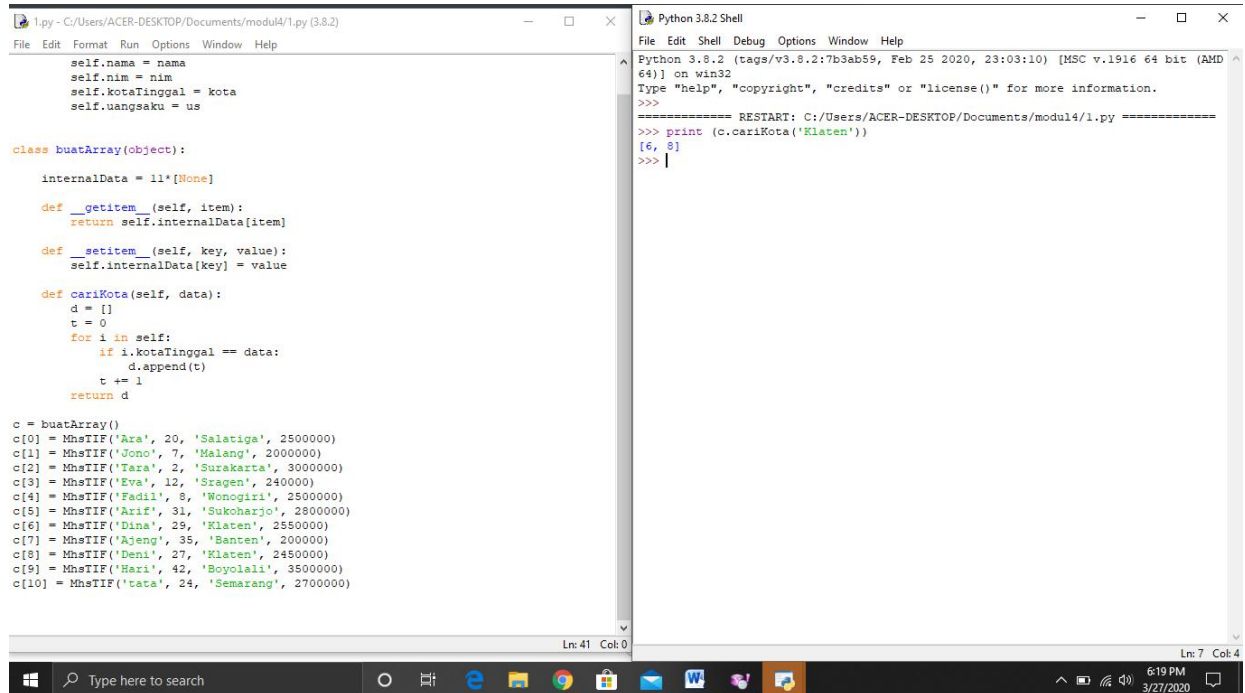


Nama : Fannisa RM

NIM : L200180048

Kelas : B

1.



```
1.py - C:/Users/ACER-DESKTOP/Documents/modul4/1.py (3.8.2)
File Edit Format Run Options Window Help

self.nama = nama
self.nim = nim
self.kotaTinggal = kota
self.uangsaku = us

class buatArray(object):
    internalData = {}

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

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

    def cariKota(self, data):
        d = []
        t = 0
        for i in self:
            if i.kotaTinggal == data:
                d.append(t)
                t += 1
        return d

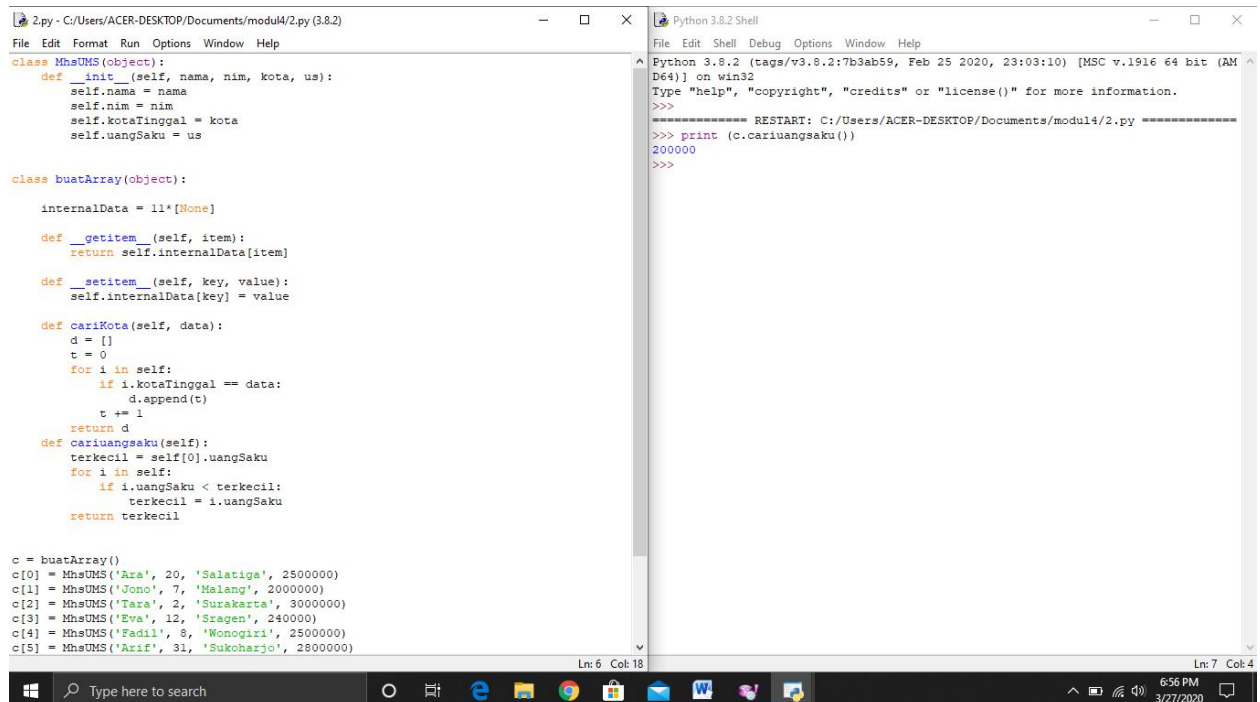
c = buatArray()
c[0] = MhsTIF('Ara', 20, 'Salatiga', 2500000)
c[1] = MhsTIF('Jono', 7, 'Malang', 2000000)
c[2] = MhsTIF('Tara', 2, 'Surakarta', 3000000)
c[3] = MhsTIF('Eva', 12, 'Sragen', 240000)
c[4] = MhsTIF('Fadil', 8, 'Wonogiri', 2500000)
c[5] = MhsTIF('Arif', 31, 'Sukoharjo', 2800000)
c[6] = MhsTIF('Dina', 29, 'Klaten', 2550000)
c[7] = MhsTIF('Ajeng', 35, 'Banten', 200000)
c[8] = MhsTIF('Deni', 27, 'Klaten', 2450000)
c[9] = MhsTIF('Hari', 42, 'Boyolali', 3500000)
c[10] = MhsTIF('tata', 24, 'Semarang', 2700000)

Ln: 41 Col: 0

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, 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/ACER-DESKTOP/Documents/modul4/1.py =====
>>> print(c.cariKota('Klaten'))
[6, 8]
>>>
```

2.



```
2.py - C:/Users/ACER-DESKTOP/Documents/modul4/2.py (3.8.2)
File Edit Format Run Options Window Help

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

class buatArray(object):
    internalData = {}

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

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

    def cariKota(self, data):
        d = []
        t = 0
        for i in self:
            if i.kotaTinggal == data:
                d.append(t)
                t += 1
        return d

    def cariuangSaku(self):
        terkecil = self[0].uangSaku
        for i in self:
            if i.uangSaku < terkecil:
                terkecil = i.uangSaku
        return terkecil

c = buatArray()
c[0] = MhsUMS('Ara', 20, 'Salatiga', 2500000)
c[1] = MhsUMS('Jono', 7, 'Malang', 2000000)
c[2] = MhsUMS('Tara', 2, 'Surakarta', 3000000)
c[3] = MhsUMS('Eva', 12, 'Sragen', 240000)
c[4] = MhsUMS('Fadil', 8, 'Wonogiri', 2500000)
c[5] = MhsUMS('Arif', 31, 'Sukoharjo', 2800000)

Ln: 6 Col: 18

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, 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/ACER-DESKTOP/Documents/modul4/2.py =====
>>> print(c.cariuangSaku())
200000
>>>
```

3.

The screenshot shows a Python IDE with a file named '3.py' and a Python 3.8.2 Shell. The code in the file defines a class 'buatArray' with methods for getting and setting items, and a 'cariKota' method. It also includes a list of data and a function 'cariuangkecil' that finds the minimum value in a list. The shell shows the execution of the code, including the output of the 'cariuangkecil' function.

```

class buatArray(object):
    internalData = {}

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

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

    def cariKota(self, data):
        d = {}
        t = 0
        for i in self:
            if i.kotaTinggal == data:
                d.append(i)
                t += 1
        return d

    def cariuangkecil(self):
        terkecil = self[0].uangSaku
        for i in self:
            if i.uangSaku < terkecil:
                terkecil = i.uangSaku
        return terkecil

    def cariuangkecil250k(self):
        terkecil = self[0].uangSaku
        d = {}
        for i in self:
            if i.uangSaku < terkecil:
                d.append((i.nama, i.nim, i.kotaTinggal, i.uangSaku))
        return d

c = buatArray()
c[0] = MhsUMS('Ara', 20, 'Salatiga', 2500000)
c[1] = MhsUMS('Jono', 7, 'Malang', 2000000)
c[2] = MhsUMS('Tara', 2, 'Surakarta', 3000000)
c[3] = MhsUMS('Eva', 12, 'Sragen', 240000)
c[4] = MhsUMS('Fadil', 8, 'Wonogiri', 2500000)
c[5] = MhsUMS('Arif', 31, 'Sukoharjo', 2800000)

# Execution in shell:
>>> print(c.cariuangkecil())
2000000
>>> print(c.cariuangkecil250k())
[('Jono', 7, 'Malang', 2000000), ('Eva', 12, 'Sragen', 240000), ('Ajeng', 35, 'Banten', 200000), ('Deni', 27, 'Klaten', 2450000)]
>>>
  
```

4.

The screenshot shows a Python IDE with a file named '4.py' and a Python 3.8.2 Shell. The code in the file defines a class 'buatArray' with methods for getting and setting items, and a 'cariKota' method. It also includes a list of data and a function 'cariuangkecil' that finds the minimum value in a list. The shell shows the execution of the code, including the output of the 'cariuangkecil' function.

```

class buatArray(object):
    internalData = {}

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

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

    def cariKota(self, data):
        d = {}
        t = 0
        for i in self:
            if i.kotaTinggal == data:
                d.append(i)
                t += 1
        return d

    def cariuangkecil(self):
        terkecil = self[0].uangSaku
        for i in self:
            if i.uangSaku < terkecil:
                terkecil = i.uangSaku
        return terkecil

    def cariuangkecil250k(self):
        terkecil = self[0].uangSaku
        d = {}
        for i in self:
            if i.uangSaku < terkecil:
                d.append((i.nama, i.nim, i.kotaTinggal, i.uangSaku))
        return d

    def cariuangkecil1250k(self):
        terkecil = 250000
        d = {}
        for i in self:
            if i.uangSaku < terkecil:
                d.append((i.nama, i.nim, i.kotaTinggal, i.uangSaku))
        return d

    def cariuangkecil1250k2(self):
        terkecil = 250000
        d = {}
        for i in self:
            if i.uangSaku < terkecil:
                d.append((i.nama, i.nim, i.kotaTinggal, i.uangSaku))
        return d

# Execution in shell:
>>> print(c.cariuangkecil1250k2())
None
>>>
  
```

5.

The screenshot shows a Python 3.8.2 IDE with two windows. The left window, titled '5.py - C:/Users/ACER-DESKTOP/Documents/modul4/5.py (3.8.2)', contains the following code:

```
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(13)
menu = a
a.next = node(20)
a = a.next
a.next = node(60)
a = a.next
a.next = node(70)
```

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

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, 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/ACER-DESKTOP/Documents/modul4/5.py =====
>>> menu.cariLinkedList(20)
Data  20 ada dalam linked list
>>> menu.cariLinkedList(40)
Data  40 tidak ada dalam linked list
>>>
```

The taskbar at the bottom shows the Windows search bar and various application icons. The system clock indicates 7:20 PM on 3/27/2020.

6.

The screenshot shows a Python 3.8.2 IDE with two windows. The left window, titled '6.py - C:/Users/ACER-DESKTOP/Documents/modul4/6.py (3.8.2)', contains the following code:

```
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 = [15, 32, 11, 22, 43, 80, 86]
target1 = 43
target2 = 40
```

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

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AM
D64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/ACER-DESKTOP/Documents/modul4/6.py =====
>>> print("listnya adalah ",listnya)
listnya adalah [15, 32, 11, 22, 43, 80, 86]
>>> print("nilai target adalah ", target1)
nilai target adalah 43
>>> print(binary(listnya, target1))
target berada di index 4
>>> print("nilai target adalah ", target2)
nilai target adalah 40
>>> print(binary(listnya, target2))
False
>>>
```

The taskbar at the bottom shows the Windows search bar and various application icons. The system clock indicates 7:29 PM on 3/27/2020.

7

7.py - C:/Users/ACER-DESKTOP/Documents/modul4/7.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 = [2,3,5,6,6,6,8,9,9,10,11,12,13,13,14]
dicari = 6
print("mencari angka ", dicari, " pada list ", s, "akan mengembalikan ",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, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/ACER-DESKTOP/Documents/modul4/7.py =====
>>>
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]
>>>

8

8.py - C:/Users/ACER-DESKTOP/Documents/modul4/8.py (3.8.2)
File Edit Format Run Options Window Help

```

print("""
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""")

print("""
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 Sn = 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"
""")

```

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
===== RESTART: C:/Users/ACER-DESKTOP/Documents/modul4/8.py =====
>>>
Soal :
Pada permainan tebak angka, 1-100 dibutuhkan maksimal 7 kali tebakan untu
k
menemukan angka yang TEPAT. untuk angka 1-1000 dibutuhkan
maksimal 10 kali tebakan. Mengapa demikian? 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 Sn = 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"

>>>