



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	<b>71230997</b>
<b>Nama Lengkap</b>	<b>Christ Jevicto Ajimas Kirana</b>
<b>Minggu ke / Materi</b>	<b>04 / Modular Programming</b>

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
YOGYAKARTA  
2024

## BAGIAN 1: MATERI MINGGU INI (40%)

Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

### MATERI 1

#### 1.1 Fungsi, Argument dan Parameter

Fungsi adalah beberapa perintah yang dijadikan satu, yang memiliki suatu tujuan dan kegunaan tertentu dan dapat diulang.

Fungsi dibagi menjadi 2 jenis yaitu=

1. Fungsi bawaan (built-in-function)
2. Fungsi yang dibuat oleh programmer sendiri

```
def cek_angka(a,b,c):  
    if a!=b and b!=c and c!=a:  
        if a+b==c or b+c==a or c+a==b:  
            return True  
        return False
```

Contoh =

Penjelasan =

1. Keyboard **def** digunakan untuk mendefinisikan sebuah fungsi.
2. Nama fungsi diatas **cek\_angka()**
3. Isi dari fungsi harus dituliskan menjorok kedalam 1 tab. **cek\_angka(a,b,c):** sebagai penanda blok.
4. Fungsi **cek\_angka()** membutuhkan argument, yang akan dikenali sebagai parameter **a,b,dan c**
5. Fungsi itu akan menghasilkan output **True** atau **False**, keyword **return** digunakan untuk mengembalikan/mengeluarkan nilai dari suatu fungsi

#### 1.2 Return Value

Keyword **return** digunakan untuk =

1. Mengeluarkan nilai yang merupakan hasil dari fungsi
2. Mengakhiri fungsi.

```
def cek_angka(a,b,c):
    if a!=b and b!=c and c!=a:
        if a+b==c or b+c==a or c+a==b:
            return True
    return False
```

Contoh =

Penjelasan = ketika variable a,b,c diisi oleh data fungsi baru bekerja dengan cara jika a,b,c tidak sama lalu jika 2 variabel dijumlahkan hasilnya sama dengan variable lainnya (sis), maka outputnya akan True, namun jika salah satu dari 2 fungsi hasilnya False maka akan langsung menghasilkan False

### 1.3 Optional Argument dan Named Argument

Suatu fungsi memiliki 2 argumen yaitu =

#### 1. Optional Argument

```
def hitung_belanja(harga, diskon=0):
    bayar = harga - (harga*diskon)/100
    return bayar

print(hitung_belanja(100000))
print(hitung_belanja(100000, 50))
```

Contoh =

Penjelasan = Fungsi bisa memiliki nilai bawaan yang sudah didefinisikan sebelumnya yang bersifat opsional, ketika fungsi diberikan argument baru fungsi akan menggunakan argument kedua, namun jika fungsi tidak diberi argument maka fungsi akan menggunakan nilai bawaan yang sudah didefinisikan sebelumnya.

#### 2. Named Argument

<pre>def cetak(a,b,c):     print ("nilai a:", a)     print ("nilai b:", b)     print ("nilai c:", c)</pre>	<pre>def cetak(a,b,c):     print ("nilai a:", a)     print ("nilai b:", b)     print ("nilai c:", c)</pre>
<pre>cetak(20,30,40)</pre>	<pre>cetak(b=20,a=30,c=40)</pre>

Contoh =

Penjelasan = pemanggil fungsi cetak() dilakukan dengan menyebutkan nama argumennya. Jika kita tidak memberi variable apa yang akan diisi maka pemanggilan fungsi akan memberikan urutan urut dari kiri ke kanan parameter pada fungsi.

### 1.4 Anonymous Function (Lambda)

Anonymous function adalah fungsi tambahan bukan fungsi utama pada Python. Lambda digunakan untuk mendefinisikan Anonymous function. Setiap anonymous function pada Python memiliki beberapa bagian, seperti berikut ini:

- Keyword: lambda
- Bound variable: argument pada lambda function
- Body: bagian utama lambda, berisi ekspresi atau statement yang menghasilkan suatu nilai.

Contoh =

```
def tambah(a,b):  
    hasil = a + b  
    return hasil  
  
print(tambah(10,20))
```

->

```
tambah = lambda a, b: a + b  
print(tambah(10,20))
```

## MATERI 2

### 2.1 Mendefinisikan Fungsi

#### Contoh 2.1 Fungsi Menghitung Tagihan Listrik

```
def tagihan(pemakaian, golongan=3):  
    if golongan == 1:  
        harga_100_kwh_pertama = 1500  
        harga_selanjutnya = 2000  
    elif golongan == 2:  
        harga_100_kwh_pertama = 2500  
        harga_selanjutnya = 3000  
    elif golongan == 3:  
        harga_100_kwh_pertama = 4000  
        harga_selanjutnya = 5000  
    elif golongan == 4:  
        harga_100_kwh_pertama = 5000  
        harga_selanjutnya = 7000  
  
    if pemakaian <= 100:  
        harga = pemakaian * harga_100_kwh_pertama  
    elif pemakaian > 100:  
        harga = 100 * harga_100_kwh_pertama + (pemakaian-100)*harga_selanjutnya  
    return harga  
  
pemakaian = int(input("masukan kwh pemakaian ="))  
golongan= int(input("masukan golongan ="))  
  
harga = tagihan(pemakaian, golongan)  
print ("tagihan listrik anda adalah = Rp. ",harga)
```

Output =

```
masukan kwh pemakaian =80  
masukan golongan =4  
tagihan listrik anda adalah = Rp. 400000
```

## 2.2 Argument dan Parameter

### Contoh 4.2 Pengiriman Argument pada Fungsi

```
def abc(a,b,c):  
    a = b + c  
    b = a + c  
    c = a + b  
  
nilai_a = 1  
nilai_b = 2  
nilai_c = 3  
  
abc(nilai_a,nilai_b,nilai_c)  
  
print(nilai_a, nilai_b, nilai_c)  
✓ 0.1s  
1 2 3
```

**Penjelasan** = pengirim parameter di python menggunakan prinsip **Call-by-Object**. Perlakuan yang diberikan ke parameter tergantung dari apakah argument yang diberikan immutable atau tidak. Jika immutable maka argument tidak bisa diubah oleh fungsi seperti tipe data integer, string, dan tuple bersifat immutable. Nilai\_a, Nilai\_b, nilai\_c bersifat integer, oleh karena itu argument bersifat immutable, oleh karena itu tidak akan terpengaruh oleh perubahan nilai parameter dalam fungsi abc().

## 2.3 Anonymous/Lambda Function

### Contoh 4.3 Konversi ke Lambda Function

```
def kelipatan Sembilan(angka):  
    if angka % 9 == 0:  
        return True  
    else:  
        return False
```

Menjadi

```
kelipatan Sembilan = lambda angka: angka % 9 == 0
```

## BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

### SOAL 1

```
tabnine: test | explain | document | ask
1  def cek_angka(a,b,c):
2      if a!=b and b!=c and c!=a:
3          if a+b==c or b+c==a or c+a==b:
4              return True
5          return False
6      if a==b or b==c or c==a:
7          return False
8
9
10 a = int(input('masukan a= '))
11 b = int(input('masukan b= '))
12 c = int(input('masukan c= '))
13
14 print(cek_angka(a,b,c))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JU

```
PS D:\contoh python> & C:/Users/chris/AppData/Local/Programs/Python/Python38-64/python.exe "d:/contoh python/coba2.py"
masukan a= 5
masukan b= 10
masukan c= 5
False
PS D:\contoh python> & C:/Users/chris/AppData/Local/Programs/Python/Python38-64/python.exe "d:/contoh python/coba2.py"
masukan a= 1
masukan b= 2
masukan c= 3
True
PS D:\contoh python> & C:/Users/chris/AppData/Local/Programs/Python/Python38-64/python.exe "d:/contoh python/coba2.py"
masukan a= 1
masukan b= 2
masukan c= 5
False
```

penjelasan = kita diminta untuk menginputkan 3 angka yang berbeda dan jika 2 dari 3 dijumlahkan hasilnya adalah variabel yang tersisa dan hasilnya adalah true tetapi jika hasilnya tidak sama dan inputannya ada yang sama hasilnya akan False

## SOAL 2

```
1 def cek_digit_belakang(a,b,c):
2     a = a % 10
3     b = b % 10
4     c = c % 10
5
6     if a==b or b==c or c == a:
7         return True
8     else:
9         return False
10
11 a = int(input('masukan a ='))
12 b = int(input('masukan b ='))
13 c = int(input('masukan c ='))
14
15 print (cek_digit_belakang(a,b,c))
16
```

```
PS D:\contoh python> &
thon.exe "d:/contoh py
masukan a =30
masukan b =20
masukan c =18
True
PS D:\contoh python> &
thon.exe "d:/contoh py
masukan a =145
masukan b =5
masukan c =100
True
PS D:\contoh python> &
thon.exe "d:/contoh py
masukan a =71
masukan b =187
masukan c =18
False
PS D:\contoh python> &
thon.exe "d:/contoh py
masukan a =1024
masukan b =14
masukan c =94
True
PS D:\contoh python> &
thon.exe "d:/contoh py
masukan a =53
masukan b =8900
masukan c =658
False
PS D:\contoh python> █
```

penjelasan = kita

ditugaskan untuk mencari apakah digit belakang 3 angka yang user masukan sama atau tidak jika ada dua sampai 3 yang angka belakangnya sama maka hasilnya akan true dan jika berbeda semua hasilnya false, disini saya menggunakan modulo 10 untuk mencari angka belakang sama atau tidak



### SOAL 3

```
1 f = lambda c: (9/5)*c+32
2 r = lambda c: 0.8*c
3
4 tabnine: test | explain | document | ask
5 def pilih_konversi():
6     print("pilih konversi yang di inginkan")
7     print("1 = Farenheit")
8     print("2= Reamur")
9     pilihan = (input("masukan konversi yang ingin anda pilih (1/2) = "))
10    return int(pilihan)
11
12 c = int(input("masukan suhu celcius ="))
13
14 pilihan_konversi = pilih_konversi()
15
16 if pilihan_konversi == 1:
17     output = f(c)
18     print ("input Celcius = {}, output Farenheit = {}".format(c,output))
19 elif pilihan_konversi == 2:
20     output = r(c)
21     print ("input Celcius = {}, output Reamur = {}".format(c,output))
```

```
PS D:\contoh python> & C:/Users/chris/AppData/Local/Programs/Python/Python38-32/python.exe "d:/contoh python/contoh pyhton.py"
masukan suhu celcius =100
pilih konversi yang di inginkan
1 = Farenheit
2= Reamur
masukan konversi yang ingin anda pilih (1/2) = 1
input Celcius = 100. output Farenheit = 212.0.
PS D:\contoh python> & C:/Users/chris/AppData/Local/Programs/Python/Python38-32/python.exe "d:/contoh python/contoh pyhton.py"
masukan suhu celcius =80
pilih konversi yang di inginkan
1 = Farenheit
2= Reamur
masukan konversi yang ingin anda pilih (1/2) = 2
input Celcius = 80. output Reamur = 64.0.
PS D:\contoh python> & C:/Users/chris/AppData/Local/Programs/Python/Python38-32/python.exe "d:/contoh python/contoh pyhton.py"
masukan suhu celcius =0
pilih konversi yang di inginkan
1 = Farenheit
2= Reamur
masukan konversi yang ingin anda pilih (1/2) = 1
input Celcius = 0. output Farenheit = 32.0.
PS D:\contoh python> 
```

penjelasan = disini saya melakukan lambda untuk rumus

farenheit dan reamur lalu meminta user untuk memasukkan derajat celcius dan meminta user untuk memilih ingin mengkonversikan ke reamur atau farenheit.

**LINK GITHUB = <https://github.com/Yeeemeki/praktikum-algoritma-dan-pemograman.git>**