

**LAPORAN RESMI**  
**MODUL 4**  
**FUNCTION ( FUNGSI )**  
**ALGORITMA PEMROGRAMAN**



<b>NAMA</b>	<b>: IKTIAR RAMADANI</b>
<b>N.R.P</b>	<b>: 230441100053</b>
<b>DOSEN</b>	<b>: MOHAMMAD SYARIEF, ST., M.Cs</b>
<b>ASISTEN</b>	<b>: KUKUH COKRO WIBOWO</b>
<b>TGL PRAKTIKUM</b>	<b>: 13 OKTOBER 2023</b>

**Disetujui : 17 Oktober 2023**  
**Asisten**

**KUKUH COKRO WIBOWO**  
**21.04.411.00102**



**LABORATORIUM BISNIS INTELIJEN SISTEM**  
**PRODI SISTEM INFORMASI**  
**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS TRUNOJOYO MADURA**

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Maraknya penggunaan computer di berbagai bidang kehidupan manusia, menuntut setiap orang untuk mengetahui dan mempelajari berbagai macam software pendukung yang dapat beroperasi pada computer tersebut. Computer merupakan salah satu penemuan terancang pada abad ini. Computer merupakan cikal bakal lahirnya teknologi baru di bidang Ilmu Pengetahuan dan Teknologi.

Kemampuan computer dalam melakukan penghitungan yang sangat cepat, dapat mempermudah manusia atau user dalam mengoperasikannya. Berbagai macam media pendukung untuk mengoptimalkan kinerja computer banyak dibuat. Media tersebut berupa software atau program aplikasi yang hubungannya tidak dapat terpisahkan dari computer.

Software atau program aplikasi tersebut dibangun dengan menggunakan software lain. Banyak software yang digunakan untuk membangun software atau program aplikasi. Salah satunya bahasa pemrograman (programming language). Bahasa pemrograman pun memiliki beberapa tingkatan, disesuaikan dengan kemudahan akses yang dilakukan oleh computer. Dalam laporan kali ini, penyusun telah merangkum materi pengertian FUNGSI (Function)

### **1.2 Tujuan**

- Mampu memahami apa yang dimaksud dengan fungsi
- Mampu membuat program dengan menggunakan function
- Dapat membedakan apa itu def, lambda, ataupun fungsi dan jenis lainnya

## **BAB II**

### **DASAR TEORI**

#### **2.1 Pengertian, keuntungan, dan kategori function atau fungsi**

Pengertian fungsi atau function adalah suatu dari bagian program yang di pergunakan untuk mengerjakan suatu tugas tertentu yang menghasilkan suatu nilai untuk di kembalikan ke program pemanggil dan letaknya di pisahkan dari bagian program yang menggunakannya. Fungsi (function) merupakan serangkaian script/kode yang mempunyai kegunaan khusus dan tertentu: merupakan seurutan atau rangkaian kode yang sering dipakai. Dengan adanya fungsi ini pemrogram dapat dipermudah karena tidak harus menulis berulang-ulang rangkaian kode script. Demikian juga dalam pengembangan script, apabila terjadi kesalahan atau perbaikan pada fungsi tersebut, pemrogram hanya melakukan perbaikan pada satu tempat, tidak perlu melakukan perbaikan pada banyak script.

Fungsi (Function) adalah suatu program terpisah dalam blok sendiri yang berfungsi sebagai sub-program (modul program) yang merupakan sebuah program kecil untuk memproses sebagian dari pekerjaan program utama. Fungsi digunakan untuk mengumpulkan beberapa perintah yang sering dipakai dalam sebuah program. Fungsi juga bisa diartikan sebagai bagian dari program yang dapat digunakan kembali. Hal ini bisa dicapai dengan memberi nama pada blok statemen, kemudian nama ini dapat dipanggil di manapun dalam program. Kita telah menggunakan beberapa fungsi builtin seperti range. Fungsi dalam Python didefinisikan menggunakan kata kunci def. Setelah def ada nama pengenalan fungsi diikuti dengan parameter yang diapit oleh tanda kurung dan diakhiri dengan tanda titik dua

Keuntungan menggunakan fungsi yaitu Program besar dapat di pisah-pisah menjadi program-program kecil melalui function, memudahkan dalam mencari kesalahan-kesalahan karena alur logika jelas dan kesalahan dapat dilokalisasi dalam suatu modul tertentu, Memperbaiki atau memodifikasi program dapat dilakukan pada suatu modul tertentu saja tanpa mengganggu keseluruhan program, dan dapat digunakan kembali (Reusability) oleh program atau fungsi lain

Fungsi memiliki kategori berupa Standard Library Function yaitu fungsi-fungsi yang telah disediakan oleh Interpreter Python dalam file-file atau librarinya. Dan juga Programme-Defined Function function yang dibuat oleh programmer sendiri. Function ini memiliki nama tertentu yang unik dalam program, letaknya terpisah dari program utama, dan bisa dijadikan satu ke dalam suatu library buatan programmer itu sendiri

## **2.2 Mendeklarasikan dan memakai fungsi**

- a) Statemen def adalah perintah standar dalam python untuk mendefinisikan sebuah fungsi. Def dalam python merupakan perintah yang executable, artinya function tidak akan aktif sampai python merunning perintah def tersebut. Statemen def digunakan untuk mendeklarasikan fungsi. Sedangkan statemen return digunakan untuk mengembalikan suatu nilai kepada bagian program yang memanggil fungsi.
- b) Statemen return dapat diletakkan di bagian mana saja dalam tubuh fungsi. Statemen return menandakan akhir dari pemanggilan fungsi dan akan mengirimkan suatu nilai balik kepada program yang memanggil fungsi tersebut. Statemen return bersifat opsional, artinya jika sebuah fungsi tidak memiliki statemen return, maka sebuah fungsi tidak akan mengembalikan suatu nilai apapun.
- c) Selain statemen def, Python juga menyediakan suatu bentuk ekspresi yang menghasilkan objek fungsi. Karena kesamaannya dengan tools dalam bahasa Lisp, ini disebut lambda Seperti def, ekspresi ini menciptakan sebuah fungsi yang akan dipanggil nanti, tapi mengembalikan fungsi dan bukan untuk menetapkan nama. Lambda dalam python lebih dikenal dengan nama Anonymous Function (Fungsi yang tidak disebutkan bukanlah sebuah perintah (statemen) namun lebih namanya). Lambda kepada ekspresi (expression). Dalam prakteknya, mereka sering digunakan sebagai cara untuk inline definisi fungsi, atau untuk menunda pelaksanaan sepotong kode. Bentuk umum lambda adalah kata kunci lambda, diikuti oleh satu atau lebih argument (persis

seperti daftar argumen dalam tanda kurung di def header), diikuti oleh ekspresi setelah tanda titik dua.

- d) Perbedaan def dengan lambda adalah Lambda sebuah ekspresi, bukan pernyataan. Karena ini, sebuah lambda dapat muncul di tempat-tempat def tidak diperbolehkan oleh sintaks Python-di dalam daftar harfiah atau pemanggilan fungsi argumen, misalnya. Sebagai ekspresi, lambda mengembalikan nilai (fungsi baru) yang opsional dapat diberi nama. Sebaliknya, pernyataan def selalu memberikan fungsi baru ke nama di header, bukannya kembali sebagai hasilnya. Tubuh lambda adalah ekspresi tunggal, bukan satu blok statemen. Tubuh lambda sama dengan apa yang akan dimasukkan ke dalam statemen return dalam tubuh def.
- e) Fungsi Rekursif merupakan suatu fungsi yang memanggil dirinya sendiri. Artinya, fungsi tersebut dipanggil di dalam tubuh fungsi itu sendiri. Tujuan di lakukan rekursif adalah untuk menyederhanakan penulisan program dan menggantikan bentuk iterasi. Dengan rekursi, program akan lebih mudah dilihat. Mencari nilai faktorial dari suatu bilangan bulat positif adalah salah satu pokok bahasan yang memudahkan pemahaman mengenai fungsi rekursif. Scope variabel atau cakupan variabel merupakan suatu keadaan dimana pendeklarasian sebuah variabel di tentukan. Dalam scope variabel dikenal dua istilah yaitu local dan global. Variabel local ad ketika variabel tersebut didefinisikan didalam sebuah fungsi (def). Artinya, variabel tersebut hanya dapat di gunakan dalam cakupan fungsi tersebut saja. Variabel global didefinisikan diluar fungsi. Artinya, variabel tersebut dapat digunakan oleh fungsi lain atau pun program utamanya.

## BAB III

### TUGAS PENDAHULUAN

#### 3.1 Soal

1. Sebutkan dan jelaskan macam-macam fungsi?
2. Buatlah contoh soal dan program sederhana dengan menggunakan fungsi?
3. Jelaskan perbedaan antara def dan lambda? dengan menggunakan tabel
4. buatlah program dengan menggunakan fungsi len(), max(), min(), abs()!

#### 3.2 Jawaban

1. a. fungsi built-in: fungsi yang dapat digunakan tanpa perlu mendefinisikannya.  
b. fungsi pengguna: fungsi yang memungkinkan anda untuk mengelompokkan sejumlah pernyataan ke dalam satu blok.  
c. fungsi lambda: fungsi untuk menyederhanakan fungsi  
d. fungsi rekursif: fungsi yang memanggil dirinya sendiri  
e. fungsi generator: fungsi yang menghasilkan nilai secara bertahap.  
f. fungsi decorator: fungsi yang mengubah fungsi lain dengan menambahkan fungsionalitas tambahan  
g. fungsi callback: fungsi yang digunakan untuk menjadwalkan tindakan tertentu setelah tindakan lain  
h. fungsi anonim: fungsi yang tidak memiliki nama  
i. fungsi import: fungsi yang di definisikan dalam modul yang dapat di impor ke program

#### 2. def luas\_persegi\_panjang():

```
    c = panjang * lebar  
    return c
```

```
panjang = int(input("masukkan panjang: "))
```

```
lebar = int(input("masukkan lebar: "))
```

```
print("luas persegi panjang adalah ", luas_persegi_panjang())
```

3.

perbedaan	def	lambda
nama	namanya jelas	namanya tidak jelas
kompleksitas	dapat berisi sejumlah pernyataan yang kompleks	digunakan untuk fungsi sederhana dalam satu ekspresi
pengembalian nilai	menggunakan return	mengembalikan nilai dari ekspresi tunggal
contoh	<pre>def contoh():     c = a + b     return c</pre>	<pre>lambda a,b: a+b</pre>

4. angka = [1, 2, 3, 4, 5]

# menggunakan len()

```
panjang_list = len(angka)
print("panjang list:", panjang_list)
output:
panjang list: 5
```

# menggunakan max()

```
nilai_maksimum = max(angka)
print("nilai maksimum:", nilai_maksimum)
output:
nilai maksimum: 5
```

# menggunakan min()

```
nilai_minimum = min(angka)
print("nilai minimum:", nilai_minimum)
output:
nilai minimum: 1
```

# menggunakan abs()

```
nilai_absolut = abs(-15)
print("nilai absolut dari -15 adalah", nilai_absolut)
output:
nilai absolut dari -15 adalah 15
```

## BAB IV

### IMPLEMENTASI

#### 4.1 Membuat Program menghitung luas permukaan suatu bangun ruang

##### 4.1.1 Source Code

Buatlah fungsi (functions) def menghitung luas permukaan bangun ruang: kerucut, limas segilima, prisma segilima secara dinamis.

```
def luas_kerucut():
    print("")
    print("==== Hitung Luas permukaan Kerucut =====")
    r = float(input("Masukkan jari-jari : "))
    s = float(input("Masukkan garis pelukis: "))

    if r < 1 or s < 1:
        print("Angka harus lebih dari 1 atau tidak boleh negatif")
    elif r % 7 == 0:
        luaskerucut = 22/7 * r * (s + r)
        print("Maka luas kerucut adalah ", luaskerucut )
    else:
        luaskerucut = 3.14 * r * (s + r)
        print("Maka luas kerucut adalah ", luaskerucut)

def luas_limassegilima():
    print("")
    print("==== Hitung luas permukaan limas segilima =====")
    s = float(input("Masukkan sisi :"))
    sisitegak = float(input("Masukkan sisi tegak:"))

    if s < 1 or sisitegak < 1:
        print("Angka harus lebih dari 1 atau tidak boleh negatif")
    else:
        luaslimassegilima = (1.72 * s * s) + (5 * sisitegak)
        print("Maka luas permukaan limas segilima adalah ",
luaslimassegilima)

def luas_prismassegilima():
    print("")
    print("==== Hitung luas permukaan prisma segilima =====")
    la = float(input("Masukkan luas alas : "))
    ka = float(input("Masukkan keliling alas: "))
    t = float(input("Masukkan tinggi prisma: "))

    if la < 1 or ka < 1 or t < 1:
        print("Angka harus lebih dari 1 atau tidak boleh negatif")
```



```

    else:
        luasprismasegilima = 2 * la + (ka + t)
        print("Maka luas permukaan prisma segilima adalah ",
luasprismasegilima)

while True:
    print("Anda ingin menghitung apa???\n\n1. Hitung luas
kerucut\n2. Hitung luas limas segilima\n3. Hitung prisma
segilima\n0. Keluar\n ")
    user = int(input("Pilih rumus yang akan dicari: "))
    if (user == 1):
        luas_kerucut()
    elif (user == 2):
        luas_limasegilima()
    elif (user == 3):
        luas_prismasegilima()
    elif (user == 0):
        print(" TERIMA KASIH")
        break
    else:
        print("Pilihan tidak valid")

```

#### 4.1.2 Penjelasan

User memilih ingin menghitung luas permukaan yang mana. Ada 3 yang dapat dicari yaitu kerucut, limas segilima, prisma segilima. Setelah dipilih program akan meminta user untuk memasukkan datanya. Contoh apabila user memilih untuk menghitung luas permukaan kerucut maka user harus memasukkan data berupa jari-jari dan garis pelukisnya. Setelah itu akan muncul hasil beserta loop apakah user ingin menghitung lagi. Pada program di atas setiap rumus dimasukkan ke dalam fungsi tersendiri. Fungsi tersebut memiliki namanya tersendiri sehingga rumus nya juga tidak akan salah. Ada 3 rumus yang nantinya program akan menghitung sesuai dengan apa yang dicari oleh user. Apabila user memasukkan nilai yang salah maka program akan menampilkan output bahwa nilai tersebut salah, namun jika benar maka program akan lanjut. Setelah menampilkan hasilnya, program akan meminta kita apakah ingin melanjutkan mencari pertanyaan lain atau tidak. Jika user memilih “y” maka program akan mengulang lagi. Namun, apabila user memilih “t” maka program akan berhenti.

#### 4.1.3 Hasil

```
Anda ingin menghitung apa???

1. Hitung luas kerucut
2. Hitung luas limas segilima
3. Hitung prisma segilima
0. Keluar

Pilih rumus yang akan dicari: 1

===== Hitung Luas permukaan Kerucut =====
Masukkan jari-jari      : 7
Masukkan garis pelukis: 15
Maka luas kerucut adalah 484.0
```

#### 4.2. Membuat Program mencari nilai faktorial

##### 4.2.1. Source Code

Buatlah program untuk mencari nilai faktorial dari suatu bilangan bulat positif dengan menggunakan Fungsi Rekursif.

```
def faktorial(user):
    if user == 0:
        return 1
    else:
        return user * faktorial(user - 1)

while True:
    user = int(input("Masukkan bilangan bulat positif: "))

    if user < 0:
        print("Maaf, faktorial hanya untuk bilangan bulat positif dan tidak boleh negatif")
    else:
        hasil = faktorial(user)
        print("Faktorial dari", user, "adalah ", hasil)

    ulangi = input("Apakah ingin menghitung lagi? (y/t): ")
    if ulangi.lower() != "y":
        break
```

##### 4.2.2. Penjelasan

User diminta untuk memasukkan nilai bilangan bulat positif. Setelah memasukkan program akan menghitung bilangan tersebut dengan rumus faktorial yang sudah disiapkan. Setelah menghitung, program akan

menampilkan hasilnya. Namun, apabila nilai yang dimasukkan salah, program akan menampilkan bahwa nilai tersebut salah dan langsung berhenti.

#### 4.2.3 Hasil

```
Masukkan bilangan bulat positif: 5
Faktorial dari 5 adalah 120
Apakah ingin menghitung lagi? (y/t):
```

### 4.3 Membuat Program menghitung Bilangan Berpangkat dua

#### 4.3.1. Source Code

Membuat program untuk menghitung Bilangan Berpangkat dua dengan menggunakan Fungsi Rekursif.

```
def pangkat_dua(bilangan):
    if bilangan == 0:
        return 0
    elif bilangan == 1:
        return 1
    else:
        return bilangan * bilangan

while True:

    bilangan = int(input("Masukkan bilangan: "))

    hasil = pangkat_dua(bilangan)
    print(bilangan, " pangkat dua adalah ", hasil)

    ulangi = input("Apakah ingin menghitung lagi? (y/t): ")
    if ulangi.lower() != "y":
        break
```

#### 4.3.2. Penjelasan

Pada program diatas user diminta untuk memasukkan nilai yang akan dicari nilai pangkat duanya. Setelah itu program akan menghitung nilai tersebut dengan mengalikan bilangan tersebut sebanyak dua kali sehingga akan menampilkan hasil dari perkalian bilangan tersebut. Setelah itu program akan meminta user apakah ingin lanjut atau tidak.

#### 4.3.3. Hasil

```
Masukkan bilangan: 5  
5 pangkat dua adalah 25  
Apakah ingin menghitung lagi? (y/t):
```

## **BAB V**

### **PENUTUP**

#### **5.1 Analisa**

Dari hasil praktikum, praktikan menganalisa bahwa Fungsi (Function) merupakan suatu program terpisah dalam blok sendiri yang berfungsi sebagai sub-program (modul program) yang merupakan sebuah program kecil untuk memproses sebagian dari pekerjaan program utama. Artinya fungsi digunakan untuk mempersingkat script yang fungsinya sama agar tidak ada penulisan script yang sama dalam satu program. Di sini juga kita bisa mempelajari apa yang dimaksud dengan function, jenis-jenis, kegunaan dan lainnya. Mengetahui apa perbedaan dari def dan lambda, serta bisa membuat program sederhana dengan menggunakan function.

#### **5.2 Kesimpulan**

1. Kita jadi bisa mengerti apa yang dimaksud dengan fungsi
2. Fungsi (Function) merupakan suatu program terpisah dalam blok sendiri yang berfungsi sebagai sub-program (modul program) yang merupakan sebuah program kecil untuk memproses sebagian dari pekerjaan program utama.
3. Fungsi digunakan untuk mengumpulkan beberapa perintah yang sering dipakai dalam sebuah program
4. Dalam python terdapat dua perintah yang dapat digunakan untuk membuat sebuah fungsi, yaitu `statement def` dan `lambda`. Dan juga terdapat 2 macam fungsi yaitu fungsi rekursif dan scope variabel.
5. untuk menyelesaikannya karena untuk membuat script program yang sama tidak perlu membuatnya lagi melainkan kita hanya membuat fungsinya saja dan kita bisa memanggilnya kapan saja ketika kita membutuhkannya