

# LAPORAN PRAKTIKUM

## ALGORITMA DAN STRUKTUR DATA

Nama : Fatwa Dwi Hidayat

Nim : L200180176

Kelas : G

### 1. Membuat suatu fungsi cetakSiku(x)

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python:
File Edit Format Run Options Window Help
def cetakSiku(x):
    for i in range (1,x+1):
        print ('*' * i)

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020
tel)] on win32
Type "help", "copyright", "credits" or "licens
>>>
===== RESTART: C:/Users/asus/AppData/Local
py =====
>>> cetakSiku(5)
*
**
***
****
*****
>>> |
```

### 2. Membuat fungsi yang menerima dua integer positif, yang akan menggambar bentuk persegi empat

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python
File Edit Format Run Options Window Help
def gambarlahPersegiEmpat(a,b):
    for i in range (a):
        if i==0 or i==a-1:
            print (b * '@')
        else:
            print ('@' + ' ' * (b-2) + '@')

|

=== RESTART: C:/Users/asus/AppData/Local/
>>> gambarlahPersegiEmpat(4,5)
@@@@@
@  @
@  @
@@@@@
>>>
```

3. a.) membuat sebuah fungsi yang menerima sebuah string dan mengembalikan sebuah list yang terdiri dari 2 integer. Dua integer kembalian ini adalah: jumlah huruf di string itu dan jumlah huruf vocal di string itu.

```
coba.py - C:/Users/asus/AppData/Local/Programs/Pyth
File Edit Format Run Options Window Help
def jumlahHurufVokal (huruf):
    vokal = 'aiueoAIUEO'
    a = 0
    hasil = 0
    for i in huruf:
        if i in vokal:
            a += len(i)
        else:
            a +=0

=== RESTART: C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
>>> k = jumlahHurufVokal('Surakarta')
>>> k
(9, 4)
>>>
```

b.) menghitung huruf konsonannya

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
File Edit Format Run Options Window Help
def jumlahHurufKonsonan (huruf):
    konsonan = 'bcd fghjklmnpqrstvwxyzBCDFGHJKLMPQ RSTVWXY Z'
    b = 0
    hasil = 0
    for i in huruf:
        if i in konsonan:
            b +=len(i)
        else:
            b += 0
    hasil = len(huruf),b
    return hasil

=== RESTART: C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
>>> k = jumlahHurufKonsonan('Surakarta')
>>> k
(9, 5)
>>>
```

4. menghitung rerata sebuah array yang berisi bilangan.

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
File Edit Format Run Options Window Help
def rerata(b):
    return sum(b)/len(b)

=== RESTART: C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
>>> rerata([1,2,3,4,5])
3.0
>>> g = [3,4,5,4,3,4,5,2,2,10,11,23]
>>> rerata(g)
6.333333333333333
>>>
```

5. menentukan suatu bilangan bulat adalah bilangan prima atau bukan.

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python
File Edit Format Run Options Window Help
from math import sqrt as sq
def apakahPrima(n):
    n = int(n)
    assert n>=0
    primaKecil = [2,3,5,7,9,11]
    bukanPrKecil = [0,1,4,6,8,9,10]
    if n in primaKecil:
        return True
    elif n in bukanPrKecil:
        return False
    else:
        for i in range(2,int(sq(n))+1):
            if n%i==0:
                return False
        return True

=== RESTART: C:/Users/asus/AppData/Local/Python
>>> apakahPrima(17)
True
>>> apakahPrima(97)
True
>>> apakahPrima(123)
False
>>>
```

6. program mencetak semua bilangan prima dari 2 sampai 1000.

```
coba.py - C:/Users/asus/AppData/Local/Program
File Edit Format Run Options Window Help
def bilanganPrima(n):
    for i in range(2,n):
        prima = True
        for j in range(2,i):
            if(i%j==0):
                prima = False
        if (prima):
            print(i)

=== RESTART: C:/Users/asus/AppData/Local/Python
>>> bilanganPrima(1000)
2
3
5
7
11
13
17
19
23
```

7. program yang menerima bilangan bulat positif dan memberikan faktorisasi prima-nya.

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python
File Edit Format Run Options Window Help
bilanganList = []
loop = 2
while loop <= x:
    if x%loop == 0:
        x/=loop
        bilanganList.append(loop)
    else:
        loop +=1
return bilanganList

=== RESTART: C:/Users/asus/AppData/Local/Python
>>> faktorPrima(10)
[2, 5]
>>> faktorPrima(120)
[2, 2, 2, 3, 5]
>>> faktorPrima(19)
[19]
>>>
```

8. membuat suatu fungsi apakahTerandung(a,b) yang menerima dua string a dan b, lalu menentukan apakah String a terhubung dalam string b.

```
coba.py - C:/Users/asus/AppData/Local/Programs/
File Edit Format Run Options Window Help
def apakahTerkandung(a,b):
    x = True
    for i in range(len(b)):
        if a in b:
            x = True
        else:
            x = False
    return x

=== RESTART: C:/Users/asus/AppData/L
>>> h = 'do'
>>> k = 'Indonesia tanah air beta'
>>> apakahTerkandung(h,k)
SyntaxError: invalid syntax
>>> apakahTerkandung(h,k)
True
>>> apakahTerkandung('pusaka',k)
False
>>>
```

9. program untuk mencetak angka dari 1 sampai 100. Kalau angkanya pas kelipatan 3 cetak 'Python'. Kalau pas kelipatan 5 cetak 'UMS'. Jika sekaligus kelipatan 3 dan 5 cetak 'Python UMS'.

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
=== RESTART: C:/Users/asus/AppData/Loc
>>> h = 'do'

coba.py - C:/Users/asus/AppData/Local/Programs/P
File Edit Format Run Options Window Help
def kelipatan(x):
    for i in range(x):
        if(i%3==0):
            pass
            elif(i%3==0 and i%5==0):
                print ('Python UMS')
            elif(i%3==0):
                print ('Python')
            elif(i%5==0):
                print ('UMS')
            else:
                print (i)

>>> kelipatan(100)
1
2
Python
4
UMS
Python
7
8
Python
UMS
11
Python
13
14
Python UMS
```

10. modifikasi Contoh 1.4, agar bisa menangkap kasus di mana determinannya kurang dari nol.

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
File Edit Format Run Options Window Help
from math import sqrt as akar
def selesaikanABC(a,b,c):
    a = float(a)
    b = float(b)
    c = float(c)
    D = float(b**2 - 4*a*c)
    if (D<0):
        hasil = "Determinannya negatif, persamaan tidak mempunyai akar real."
        return hasil
    else:
        x1 = (-b + akar(D))/(2*a)
        x2 = (-b - akar(D))/(2*a)
        hasil = (x1,x2)
        return hasil

>>> selesaikanABC(1,2,3)
'Determinannya negatif, persamaan tidak mempunyai akar real.'
```

11. membuat suatu fungsi apakahKabisat() yang menerima suatu angka (tahun). Jika tahun itu kabisat, kembalikan True. jika bukan, kembalikan False.

```
*coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba
File Edit Format Run Options Window Help
def apakahKabisat(tahun):
    hasil = False
    if (tahun%4==0 and tahun%100!=0 and tahun%400!=0):
        hasil = True
    elif (tahun%100==0 and tahun%400!=0):
        hasil = False
    elif (tahun%400==0):
        hasil = True
    else:
        hasil = False
    return hasil

=== RESTART: C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py
>>> apakahKabisat(2020)
True
>>> apakahKabisat(2019)
False
>>>
```

12. membuat permainan tebak angka

```
*coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
File Edit Format Run Options Window Help
import random
def tebak():
    a = random.randrange(1,101,1)
    b = -1
    n = 0
    print ('Permainan Tebak Angka')
    print ('Saya Menyimpan Sebuah Angka Bulat Antara 1 Sampai 100.')
    while a != b:
        n = n + 1
        b = int (input('Masukkan tebakan ke- ' + str(n) + ' :> '))
        if b < a:
            print ('Itu Terlalu Kecil, Coba Lagi')
        elif b > a:
            print ('Itu Terlalu Besar, Coba Lagi')
        else:
            print ('Ya, Anda Benar')
            break

>>> tebak()
Permainan Tebak Angka
Saya Menyimpan Sebuah Angka Bulat Antara 1 Sampai 100.
Masukkan tebakan ke- 1:> 2
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 2:> 1
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 3:> 101
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 4:> 58
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 5:> 10
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 6:> 10
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 7:> 12
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 8:> 15
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 9:> 20
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 10:> 18
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 11:> 17
Ya, Anda Benar
```

13. membuat suatu fungsi katakan() yang menerima bilangan bulat positif dan mengembalikan suatu string yang merupakan pengucapan angka itu dalam Bahasa Indonesia.

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python38-32/coba.py (3.8.2)
File Edit Format Run Options Window Help

def katakan(bilangan):
    angka=['','Satu','Dua','Tiga','Empat','Lima','Enam',
           'Tujuh','Delapan','Sembilan','Sepuluh','Sebelas']
    Hasil = ' '
    n = int(bilangan)
    if (n >= 0 and n <= 11):
        Hasil = Hasil + angka[n]
    elif (n < 20):
        Hasil = katakan(n % 10) + ' Belas'
    elif (n < 100):
        Hasil = katakan(n / 10) + ' Puluh' + katakan(n % 10)
    elif (n < 200):
        Hasil = ' Seratus' + katakan(n-100)
    elif (n < 1000):
        Hasil = katakan(n / 100) + ' Ratus' + katakan(n % 100)
    elif (n < 2000):
        Hasil = ' Seribu' + katakan(n-1000)
    elif (n < 10000):
        Hasil = katakan(n / 1000) + ' Ribu' + katakan(n % 1000)
    elif (n < 20000):
        Hasil = ' Sepuluh Ribu' + katakan(n-10000)
    elif (n < 100000):
        Hasil = katakan(n / 10000) + ' Puluh' + katakan(n % 10000)
    elif (n < 200000):
        Hasil = ' Seratus' + katakan(n-100000)
    elif (n < 1000000):
        Hasil = katakan(n / 100000) + ' Ratus' + katakan(n % 100000)
    elif (n < 2000000):
        Hasil = ' Satu Juta' + katakan(n-1000000)
    elif (n < 10000000):
        Hasil = katakan(n / 1000000) + ' Juta' + katakan(n % 1000000)
    elif (n < 100000000):
        Hasil = ' Satu Milyar' + katakan(n % 100000000)
    else:
        Hasil = 'Angka hanya sampai satu milyar'
    return Hasil

=== RESTART: C:/Users/asus/AppData/Local/Programs/Python/Python38-32/Py
>>> katakan(12300000)
'Angka hanya sampai satu milyar'
>>> katakan(120)
'Seratus Dua Puluh '
>>>
```

14. membuat suatu fungsi formatRupiah() yang menerima suatu bilangan bulat positif dan mengembalikan suatu fungsi yang merupakan bilangan itu tapi dengan 'format rupiah'.

```
coba.py - C:/Users/asus/AppData/Local/Programs/Python/Python38-32/co
File Edit Format Run Options Window Help

def formatRupiah(bilangan):
    y = str(bilangan)
    if len(y) <= 3:
        return ('Rp ' + y)
    else:
        p = y[-3:]
        q = y[:-3]
        return formatRupiah(q) + '.' + p
    print ('Rp' + formatRupiah(q) + '.' + p)

=== RESTART: C:/Users/asus/AppData/Local/Programs/Py
>>> formatRupiah(1500)
'Rp 1.500'
>>> formatRupiah(2560000)
'Rp 2.560.000'
>>>
```