

Nama : Amartya Bintang Wijat Ranti  
NIM : L200180193  
Kelas : G  
Mata Kuliah : Praktikum Algoritma dan Struktur Data

## Modul 1 Tinjauan Ulang Python

1. Buatlah suatu fungsi cetakSiku(x)

```
Bintang.py - E:\Materi Kuliah\Semester 4\Praktikum Python\Modul 1\Tinjauan Ulang Python\Bintang.py
File Edit Format Run Options Window Help

#1
def cetakSiku(x):
    for i in range(1, x+1):
        print('*'*i)

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

2. Buatlah sebuah fungsi yang menerima dua integer positif

```
#2
def gambarlahPersegiEmpat(a,b):
    for i in range(a):
        if i==0 or i==a-1:
            print(b * '@')
        else:
            print('@' + ' ' * (b-2) + '@')

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

3. A. Buatlah sebuah fungsi yang menerima sebuah string dan mengembalikan sebuah list yang terdiri dari dua integer. Menghitung jumlah vokal.

```
#3a
def jumlahHurufVokal(huruf):
    vokal = 'aiueoAIUEO'
    a = 0
    hasil = 0
    for i in huruf:
        if i in vokal:
            a += len(i)
        else:
            a += 0
    hasil = len(huruf), a
    return hasil

>>> jumlahHurufVokal('Surakarta')
(9, 4)
>>> |
```

- B. Sama seperti soal (A), tapi sekarang yang dihitung adalah huruf konsonan.

```
#3b
def jumlahHurufKonsonan(huruf):
    konsonan = 'bcdfghjklmnpqrstvwxyzBCDFGHIJKLMNOPQRSTUVWXYZ'
    b = 0
    hasil = 0
    for i in huruf:
        if i in konsonan:
            b += len(i)
        else:
            b += 0
    hasil = len(huruf), b
    return hasil

>>> jumlahHurufKonsonan('Surakarta')
(9, 5)
>>> |
```

4. Buatlah sebuah fungsi yang menghitung rerata sebuah array yang berisi bilangan.

```
#4
def rerata(b):
    return sum(b)/len(b)

>>> rerata([1,2,3,4,5])
3.0
>>> rerata([3,4,5,4,3,4,5,2,2,10,11,23])
6.333333333333333
>>> |
```

5. Buatlah suatu fungsi untuk menentukan apakah suatu bilangan bulat adalah bilangan prima atau bukan.

```
#5
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
```

```
>>> apakahPrima(17)
True
>>> apakahPrima(97)
True
>>> apakahPrima(123)
False
>>>
```

6. Buatlah suatu program yang mencetak suatu bilangan prima dari 2 sampai 1000

```
#6
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)
```

```
>>> bilanganPrima(1000)
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97
101
103
107
109
113
127
131
137
139
149
151
157
163
167
173
179
181
191
193
337
467
769
199
211
223
227
229
233
239
241
251
257
263
269
271
277
281
283
293
307
311
313
317
331
347
479
619
773
349
487
631
787
353
491
641
797
359
499
643
809
367
503
647
811
373
509
653
821
379
521
659
823
383
523
661
827
389
541
673
829
397
547
677
839
401
557
683
853
409
563
691
857
419
569
701
859
421
571
709
863
431
577
719
877
433
587
727
881
439
593
733
883
443
599
739
887
449
601
743
907
457
607
751
911
461
613
757
919
463
617
761
929
937
941
947
953
967
971
977
983
991
997
>>>
```

7. Buatlah suatu program yang menerima bilangan bulat positif dan memberikan faktorisasi primanya.

```
#7
def faktorPrima(x):
    bilanganList = []
    loop = 2
    while loop <= x:
        if x%loop == 0:
            x/=loop
            bilanganList.append(loop)
        else:
            loop +=1
    return bilanganList

>>> faktorPrima(10)
[2, 5]
>>> faktorPrima(120)
[2, 2, 2, 3, 5]
>>> faktorPrima(19)
[19]
>>> |
```

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

```
#8
def apakahTerkandung(a,b):
    x = True
    for i in range(len(b)):
        if a in b:
            x = True
        else:
            x = False
    return x

>>> a = 'do'
>>> b = 'Indonesia tanah air beta'
>>> apakahTerkandung(a,b)
True
>>> apakahTerkandung('pusaka',b)
False
>>> |
```

9. Buat program untuk mencetak angka dari 1 sampai 100. kalau angkanyanpas kelipatan 3 cetak 'Python', kalau kelipatan 5 cetak 'UMS', kalau pas kelipatan 3 sekaligus 5 cetak 'Python UMS'

```
#9
def kelipatan(x):
    for i in range(x):
        if (i<=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(101)
1      22      44      Python
2      23      Python UMS 67
Python 26      Python 68
4      UMS      47      Python
UMS     28      Python  UMS
Python 29      Python  71
7      31      UMS     Python
8      32      Python  73
Python 34      Python UMS 74
UMS     37      53      Python UMS 88
11      38      Python  76      Python UMS 89
Python 41      UMS     77      91
13      43      Python  79      Python
14      UMS     58      UMS     94
Python UMS 39      Python  82      Python
16      40      59      Python  UMS
17      42      Python UMS 83      Python
Python 44      61      84      97
19      UMS     62      Python  98
UMS     46      Python  UMS     Python
Python 48      64      86      UMS
22      49      UMS     Python  >>> |
```

10. Buat modifikasi pada contoh 1.4 agar bisa menangkap kasus dimana determinannya kurang dari nol.

```
#10
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. Buat suatu fungsi apakahKabisat() yang menerima suatu angka (tahun). jika tahun itu kabisat, kembalikan True. Jika bukan kabisat, kembalikan False

```
#11
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
```



```

>>> apakahKabisat(1896)
True
>>> apakahKabisat(1897)
False
>>> apakahKabisat(1900)
False
>>> apakahKabisat(2000)
True
>>> apakahKabisat(2004)
True
>>> apakahKabisat(2008)
True
>>> apakahKabisat(2012)
True
>>> apakahKabisat(2016)
True
>>> apakahKabisat(2096)
True
>>> apakahKabisat(2100)
False
>>> apakahKabisat(2200)
False
>>> apakahKabisat(2300)
False
>>> apakahKabisat(2400)
True
>>> |

```

## 12. Program permainan tebak angka:

```

#12
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:> 50
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 2:> 75
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 3:> 31
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 4:> 39
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 5:> 49
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 6:> 80
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 7:> 76
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 8:> 66
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 9:> 59
Itu Terlalu Besar, Coba Lagi
Masukkan tebakan ke- 10:> 55
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 11:> 56
Itu Terlalu Kecil, Coba Lagi
Masukkan tebakan ke- 12:> 58
Ya, Anda Benar
>>> |
```

13. Buat suatu fungsi kataka() yang menerima bilangan bulat positif dan mengembalikan suatu string yang merupakan pengucapan angka itu dalam bahasa Indonesia.

```
#13
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 < 10000000):
    Hasil = ' Satu Milyar' + katakan(n % 10000000)
else:
    Hasil = 'Angka hanya sampai satu milyar'
return Hasil

```

```

>>> katakan(3125750)
' Tiga Juta Seratus Dua Puluh Lima Ribu Tujuh Ratus Lima Puluh '
>>> katakan(6758367)
' Enam Juta Tujuh Ratus Lima Puluh Delapan Ribu Tiga Ratus Enam Puluh Tujuh'
>>> katakan(3902812)
' Tiga Juta Sembilan Ratus Dua Ribu Delapan Ratus Dua Belas'
>>> katakan(3667489100)
'Angka hanya sampai satu milyar'
>>> katakan(14538)
' Sepuluh Ribu Empat Ribu Lima Ratus Tiga Puluh Delapan'
>>> |

```

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

```

#14
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)

```

```

>>> formatRupiah(1500)
'Rp 1.500'
>>> formatRupiah(2560000)
'Rp 2.560.000'
>>> |

```