

**PRAKTIKUM ALGORITMA DAN
STRUKTUR DATA**

Modul 8

Stack and Queues



Disusun oleh:

DONI WAHYU SAPUTRO

L200200169

G

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS KOMUNIKASI DAN INFORMATIKA

UNIVERSITAS MUHAMMADIYAH SURAKARTA

Tugas

1.

```
no1.py - C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no1.py (3.8.5)
File Edit Format Run Options Window Help
#DoniWahyuSaputro
#L200200169
class Stack(object):
    def __init__(self):
        self.items = []

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

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

    def peek(self):
        assert not self.isEmpty(), "Tidak bisa diintip. Stack kosong"
        return self.items[-1]

    def pop(self):
        assert not self.isEmpty(), "Tidak bisa dipop dari Stack kosong"
        return self.items.pop()

    def push(self, data):
        self.items.append(data)

def cetakHexa(d):
    f = Stack()
    if d == 0: f.push(0)
    while d != 0:
        sisa = d%16
        d = d//16
        if sisa == 10:
            sisa = "A"
        elif sisa == 11:
            sisa = "B"
        elif sisa == 12:
            sisa = "C"
        elif sisa == 13:
            sisa = "D"
        elif sisa == 14:
            sisa = "E"
        elif sisa == 15:
            sisa = "F"
        f.push(sisa)
    st = ""
    for i in range(len(f)):
        st = st + str(f.pop())
    return st

Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no1.py =====
>>> cetakHexa(12)
'C'
>>> cetakHexa(31)
'1F'
>>> cetakHexa(229)
'E5'
>>> cetakHexa(255)
'FF'
>>> cetakHexa(31519)
'7B1F'
>>>
```

2.

```
no2.py - C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no2.py (3.8.5)
File Edit Format Run Options Window Help
#DoniWahyuSaputro
#L200200169
class Stack(object):
    def __init__(self):
        self.items = []

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

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

    def peek(self):
        assert not self.isEmpty(), "Tidak bisa diintip. Stack kosong"
        return self.items[-1]

    def pop(self):
        assert not self.isEmpty(), "Tidak bisa dipop dari Stack kosong"
        return self.items.pop()

    def push(self, data):
        self.items.append(data)

nilai = Stack()
for i in range(16):
    if i%3 == 0:
        nilai.push(i)
print(nilai.items)

Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no2.py =====
>>>
[0, 3, 6, 9, 12, 15]
>>>
```

3.

The screenshot shows a Python IDE with two windows. The left window, titled 'no3.py - C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no3.py (3.8.5)', contains the following code:

```
#DoniWahyuSaputro
#L200200169

class Stack(object):
    def __init__(self):
        self.items = []

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

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

    def peek(self):
        assert not self.isEmpty(), "Tidak bisa diintip. Stack kosong"
        return self.items[-1]

    def pop(self):
        assert not self.isEmpty(), "Tidak bisa dipop dari Stack kosong"
        return self.items.pop()

    def push(self, data):
        self.items.append(data)

nilai = Stack()
for i in range(16):
    if i%3 == 0:
        nilai.push(i)
    elif i%4 == 0:
        nilai.pop()
print(nilai.items)
```

The right window, titled 'Python 3.8.5 Shell', shows the execution output:

```
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no3.py =====
[0, 9, 12, 15]
>>>
```

4.

The screenshot shows a Python IDE with two windows. The left window, titled 'no4.py - C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no4.py (3.8.5)', contains the following code:

```
#DoniWahyuSaputro
#L200200169

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)

    def getFrontMost(self):
        return self.qlist[0]

    def getRearMost(self):
        return self.qlist[-1]

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

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

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

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

    def getFrontMost(self):
        x = 0
        while self.qlist[x].priority != 0:
            x+=1
        return self.qlist[x].item

    def getRearMost(self):
        a = []
        for i in self.qlist:
            a.append(i.priority)
        print (self.qlist[a.index(max(a))].item)

class _PriorityEntry(object):
    def __init__(self, data, priority):
        self.item = data
        self.priority = priority

A = Queue()
```

The right window, titled 'Python 3.8.5 Shell', shows the execution output:

```
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no4.py =====
>>> A.getFrontMost()
20
>>> A.getRearMost()
Traceback (most recent call last):
  File "<pyshell#1>", line 1, in <module>
    A.getRearMost()
AttributeError: 'Queue' object has no attribute 'getRearMost'
>>> A.getRearMost()
7
>>> S.getFrontMost()
'Mangga'
>>> S.getRearMost()
Duku
>>>
```

```

A = Queue()
A.enqueue(28)
A.enqueue(19)
A.enqueue(45)
A.enqueue(13)
A.enqueue(7)

S = PriorityQueue()
S.enqueue("Jeruk", 4)
S.enqueue("Tomat", 2)
S.enqueue("Mangga", 0)
S.enqueue("Duku", 5)
S.enqueue("Pepaya", 2)

```

5.

```

no5.py - C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no5.py (3.8.5)
File Edit Format Run Options Window Help
#DoniWahyuSaputro
#L200200169

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

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

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

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

    def dequeue(self):
        assert not self.isEmpty(), "Antrian sedang kosong"
        a = []
        for i in self.qlist:
            a.append(i.priority)
        print (self.qlist.pop(a.index(min(a))).item)

class _PriorityEntry(object):
    def __init__(self, data, priority):
        self.item = data
        self.priority = priority

S = PriorityQueue()
S.enqueue("Jeruk", 4)
S.enqueue("Tomat", 2)
S.enqueue("Mangga", 0)
S.enqueue("Duku", 4)
S.enqueue("Pepaya", 2)

```

```

Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no5.py =====
>>> S.dequeue()
Mangga
>>> S.dequeue()
Tomat
>>> S.dequeue()
Pepaya
>>> S.dequeue()
Jeruk
>>> S.dequeue()
Duku
>>> S.dequeue()
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    S.dequeue()
  File "C:/Kuliah/SEMESTER 4/Praktikum ASD/modul 8/no5.py", line 19, in dequeue
    assert not self.isEmpty(), "Antrian sedang kosong"
AssertionError: Antrian sedang kosong
>>>

```