Nama : Rifqi Alwan P
NIM : L200180008
Kelas A

Soal - soal Mahasiswa

1. Program mengubah representasi bilangan dari sepuluh ke basis dua

```
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(), "Stack kosong. Tidak bisa diintip"
        return self.items[-1]
    def pop(self):
        assert not self.isEmpty(), "Stack kosong. Tidak bisa di-pop"
        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:
        if d % 16 == 10:
            sisa = 'A'
        elif d % 16 == 11:
            sisa = 'B'
        elif d % 16 == 12:
            sisa = 'C'
        elif d % 16 == 13:
            sisa = 'D'
        elif d % 16 == 14:
            sisa = 'E'
        elif d % 16 == 15:
            sisa = 'F'
        else:
            sisa = d % 16
        d = d // 16
        f.push(sisa)
    st = ""
    for i in range(len(f)):
        st = st + str(f.pop())
    return st
```

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: E:\TUGAS TUGAS KULIAH\SEMESTER 4\Praktikum Algoritma Struktur
DUL_8\soal01.py
>>> cetakHexa(12)
'C'
>>> cetakHexa(31)
'1F'
>>> cetakHexa(229)
'E5'
>>> cetakHexa(255)
'FF'
>>> cetakHexa(31519)
'7B1F'
>>>
```

2. Eksekusi program dengan pensil dan kertas

```
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(), "Stack kosong. Tidak bisa diintip"
        return self.items[-1]

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

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


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

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (In
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: E:\TUGAS TUGAS KULIAH\SEMESTER 4\Praktikum Algoritma Struktur data\MO
DUL_8\soal02.py
0
[0]
1
[0]
2
[0]
3
[0, 3]
4
[0, 3]
5
[0, 3]
6
[0, 3, 6]
7
[0, 3, 6]
8
[0, 3, 6]
9
[0, 3, 6, 9]
10
[0, 3, 6, 9]
11
[0, 3, 6, 9]
12
[0, 3, 6, 9, 12]
13
[0, 3, 6, 9, 12]
14
[0, 3, 6, 9, 12]
15
[0, 3, 6, 9, 12, 15]
```

3. Eksekusi program dengan pensil dan kertas

```
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(), "Stack kosong. Tidak bisa diintip"
        return self.items[-1]
    def pop(self):
        assert not self.isEmpty(), "Stack kosong. Tidak bisa di-pop"
        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(i, 'stack:', nilai.items)
```

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (In
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: E:\TUGAS TUGAS KULIAH\SEMESTER 4\Praktikum Algoritma Struktur data\MO
DUL_8\soal03.py
0 stack: [0]
1 stack: [0]
2 stack: [0]
3 stack: [0, 3]
4 stack: [0]
5 stack: [0]
6 stack: [0, 6]
7 stack: [0, 6]
8 stack: [0]
9 stack: [0, 9]
10 stack: [0, 9]
11 stack: [0, 9]
12 stack: [0, 9, 12]
13 stack: [0, 9, 12]
14 stack: [0, 9, 12]
15 stack: [0, 9, 12, 15]
>>>
```

4. Class Queue dan PriorityQueue

```
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)

    # 4a soal-soal untuk mahasiswa
    def getFrontMost(self):
        return self.qlist[0]

    # 4b soal-soal untuk mahasiswa
    def getRearMost(self):
        return self.qlist[len(self.qlist) - 1]
```

```
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: E:\TUGAS TUGAS KULIAH\SEMESTER 4\Praktikum Algoritma Struktur data\MO
DUL_8\soal04.py
>>> A = Queue()
>>> A.enqueue(12)
>>> A.enqueue(23)
>>> A.enqueue(39)
>>> A.enqueue(3)
>>> A.enqueue(7)
>>> A.enqueue(37)
>>> A.qlist
[12, 23, 39, 3, 7, 37]
>>> A.getFrontMost()
12
>>> A.getRearMost()
37
>>>
```

5. Metode dequeue

```
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)

    # Soal 5
    def dequeue(self):
        n = []
        for i in self.qlist:
            n.append(i.priority)
        print(self.qlist.pop(n.index(min(n))).item)

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

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (In
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: E:\TUGAS TUGAS KULIAH\SEMESTER 4\Praktikum Algoritma Struktur data\MO
DUL_8\soal05.py
>>> P = PriorityQueue()
>>> P.enqueue('Jeruk', 4)
>>> P.enqueue('Tomat', 2)
>>> P.enqueue('Mangga', 0)
>>> P.enqueue('Duku', 5)
>>> P.enqueue('Pepaya', 2)
>>> P.dequeue()
Mangga
>>> P.dequeue()
Tomat
>>> P.dequeue()
Pepaya
>>>
```