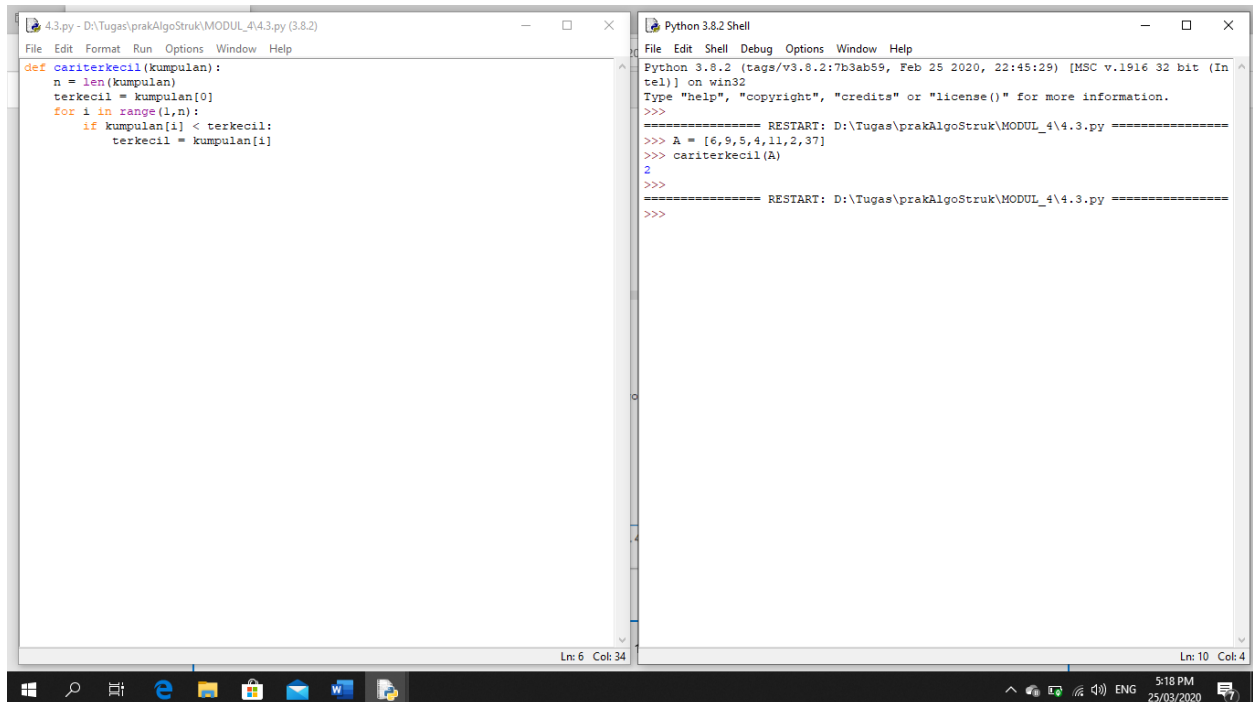
#pi = Manusia("Fatimah")
#pi.ucapkanksalam()

class Mahasiswa(Manusia):
    """Class mahasiswa yang dibangun dari kelas manusia"""
    def __init__(self, nama, NIM, kota, us):
        """metode inisiasi ini menutupi metode inisiasi di kelas manusia"""
        self.nama = nama
        self.NIM = NIM
        self.kotatinggal = kota
        self.uangaku = us
    def __str__(self):
        s = self.nama + ", NIM" + str(self.NIM) \
            + ", tinggal di" + self.kotatinggal \
            + ", uangaku Rp" + str(self.uangaku) \
            + "tiap bulannya"
        return s
    def ambilnama(self):
        return self.nama
    def ambilNIM(self):
        return self.NIM
    def ambiluangaku(self):
        return self.uangaku
    def makan(self, s):
```

The right window, titled 'Python 3.8.2 Shell', shows the execution output:

```
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: D:\Tugas\prakAlgoStruk\MODUL_4\4.2.py =====
>>>
Deni tinggal di Klaten
Janto tinggal di Klaten
>>>
```

4.3 Mencari nilai yang terkecil pada array yang tidak urut



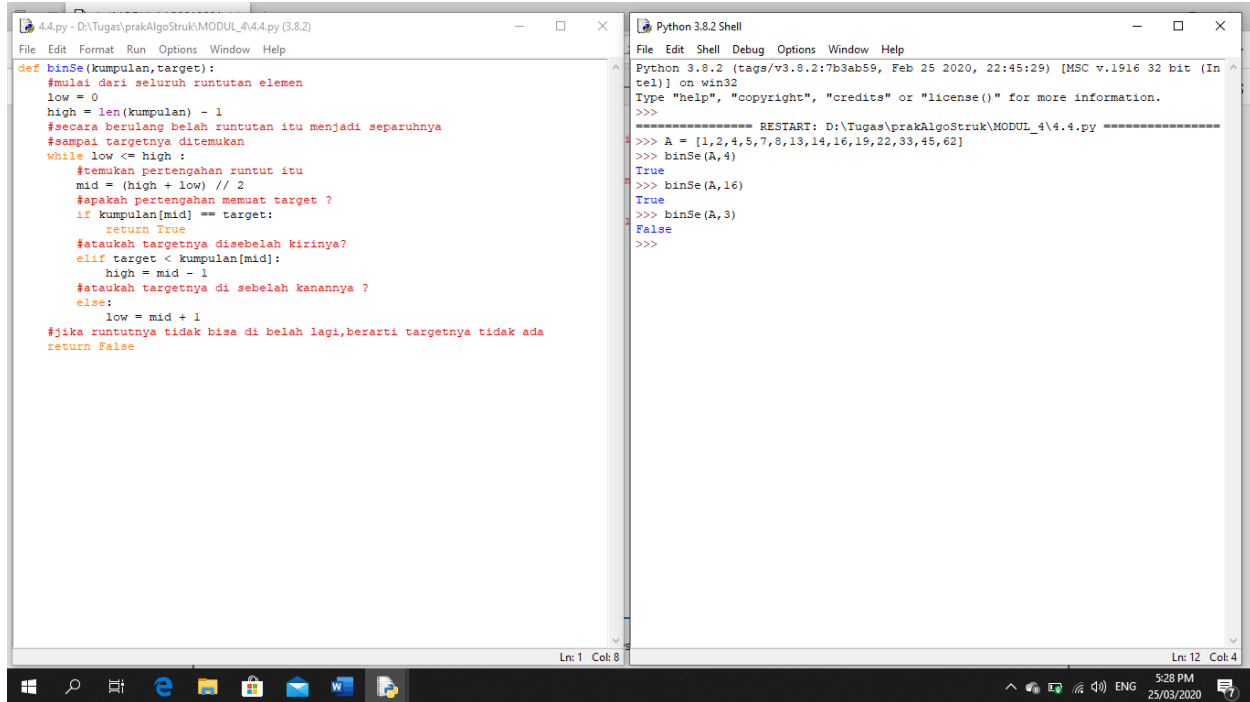
The screenshot shows a Python IDE with two windows. The left window, titled '4.3.py - D:\Tugas\prakAlgoStruk\MODUL_4\4.3.py (3.8.2)', contains the following code:

```
def cariterkecil(kumpulan):
    n = len(kumpulan)
    terkecil = kumpulan[0]
    for i in range(1, n):
        if kumpulan[i] < terkecil:
            terkecil = kumpulan[i]
    return terkecil
```

The right window, titled 'Python 3.8.2 Shell', shows the execution output:

```
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: D:\Tugas\prakAlgoStruk\MODUL_4\4.3.py =====
>>> A = [6,9,5,4,11,2,37]
>>> cariterkecil(A)
2
>>>
===== RESTART: D:\Tugas\prakAlgoStruk\MODUL_4\4.3.py =====
>>>
```

4.4 Binary Search



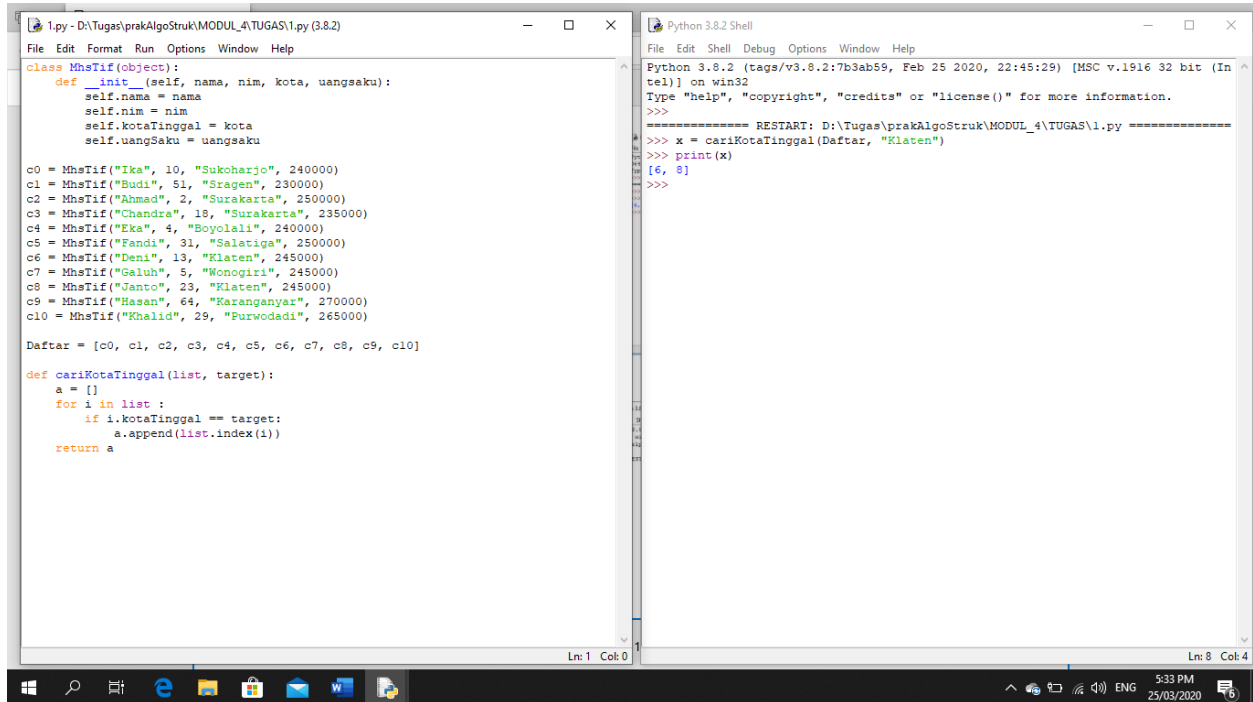
The image shows a screenshot of a Python IDE with two windows. The left window, titled '4.4.py - D:\Tugas\prakAlgoStruk\MODUL_4\4.4.py (3.8.2)', contains a Python function `binSe(kumpulan, target):` that implements a binary search algorithm. The function uses a `while` loop to repeatedly divide the search range in half until the target is found or the range is empty. The right window, titled 'Python 3.8.2 Shell', shows the execution of the function with a list `A = [1, 2, 4, 5, 7, 8, 13, 14, 16, 19, 22, 33, 45, 62]`. The function is called with `binSe(A, 4)` and `binSe(A, 16)`, both returning `True`, and with `binSe(A, 3)`, returning `False`.

```
def binSe(kumpulan, target):
    #mulai dari seluruh runtutan elemen
    low = 0
    high = len(kumpulan) - 1
    #secara berulang belah runtutan itu menjadi separuhnya
    #sampai targetnya ditemukan
    while low <= high :
        #temukan pertengahan runtut itu
        mid = (high + low) // 2
        #apakah pertengahan memuat target ?
        if kumpulan[mid] == target:
            return True
        #ataukah targetnya disebelah kirinya?
        elif target < kumpulan[mid]:
            high = mid - 1
        #ataukah targetnya di sebelah kanannya ?
        else:
            low = mid + 1
    #jika runtutnya tidak bisa di belah lagi, berarti targetnya tidak ada
    return False

>>>
===== RESTART: D:\Tugas\prakAlgoStruk\MODUL_4\4.4.py =====
>>> A = [1, 2, 4, 5, 7, 8, 13, 14, 16, 19, 22, 33, 45, 62]
>>> binSe(A, 4)
True
>>> binSe(A, 16)
True
>>> binSe(A, 3)
False
>>>
```

• Tugas :

1. Soal nomor 1



```
1.py - D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS1.1.py (3.8.2)
File Edit Format Run Options Window Help
class MhsTif(object):
    def __init__(self, nama, nim, kota, uangsku):
        self.nama = nama
        self.nim = nim
        self.kotaTinggal = kota
        self.uangSaku = uangsku

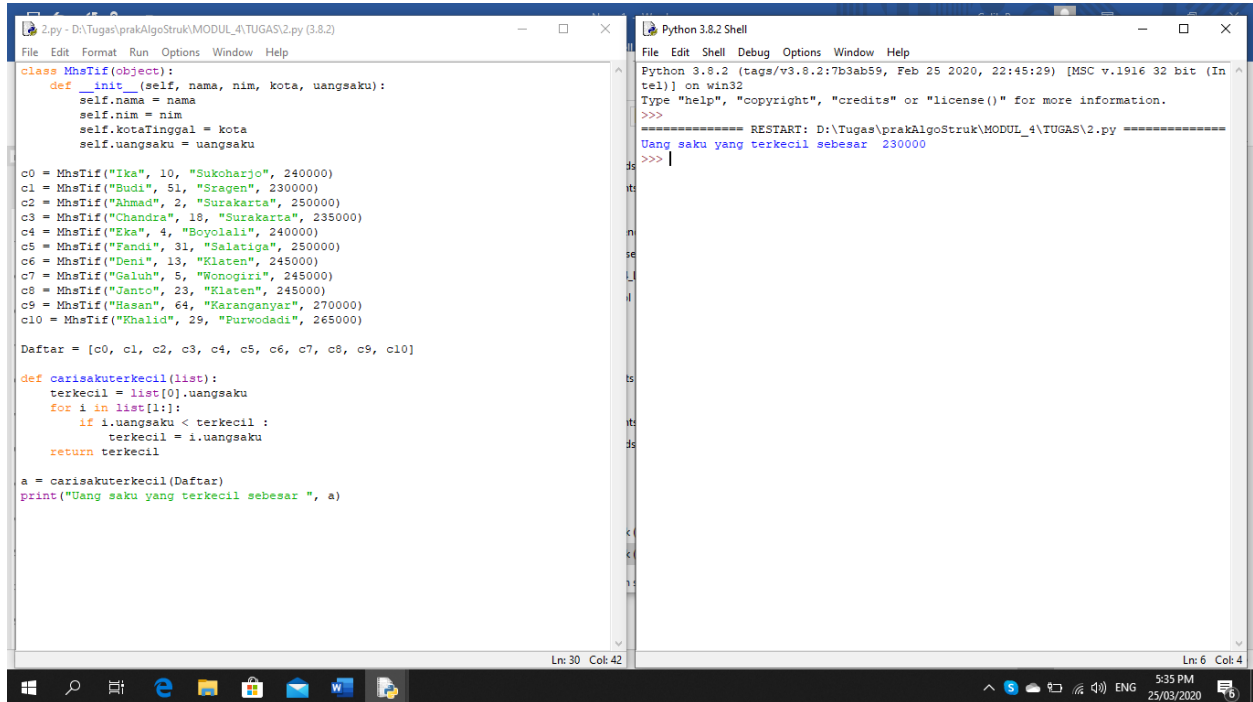
c0 = MhsTif("Ika", 10, "Sukoharjo", 240000)
c1 = MhsTif("Budi", 51, "Sragen", 230000)
c2 = MhsTif("Ahmad", 2, "Surakarta", 250000)
c3 = MhsTif("Chandra", 18, "Surakarta", 235000)
c4 = MhsTif("Eka", 4, "Boyolali", 240000)
c5 = MhsTif("Fandi", 31, "Salatiga", 250000)
c6 = MhsTif("Deni", 13, "Klaten", 245000)
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]

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

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: D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS1.1.py =====
>>> x = cariKotaTinggal(Daftar, "Klaten")
>>> print(x)
[6, 8]
>>>
```

2. Soal nomor 2



```
2.py - D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS2.py (3.8.2)
File Edit Format Run Options Window Help
class MhsTif(object):
    def __init__(self, nama, nim, kota, uangsku):
        self.nama = nama
        self.nim = nim
        self.kotaTinggal = kota
        self.uangsku = uangsku

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)

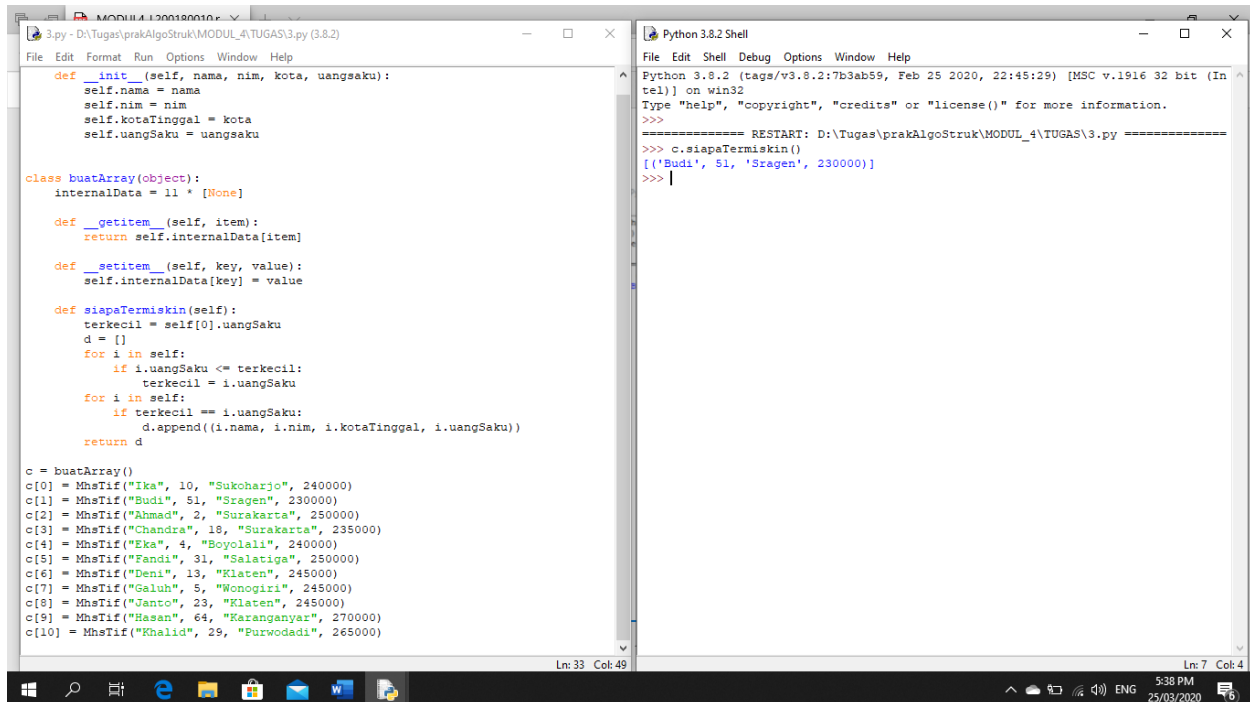
Daftar = [c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10]

def carisakuterkecil(list):
    terkecil = list[0].uangsku
    for i in list[1:]:
        if i.uangsku < terkecil:
            terkecil = i.uangsku
    return terkecil

a = carisakuterkecil(Daftar)
print("Uang saku yang terkecil sebesar ", a)

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: D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS2.py =====
>>>
Uang saku yang terkecil sebesar 230000
>>>
```

3. Soal nomor 3



```
3.py - D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS3.py (3.8.2)
File Edit Format Run Options Window Help

def __init__(self, nama, nim, kota, uangsaku):
    self.nama = nama
    self.nim = nim
    self.kotaTinggal = kota
    self.uangSaku = uangsaku

class bustArray(object):
    internalData = {}

    def __getitem__(self, item):
        return self.internalData[item]

    def __setitem__(self, key, value):
        self.internalData[key] = value

    def siapaTermiskin(self):
        terkecil = self[0].uangSaku
        d = []
        for i in self:
            if i.uangSaku <= terkecil:
                terkecil = i.uangSaku
        for i in self:
            if terkecil == i.uangSaku:
                d.append((i.nama, i.nim, i.kotaTinggal, i.uangSaku))
        return d

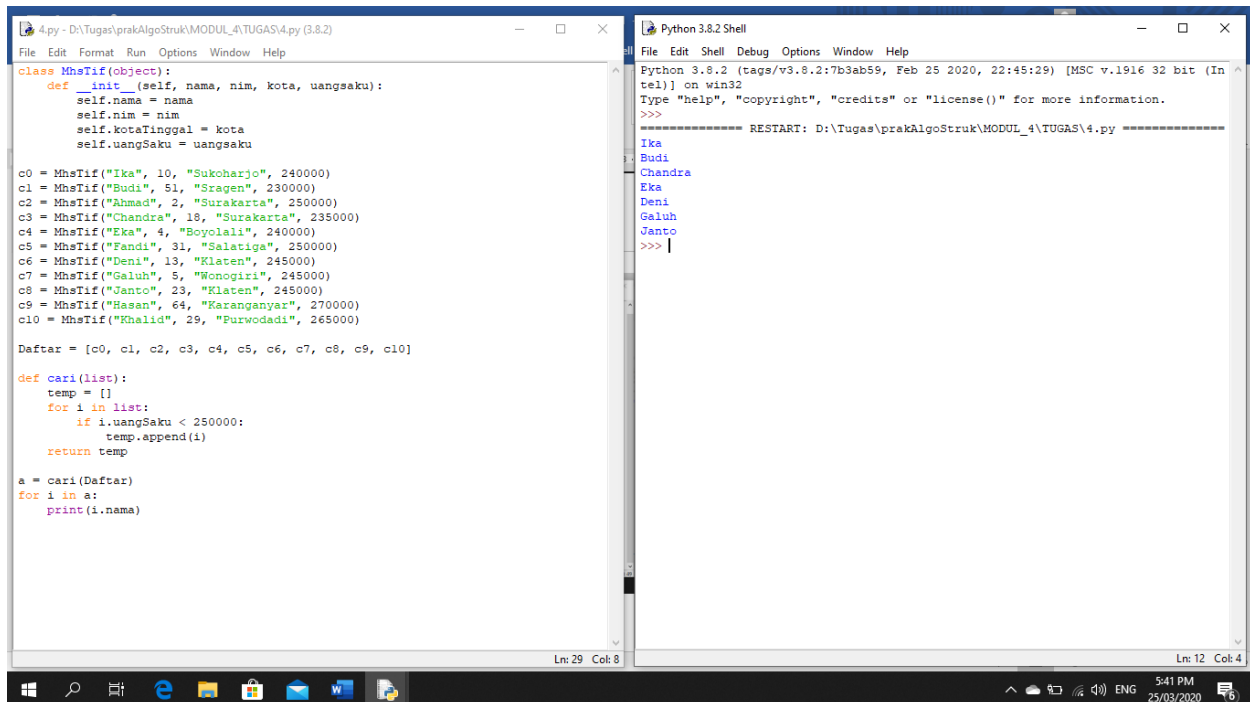
c = bustArray()
c[0] = MhsTif("Ika", 10, "Sukoharjo", 240000)
c[1] = MhsTif("Budi", 51, "Sragen", 230000)
c[2] = MhsTif("Ahmad", 2, "Surakarta", 250000)
c[3] = MhsTif("Chandra", 18, "Surakarta", 235000)
c[4] = MhsTif("Eka", 4, "Boyolali", 240000)
c[5] = MhsTif("Fandi", 31, "Salatiga", 250000)
c[6] = MhsTif("Deni", 13, "Klaten", 245000)
c[7] = MhsTif("Galuh", 5, "Wonogiri", 245000)
c[8] = MhsTif("Janto", 23, "Klaten", 245000)
c[9] = MhsTif("Hasan", 64, "Karanganyar", 270000)
c[10] = MhsTif("Khalid", 29, "Purwodadi", 265000)

Ln: 33 Col: 49

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: D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS3.py =====
>>> c.siapaTermiskin()
[('Budi', 51, 'Sragen', 230000)]
>>>
```

4. Soal nomor 4



```
4.py - D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS4.py (3.8.2)
File Edit Format Run Options Window Help

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

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]

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

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

Ln: 29 Col: 8

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: D:\Tugas\prakAlgoStruk\MODUL_4\TUGAS4.py =====
>>>
Ika
Budi
Chandra
Eka
Deni
Galuh
Janto
>>>
```

5. Soal nomor 5

The screenshot shows a Python IDE with two windows. The left window, titled '5.py', contains the following code:

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

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

The right window, titled 'Python 3.8.2 Shell', shows the execution of the code:

```
>>> a = node(12)
>>> menu = a
>>> a.next = node(11)
>>> a = a.next
>>> a.next = node(26)
>>> a = a.next
>>> a.next = node(43)
>>> menu.cari(12)
Data 12 ada dalam linked list
>>> menu.cari(44)
Data 44 tidak ada dalam linked list
>>>
```

The taskbar at the bottom shows the Windows Start button, search icon, and several application icons. The system tray on the right indicates the time is 5:47 PM on 25/03/2020.

6. Soal nomor 6

The screenshot shows a Python IDE with two windows. The left window, titled '6.py', contains the following code:

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

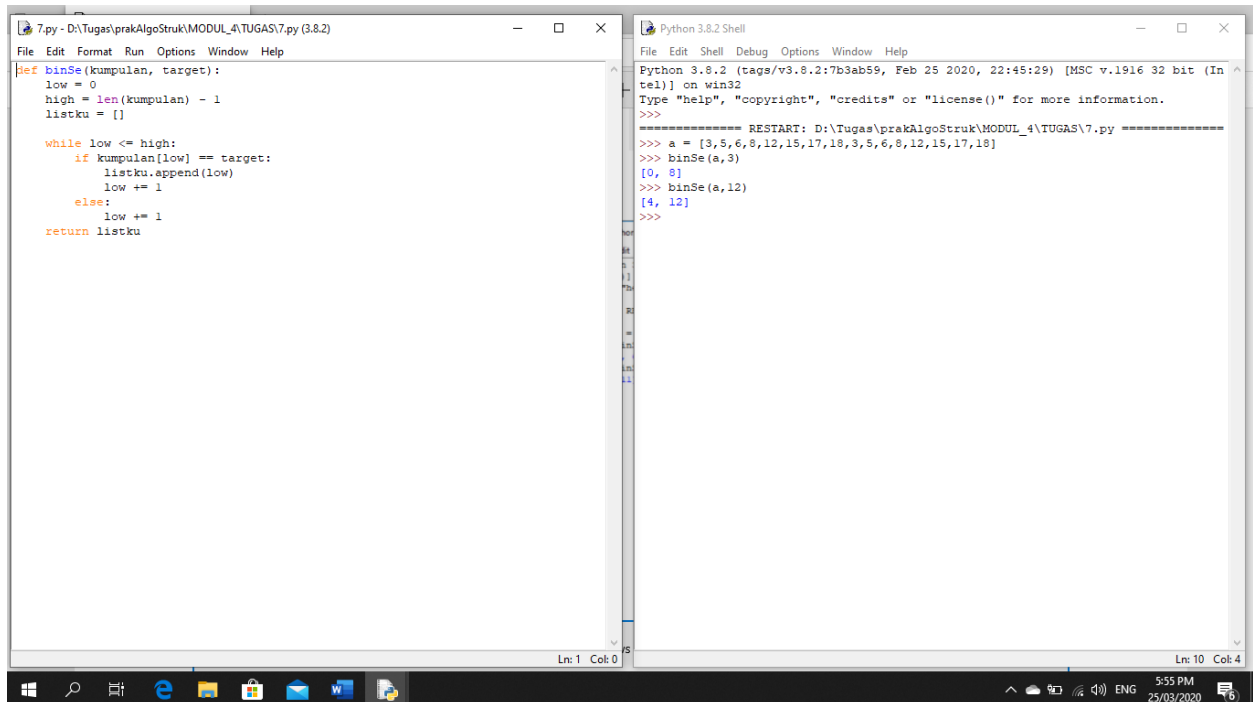
    while low <= high:
        mid = (high + low) // 2
        if kumpulan[mid] == target:
            return "target berada di index " + str(mid)
            break
        elif target < kumpulan[mid]:
            high = mid - 1
        else:
            low = mid + 1
    return False
```

The right window, titled 'Python 3.8.2 Shell', shows the execution of the code:

```
>>> list = [5,14,23,27,34,41]
>>> binSe(list, 27)
'target berada di index 3'
>>> binSe(list, 21)
False
>>>
```

The taskbar at the bottom shows the Windows Start button, search icon, and several application icons. The system tray on the right indicates the time is 5:50 PM on 25/03/2020.

7. Soal nomor 7



The screenshot shows a Python IDE with two windows. The left window is a script editor for '7.py' containing a binary search function. The right window is a Python 3.8.2 Shell showing the execution of the script.

```
def binSe(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

>>> a = [3, 5, 6, 8, 12, 15, 17, 18, 3, 5, 6, 8, 12, 15, 17, 18]
>>> binSe(a, 3)
[0, 8]
>>> binSe(a, 12)
[4, 12]
>>>
```

8. Soal : Pada permainan tebak angka, 1-100 dibutuhkan maksimal 7 kali tebakan untuk menemukan angka yang TEPAT. untuk angka 1-1000 dibutuhkan maksimal 10 kali tebakan. Mengapa demikian? Bagaimana polanya

Jawaban : 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 diebak 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"