

Laporan Praktikum Algoritma dan Struktur Data

Modul 01

“Tinjauan Ulang Python”

Nama : Aiza Fravy Qanza

NIM : L200170144

Kelas : D

Soal-soal untuk Mahasiswa

1. Fungsi cetakSiku(x) yang akan mencetak seperti segitiga siku

```
>>> #Nomer 1
>>> def cetakSiku(x):
        for i in range (1, x+1):
            print ("***i)

>>> cetakSiku(5)
*
**
***
****
*****
```

2. Fungsi yang menerima dua integer positif, yang akan menggambar persegi empat

```
>>> #Nomer 2
>>> def gambarlahPersegiEmpat(a,b):
        i = 1
        print("@"*b)
        while (i<a):
            print ("@"+" "*(b-2)+"@")
            i+=1
        print ("@"*b)

>>> gambarlahPersegiEmpat (4,5)
@@@@@
@  @
@  @
@  @
@@@@@
```

3. (a) Fungsi yang menerima sebuah string dan mengembalikan sebuah list yang terdiri dari dua integer. Dua integer kembalian ini adalah jumlah huruf di string itu dan jumlah huruf vokal (a,i,u,e,o) di string itu

```
>>> #Nomer 3(a)
>>> def jumlahHurufVokal (s):
    vok = "aiueo"
    jumlah = 0
    for i in s :
        if i.lower() in vok:
            jumlah += 1
    return (len(s), jumlah)

>>> jumlahHurufVokal("surakarta")
(9, 4)
```

- (b) Sama dengan soal (a) diatas, tetapi sekarang yang dihitung adalah huruf konsonan

```
>>> #Nomer 3(b)
>>> def jumlahHurufKonsonan (s):
    vok = "aiueo"
    jumlah = 0
    for i in s :
        if i.lower() not in vok:
            jumlah += 1
    return (len(s), jumlah)

>>> jumlahHurufKonsonan("surakarta")
(9, 5)
```

4. Fungsi yang menghitung rata-rata sebuah array yang berisi bilangan

```
>>> #Nomer 4
>>> def rerata (b):
    jumlah = 0
    for i in b:
        jumlah += i
    return jumlah/len(b)

>>> print (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. Fungsi untuk menentukan apakah suatu bilangan bulat adalah bilangan prima / bukan

```
>>> #Nomer 5
>>> from math import sqrt as sq
>>> def apakahPrima(n):
    n=int(n)
    assert n>=0
    primakecil=[2, 3, 5, 7, 11]
    bukanprima=[0, 1, 4, 6, 8, 9, 10]
    if n in primakecil:
        return True
    elif n in bukanprima:
        return False
    else:
        for i in range(2,int(sq(n))+1):
            if(n%i==0):
                return False
        return True

>>> print(apakahPrima(17))
True
>>> print(apakahPrima(123))
False
```

6. Program yang mencetak semua bilangan prima dari 2 sampai 1000

```
>>> #Nomer 6
>>> def cetakbilanganprima():
    prima=list()
    for i in range(2,1000):
        a = True
        for iter in prima:
            if(i%iter==0):
                a=False
                break
        if(a):
            print(i)
            prima.append(i)

>>> cetakbilanganprima()
```

Hasilnya :

2	157	367	599	
3	163	373	601	
5	167	379	607	
7	173	383	613	
11	179	389	617	
13	181	397	619	
17	191	401	631	
19	193	409	641	
23	197	419	643	
29	199	421	647	
31	211	431	653	
37	223	433	659	
41	227	439	661	829
43	229	443	673	839
47	233	449	677	853
53	239	457	683	857
59	241	461	691	859
61	251	463	701	863
67	257	467	709	877
71	263	479	719	881
73	269	487	727	883
79	271	491	733	887
83	277	499	739	907
89	281	503	743	911
97	283	509	751	919
101	293	521	757	929
103	307	523	761	937
107	311	541	769	941
109	313	547	773	947
113	317	557	787	953
127	331	563	797	967
131	337	569	809	971
137	347	571	811	977
139	349	577	821	983
149	353	587	823	991
151	359	593	827	997

7. Program yang menerima bilangan bulat positif dan memberikan faktorisasi primanya

```
>>> #Nomer 7
>>> def faktorprima(n):
    prima=list()
    for i in range(2,n):
        a = True
        for iter in prima:
            if(i%iter==0):
                a=False
                break
        if a and n%i==0:
            prima.append(i)
    return prima

>>> faktorprima(10)
[2, 5]
```

8. Fungsi apakahTerkandung (a,b) yang menerima dua string a dan b, lalu menentukan apakah string a terkandung dalam string b

```
>>> #Nomer 8
>>> def apakahTerkandung(a,b):
    return a.lower() in b.lower()

>>> h = "do"
>>> k = " Indonesia tanah air beta "
>>> 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'. Kalau pas kelipatan 3 sekaligus 5, cetak 'Python UMS'

```
>>> #Nomer 9
>>> def iterasi():
    for i in range(1,100):
        if (i%3)!=0 and (i%5)!=0:
            print(i)
        else:
            if (i%15)==0:
                print("python UMS")
            elif (i%3)==0:
                print("python")
            elif (i%5)==0:
                print("UMS")
```

Hasilnya :

>>> iterasi() 1 2 python 4 UMS python 7 8 python UMS 11 python 13 14 pyton UMS 16 17 python 19 UMS python 22 23 python UMS	26 python 28 29 pyton UMS 31 32 python 34 UMS python 37 38 python UMS 41 python 43 44 pyton UMS 46 47 python 49 UMS	python 52 53 python UMS 56 python 58 59 pyton UMS 61 62 python 64 UMS python 67 68 python UMS 71 python 73 74 pyton UMS	76 77 python 79 UMS python 82 83 python UMS 86 python 88 89 pyton UMS 91 92 python 94 UMS python 97 98 python
---	---	---	--

10. Modifikasi pada contoh 1.4, agar bisa menangkap kasus dimana determinannya kurang dari nol

```
>>> #Nomer 10
>>> from math import sqrt as akar
>>> def selesaikanABC(a,b,c):
    a=float(a)
    b=float(b)
    c=float(c)
    D=(b**2)-(4*a*c)
    if D<0:
        return "Determinan negatif. Persamaan tidak mempunyai akar real"
    else:
        x1 = (-b+akar(D))/(2*a)
        x2 = (-b-akar(D))/(2*a)
        hasil = (x1,x2)
        return hasil

>>> selesaikanABC(1,2,3)
'Determinan negatif. Persamaan tidak mempunyai akar real'
>>> selesaikanABC(1,-5,6)
(3.0, 2.0)
```

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

```
>>> #Nomer 11
>>> def apakahKabisat(a):
    if(a%400==0):
        return True
    if(a%100==0):
        return False
    if(a%4==0):
        return True
    return False

>>> apakahKabisat(2000)
True
>>> apakahKabisat(1900)
False
```

12. Program permainan tebak angka. Buat program yang alurnya secara global seperti ini:
- Komputer membangkitkan bilangan bulat random antara 1 sampai 100. Nilainya disimpan disuatu variabel dan tidak ditampilkan ke pengguna
 - Pengguna diminta menebak angka itu, diinputkan lewat keyboard
 - Jika angka yang diinputkan terlalu kecil atau terlalu besar, pengguna mendapatkan umpan balik dari komputer (“Angka itu terlalu kecil. Coba lagi”)
 - Proses diulangi sampai angka itu tertebak atau sampai sekian tebakan meleset

```
>>> #Nomer 12
>>> import random
>>> def permainan():
    a=random.randrange(0, 100)
    while(True):
        b=int(input("Masukan tebakan: "))
        if(b>a):
            print("Itu terlalu besar, coba lagi")
        elif(b<a):
            print("Itu terlalu kecil, coba lagi")
        else:
            print("Ya, Anda benar")
            break
```

Hasilnya :

```
>>> permainan()
Masukan tebakan: 28
Itu terlalu besar, coba lagi
Masukan tebakan: 12
Itu terlalu kecil, coba lagi
Masukan tebakan: 16
Itu terlalu kecil, coba lagi
Masukan tebakan: 18
Itu terlalu besar, coba lagi
Masukan tebakan: 17
Ya, Anda benar
```

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

```
>>> #Nomer 13
>>> def katakan(a):
    x={"0":"","1":"Se", "2":"Dua ", "3":"Tiga ", "4":"Empat ", "5":"Lima ", "6":"Enam ",
      "7":"Tujuh ", "8":"Delapan ", "9":"Sembilan "}
    y={-1:"", -2:"Puluh ", -3:"Ratus ", -4:"Ribu ", -5:"Puluh ", 6:"Ratus ", 7:"Juta ", 8:"Puluhjuta "}
    b=str(a)
    f=""
    i=-1
    while i>= -len(b):
        f=x[b[i]]+y[i]+f
        i-=1
    return f

>>> katakan (1999)
'SeRibu Sembilan Ratus Sembilan Puluh Sembilan '
```

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

```
>>> #Nomer 14
>>> def formatRupiah(a):
    b=str(a)
    c=""
    i = -1
    while i>= -len(b):
        if ((i+1)%3==0 and (i+1)!=0):
            c="."+c
        c=b[i]+c
        i-=1
    return "Rp "+c

>>> formatRupiah(150000)
'Rp 150.000'
```