PRAKTIKUM

ALGORITMA DAN STRUKTUR DATA

(Algorithm and Data Structure)

LAPORAN TUGAS

MODUL 8

Nama : Shafa Bani Saputra

NIM : L200190151

Kelas : G

PROGRAM STUDI INFORMATIKA

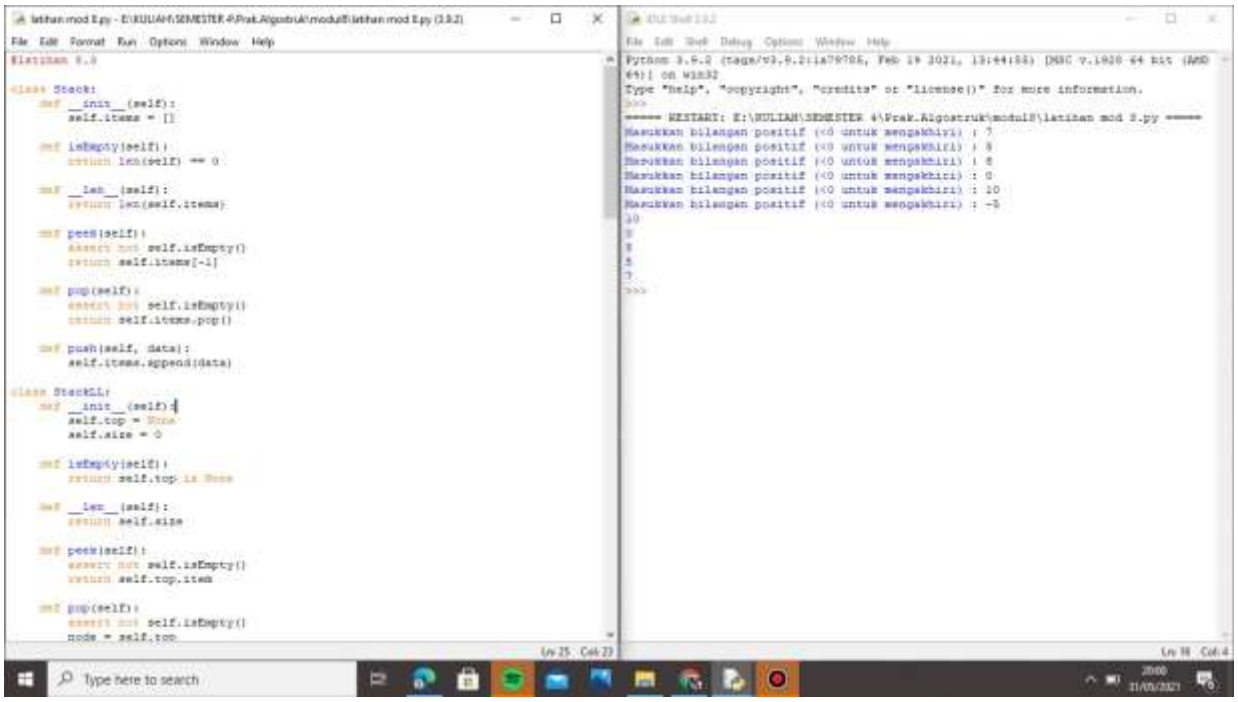
FAKULTAS KOMUNIKASI DAN INFORMATIKA

UNIVERSITAS MUHAMMADIYAH

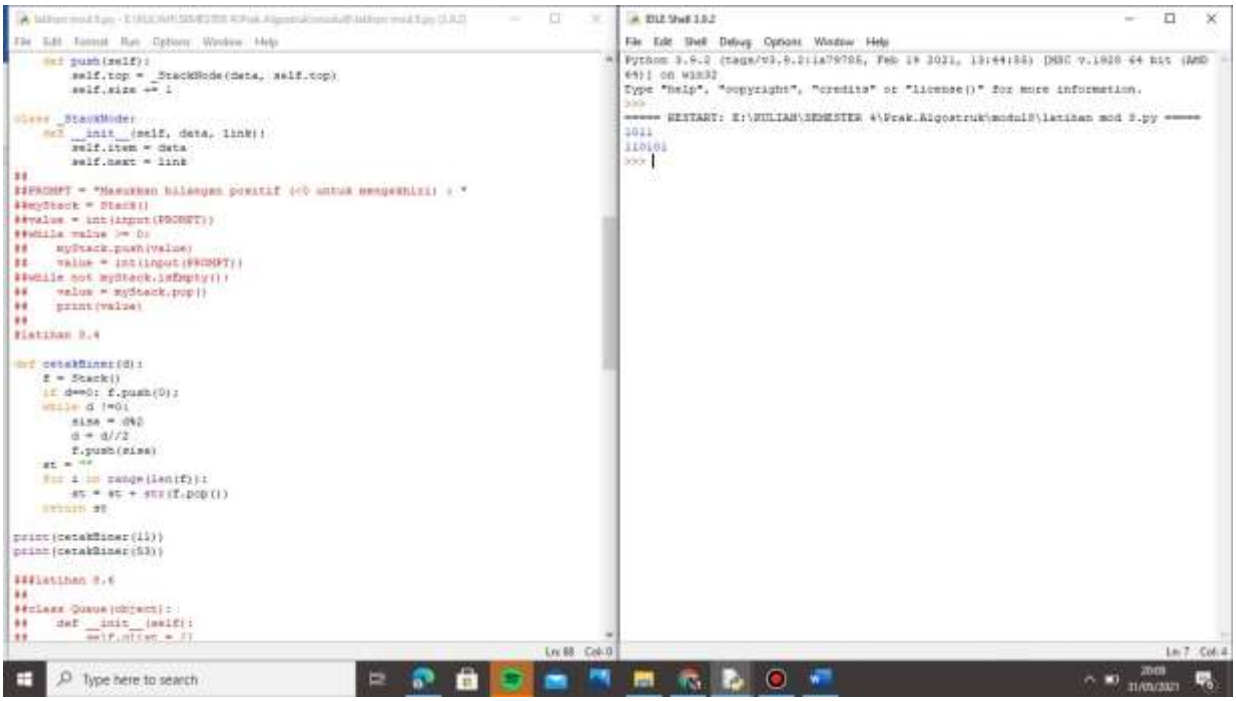
SURAKARTA

LATIHAN

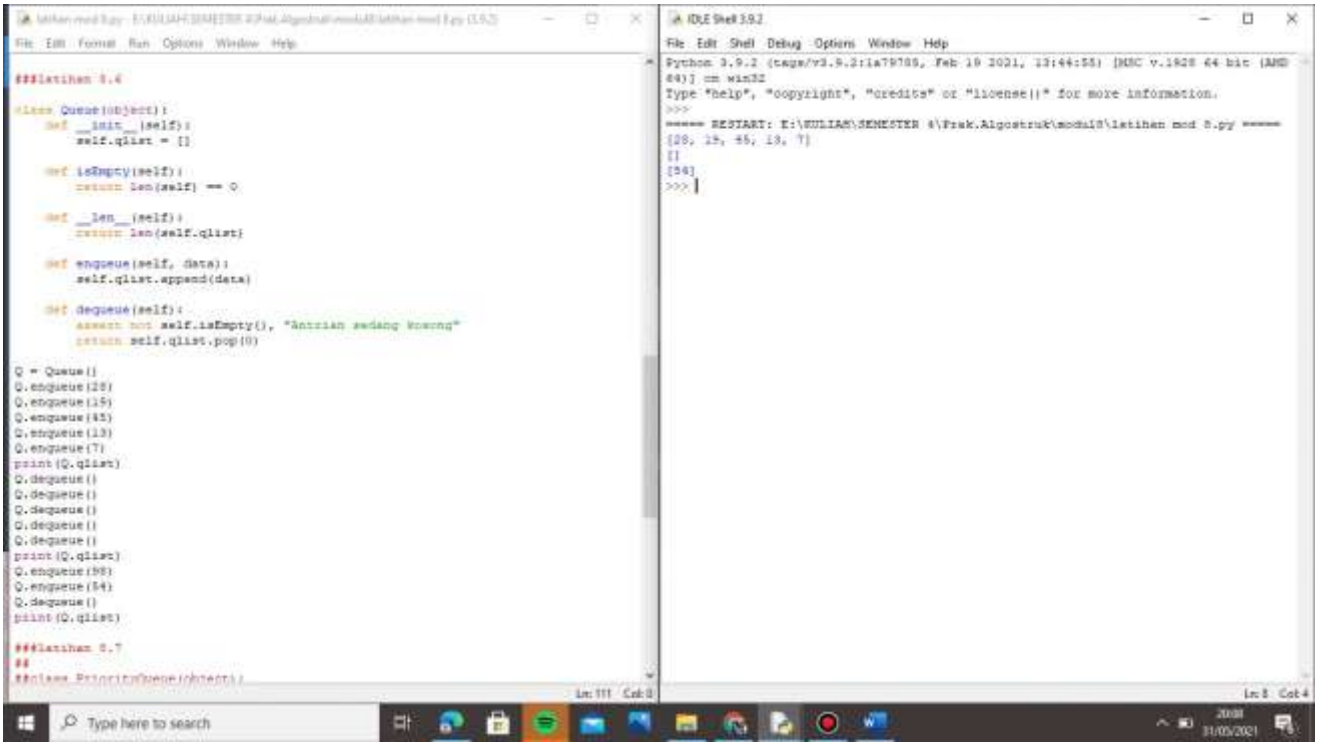
8.3



8.4



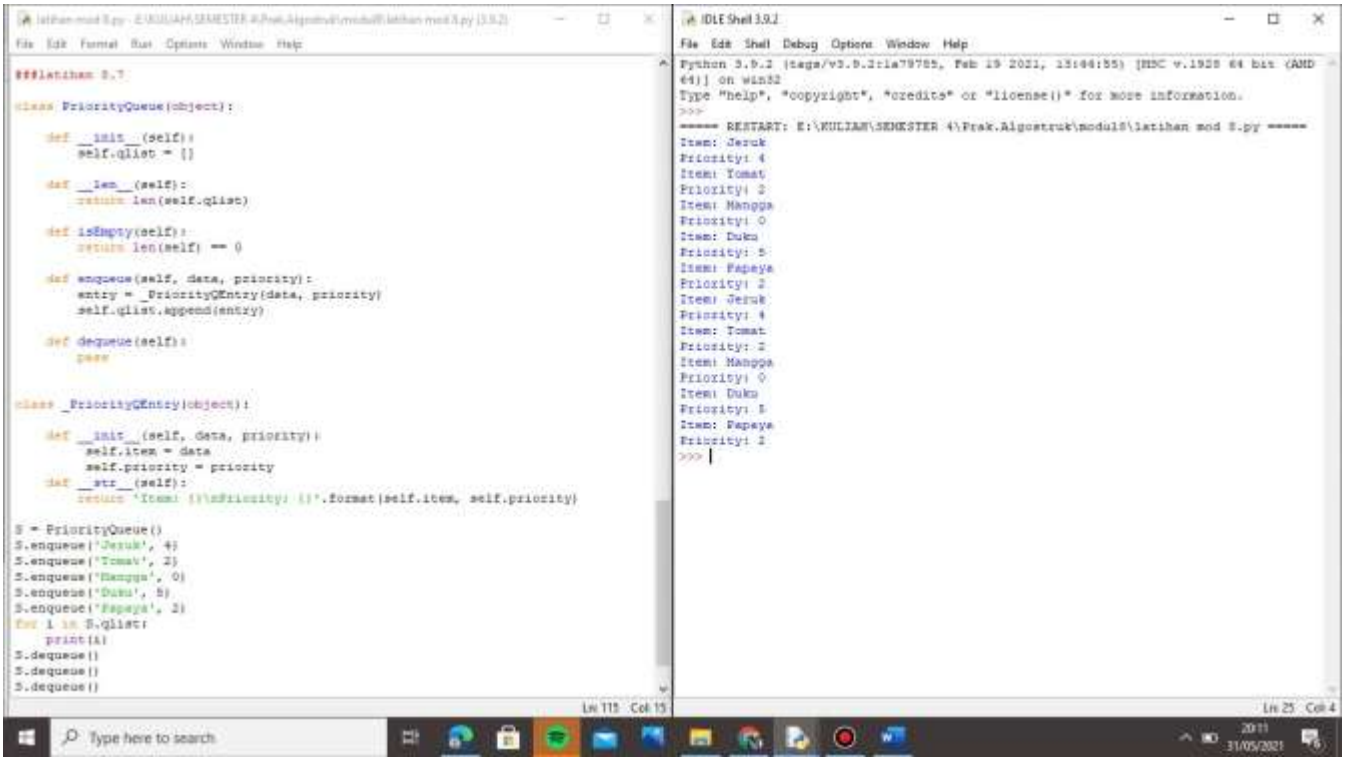
8.6



```
###latihan 8.6

class Queue(object):
    def __init__(self):
        self.qlist = []

    def isEmpty(self):
        return len(self) == 0

    def __len__(self):
        return len(self.qlist)

    def enqueue(self, data):
        self.qlist.append(data)

    def dequeue(self):
        assert not self.isEmpty(), "Antrian sedang kosong"
        return self.qlist.pop(0)

Q = Queue()
Q.enqueue(28)
Q.enqueue(15)
Q.enqueue(45)
Q.enqueue(13)
Q.enqueue(7)
print(Q.qlist)
Q.dequeue()
Q.dequeue()
Q.dequeue()
Q.dequeue()
Q.dequeue()
print(Q.qlist)
Q.enqueue(98)
Q.enqueue(54)
Q.dequeue()
print(Q.qlist)

###latihan 8.7
##
##class PriorityQueue(object):
```

```
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD
64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul8\latihan mod 8.py =====
[28, 15, 45, 13, 7]
[]
[54]
>>>
```

8.7

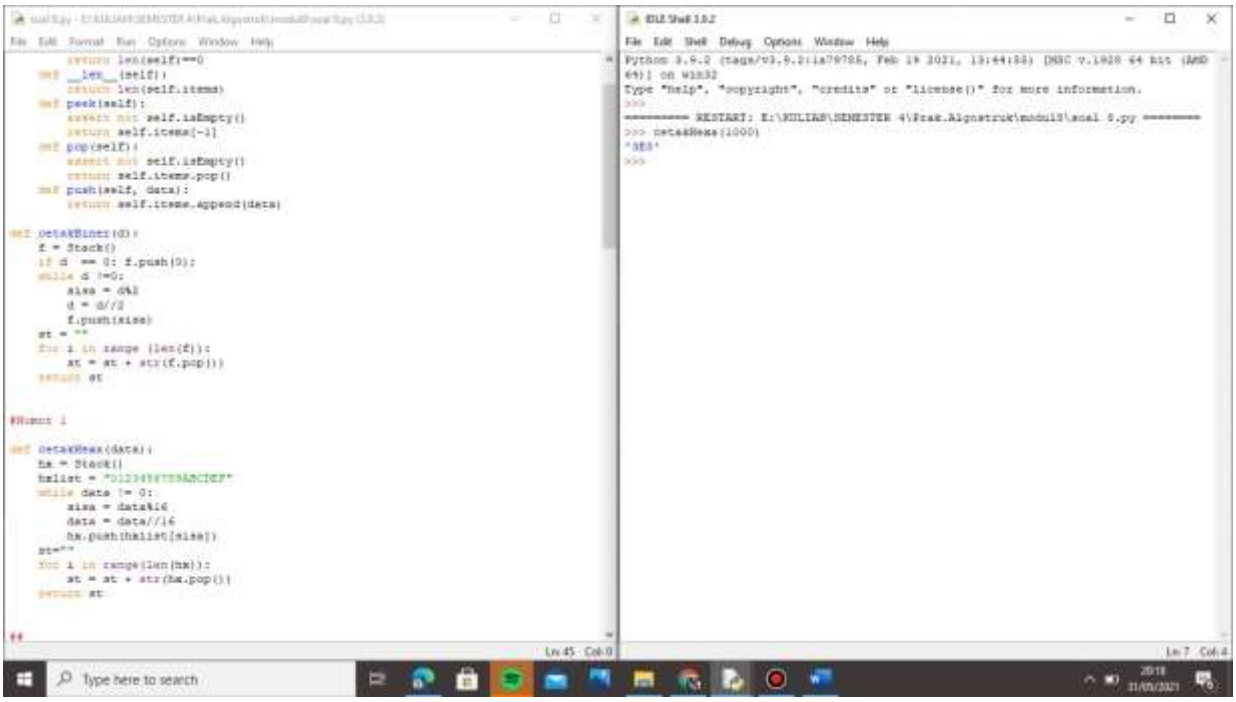


```
###latihan 8.7

class PriorityQueue(object):

    def __init__(self):
        self.qlist = []

    def __len__(self):
        return len(self.qlist)

    def isEmpty(self):
        return len(self) == 0

    def enqueue(self, data, priority):
        entry = _PriorityQEntry(data, priority)
        self.qlist.append(entry)

    def dequeue(self):
        pass


class _PriorityQEntry(object):

    def __init__(self, data, priority):
        self.item = data
        self.priority = priority
    def __str__(self):
        return 'Item: {}\nPriority: {}'.format(self.item, self.priority)

S = PriorityQueue()
S.enqueue('Jeruk', 4)
S.enqueue('Tomat', 2)
S.enqueue('Mangga', 0)
S.enqueue('Duku', 5)
S.enqueue('Papaya', 2)
for i in S.qlist:
    print(i)
S.dequeue()
S.dequeue()
S.dequeue()
```

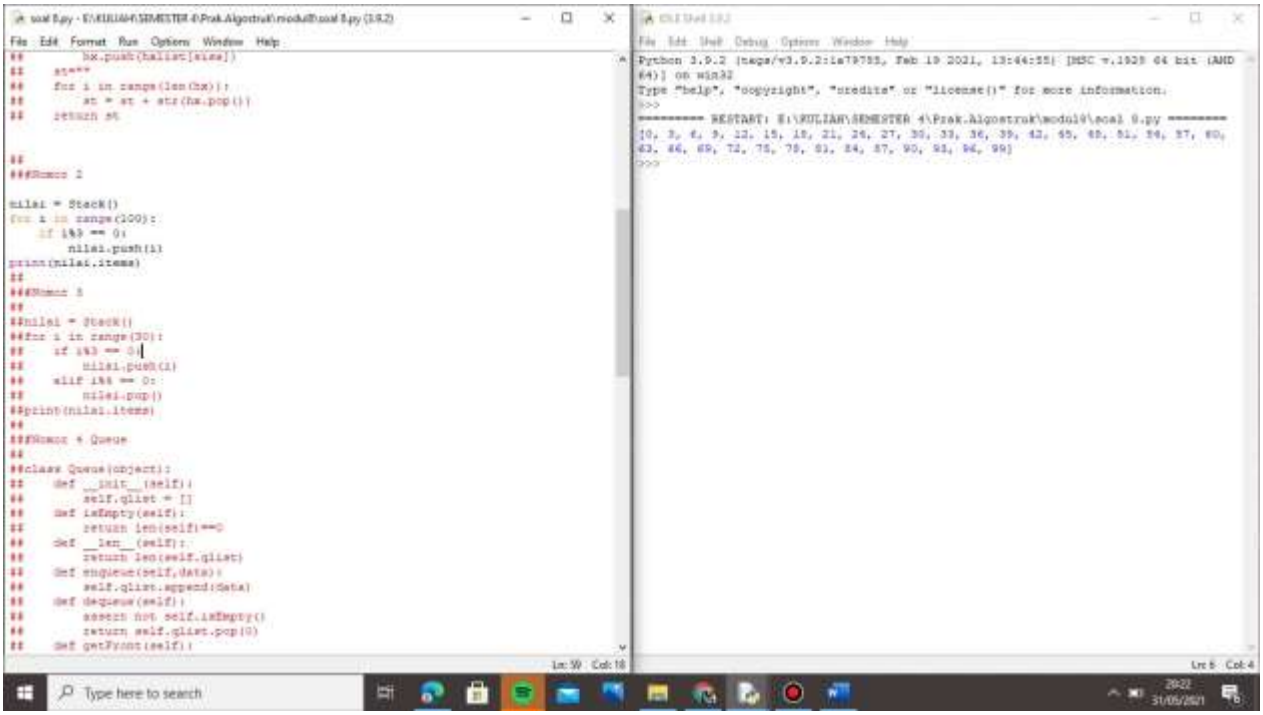
```
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD
64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul8\latihan mod 8.py =====
Item: Jeruk
Priority: 4
Item: Tomat
Priority: 2
Item: Mangga
Priority: 0
Item: Duku
Priority: 5
Item: Papaya
Priority: 2
Item: Jeruk
Priority: 4
Item: Tomat
Priority: 2
Item: Mangga
Priority: 0
Item: Duku
Priority: 5
Item: Papaya
Priority: 2
>>>
```

TUGAS

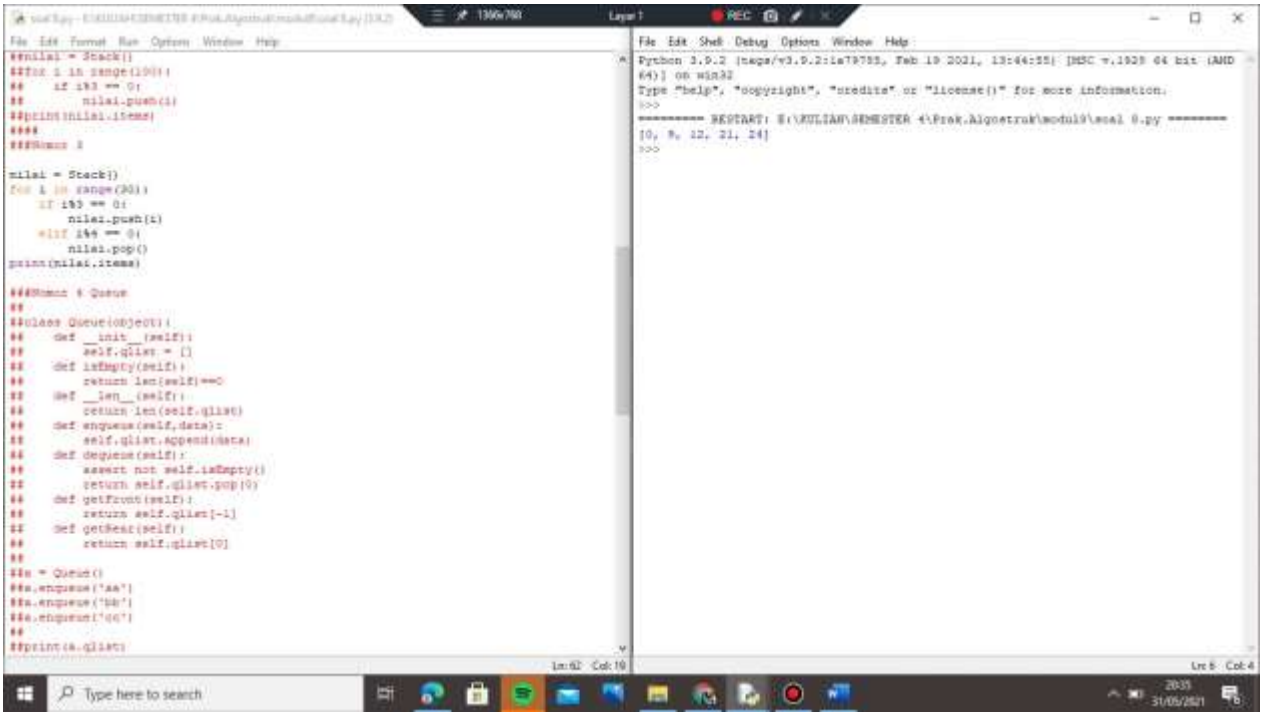
1.


File Edit Format Run Options Window Help
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD 64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========= RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul3\soal 6.py ========
>>> getaHexa(1000)
'3E8'
>>>
Type here to search

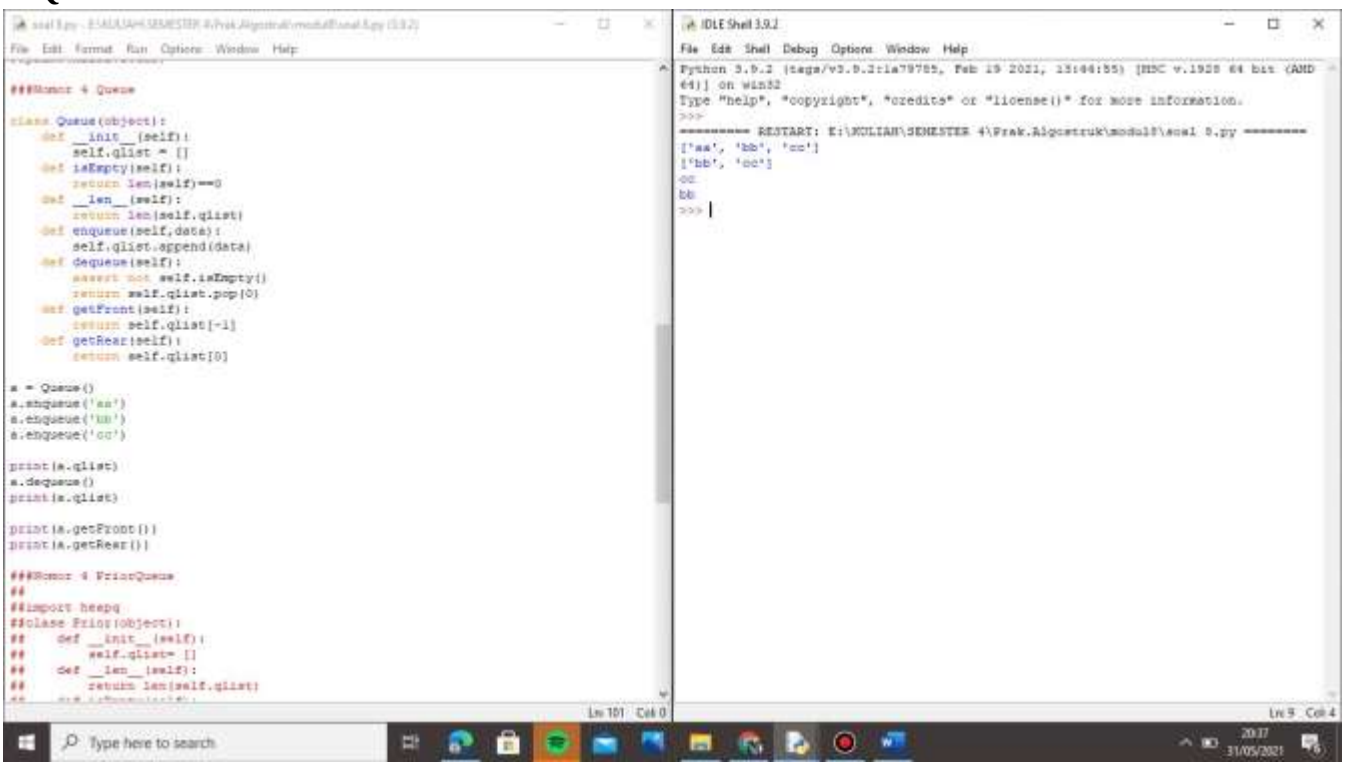
2.


File Edit Format Run Options Window Help
##Nomor 2
nilai = Stack()
for i in range(100):
if i%3 == 0:
nilai.push(i)
print(nilai.items)
##Nomor 3
##Nomor 4 Queue
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1929 64 bit (AMD 64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========= RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul9\soal 8.py =========
[0, 3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96, 99]
>>>

3.


File Edit Format Run Options Window Help
##Nomor 2
nilai = Stack()
for i in range(30):
if i%3 == 0:
nilai.push(i)
elif i%4 == 0:
nilai.pop()
print(nilai.items)
##Nomor 4 Queue
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1929 64 bit (AMD 64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========= RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul9\soal 8.py =========
[0, 9, 12, 21, 24]
>>>

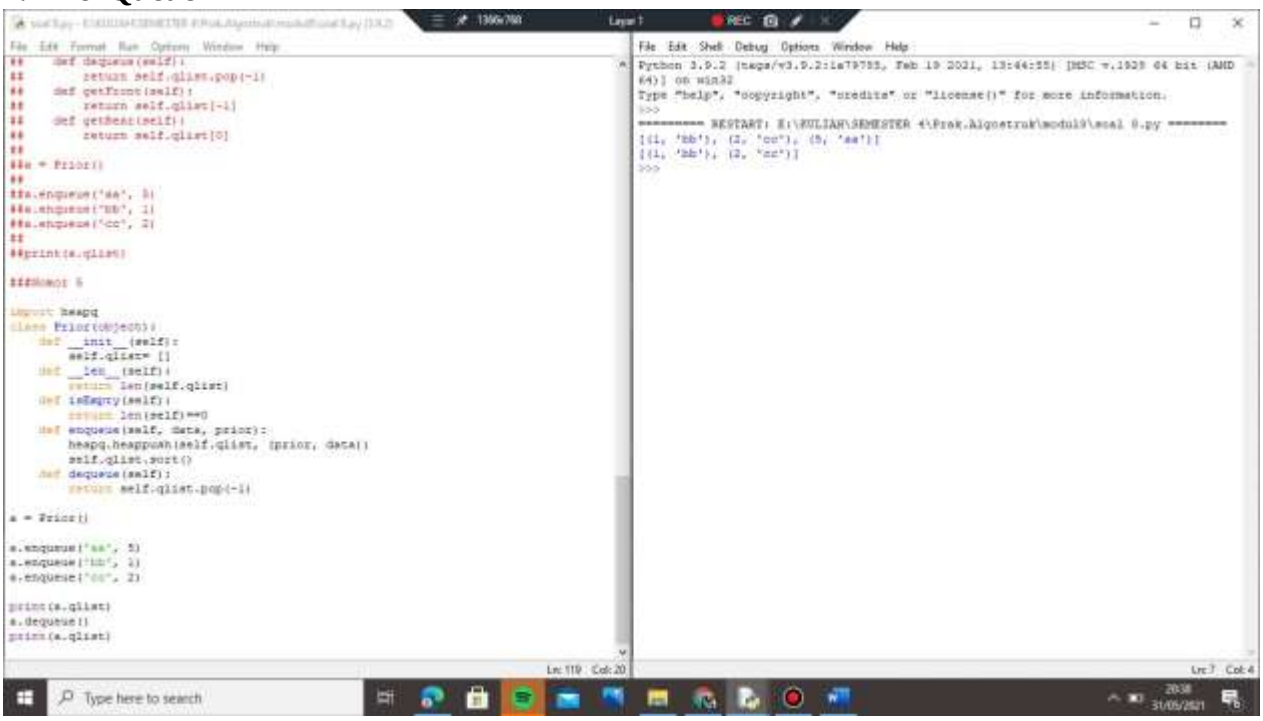
4. Queue



```
###Nomor 4 Queue

class Queue(object):
    def __init__(self):
        self.qlist = []
    def isEmpty(self):
        return len(self)==0
    def __len__(self):
        return len(self.qlist)
    def enqueue(self,data):
        self.qlist.append(data)
    def dequeue(self):
        assert not self.isEmpty()
        return self.qlist.pop(0)
    def getFront(self):
        return self.qlist[-1]
    def getRear(self):
        return self.qlist[0]

a = Queue()
a.enqueue('aa')
a.enqueue('bb')
a.enqueue('cc')

print(a.qlist)
a.dequeue()
print(a.qlist)

print(a.getFront())
print(a.getRear())

###Nomor 4 PriorQueue
##
##import heapq
##class Prior(object):
##    def __init__(self):
##        self.qlist= []
##    def __len__(self):
##        return len(self.qlist)
```

```
IDLE Shell 3.9.2
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD
64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========= RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul9\soal 8.py =======
['aa', 'bb', 'cc']
['bb', 'cc']
cc
bb
>>>
```

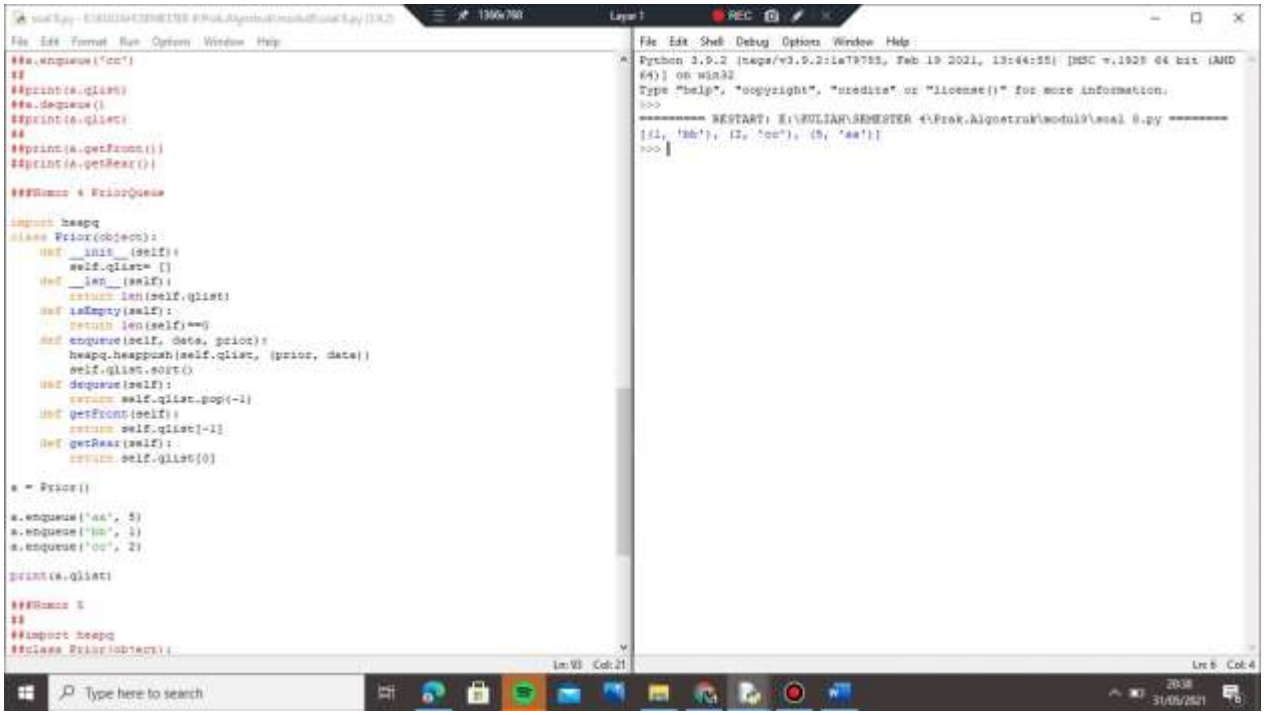
4. PriorQueue



```
##    def dequeue(self):
##        return self.qlist.pop(-1)
##    def getFront(self):
##        return self.qlist[-1]
##    def getRear(self):
##        return self.qlist[0]
##
##a = Prior()
##
##a.enqueue('aa', 5)
##a.enqueue('bb', 1)
##a.enqueue('cc', 2)
##
##print(a.qlist)

###Nomor 5

import heapq
class Prior(object):
    def __init__(self):
        self.qlist= []
    def __len__(self):
        return len(self.qlist)
    def isEmpty(self):
        return len(self)==0
    def enqueue(self, data, prior):
        heapq.heappush(self.qlist, (prior, data))
        self.qlist.sort()
    def dequeue(self):
        return self.qlist.pop(-1)

a = Prior()

a.enqueue('aa', 5)
a.enqueue('bb', 1)
a.enqueue('cc', 2)

print(a.qlist)
a.dequeue()
print(a.qlist)
```

```
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD
64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========= RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul9\soal 8.py =======
[(1, 'bb'), (2, 'cc'), (5, 'aa')]
[(1, 'bb'), (2, 'cc')]
>>>
```

5.



```
##a.enqueue('cc')
##
##print(a.qlist)
##a.dequeue()
##print(a.qlist)
##
##print(a.getFront())
##print(a.getRear())

###Nomor 4 PriorQueue

import heapq
class Prior(object):
    def __init__(self):
        self.qlist= []
    def __len__(self):
        return len(self.qlist)
    def isEmpty(self):
        return len(self)==0
    def enqueue(self, data, prior):
        heapq.heappush(self.qlist, (prior, data))
        self.qlist.sort()
    def dequeue(self):
        return self.qlist.pop(-1)
    def getFront(self):
        return self.qlist[-1]
    def getRear(self):
        return self.qlist[0]

a = Prior()

a.enqueue('aa', 5)
a.enqueue('bb', 1)
a.enqueue('cc', 2)

print(a.qlist)

###Nomor 5
##
##import heapq
##class Prior(object):
```

```
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1929 64 bit (AMD
64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: E:\KULIAH\SEMESTER 4\Prak.Algostruk\modul9\soal 8.py ========
[(1, 'bb'), (2, 'cc'), (5, 'aa')]
>>>
```