

<p>Nama: Steven Raja P.M</p> <p>NIM: 064102400018</p>	 <p>Praktikum Algoritma &amp; Pemrograman</p>	<h2>MODUL 2</h2> <p>Nama Dosen: Anung B. Ariwibowo, M. Kom</p> <p>Nama Asisten Laboratorium:</p> <ol style="list-style-type: none"><li>Yustianas Rombon - 064002300015</li><li>Vira Aditya Kurniawan - 065002300012</li></ol>
<p>Hari/Tanggal: Jumat, 27 09 2024</p>		

## Operasi Aritmatika pada Python

### 1. Teori Singkat

#### Operasi Aritmatika

Operasi Aritmatika adalah bagian dari pengolahan bilangan dari sebuah komputer untuk melakukan operasi hitung. Selain melakukan operasi hitung, operasi aritmatika juga bisa dilakukan untuk operasi logika. Dasar melakukan operasi hitung dalam aritmatika komputer adalah penjumlahan atau yang disebut adder

Berikut ini adalah Operator Aritmatika pada Bahasa Pemrograman Python

Operator	Simbol
Penjumlahan	+
Pengurangan	-
Perkalian	*
Pembagian	/
Sisa Bagi	%
Pemangkatan	**



Berikut ini adalah contoh sederhana program operasi aritmatika:

Anaconda Prompt



```
(base) C:\Users\Azhar Rizky Zulma>python
Python 3.7.3 (default, Apr 24 2019, 15:29:51) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license"
for more information.

>>> 3 + 2
5
>>> 18 % 5
3
>>> abs(-7)
7
>>> float(9)
9.0
>>> int(5.3)
5
>>> complex(1, 2)
(1+2j)
>>> 2 ** 8
256
```

Operator matematik berfungsi dengan normal di Python seperti dalam bahasa pemrograman yang lain. Ada beberapa catatan yang harus diperhatikan.

- Operasi pengisian beberapa variabel dengan nilai yang sama dapat dilakukan sekali jalan.
- Tanda () dipakai untuk mengelompokkan operasi yang harus dilakukan terlebih dahulu.



- Pembagian bilangan integer dengan bilangan integer akan dibulatkan ke bawah.
- Bilangan integer akan dikonversikan menjadi bilangan floating point dalam operasi yang melibatkan bilangan integer dan bilangan floating point.
- Kita tidak dapat mengkonversikan bilangan kompleks ke bilangan real (floating point atau integer); hanya bilangan mutlaknya yang bisa kita dapatkan.

## Operator Penugasan

Seperti namanya, operator ini digunakan untuk memberikan tugas pada variabel. Misalnya:

umur = 18

Maka variabel umur telah kita berikan tugas untuk menyimpan angka 18. Selain menyimpan atau pengisian nilai, ada juga menjumlahkan, mengurangi, perkalian, pembagian, dsb. Selengkapnya bisa dilihat di tabel berikut.

Operator	Simbol
Pengisian	=
Penjumlahan	+=
Pengurangan	-=
Perkalian	*=
Pembagian	/=
Sisa Bagi	%=
Pemangkatan	**=

Karena pada python tidak terdapat operator increment & decrement maka operator tersebut digantikan oleh operator penugasan dengan menginputkan seperti ini  $i += l$ .

## Python Package

Package merupakan sebuah bundle atau pengelompokan dari banyak fungsi serta kelas (source code) menjadi satu kesatuan unit tunggal dalam library yang dapat digunakan serta dipanggil pada source code yang sedang anda kembangkan untuk mendapatkan sebuah fungsi tanpa harus mengetikkan source codenya secara berulang. Python sendiri menyediakan package untuk menyediakan operasi-operasi standar. Untuk operasi-operasi yang lebih khusus, perlu menggunakan fungsi dari package lain. Dalam praktikum kali ini kita akan belajar operasi



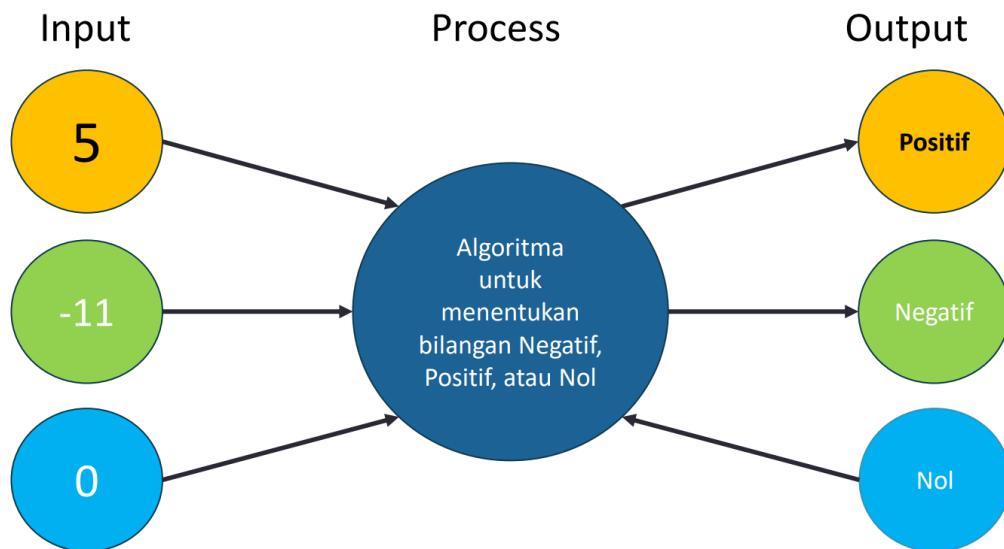
aritmatika dan belajar tentang menggunakan package untuk memanggil fungsi operasi trigonometri yang packagenya sudah disediakan pada Python yaitu dalam package *Math*:

## IPO (Input Process Output)

Konsep Dasar Input, Process, dan Output (IPO)

- Konsep input, process, dan output adalah prinsip dasar dalam pemrograman dan pengembangan algoritma.
- Setiap algoritma melibatkan tiga tahap utama: mengambil data masukan (input), melakukan operasi atau pengolahan data (process), dan menghasilkan hasil akhir (output).
- Konsep ini menggambarkan bagaimana algoritma beroperasi untuk memproses informasi.

## Gambaran IPO (Menentukan Bilangan)



## Pseudocode

Pseudocode adalah suatu bentuk deskripsi informal yang mirip dengan bahasa manusia dan digunakan untuk menggambarkan algoritma atau proses secara naratif. Ini tidak terikat pada



bahasa pemrograman tertentu, tetapi memberikan panduan tentang langkah-langkah yang harus diambil dalam suatu algoritma dengan bahasa yang lebih mudah dimengerti.

## Contoh PseudoCode

**Inisiasi Variabel:**

```
N      = 0
total = 0.0
```

**Pengulangan:**

```
UNTUK i DARI 1 SAMPAI 10 LANGKAH 2
    CETAK i
END UNTUK
```

**Pengkondisional (Conditional):**

```
JIKA nilai > 10
    CETAK "Nilai lebih dari 10"
SELAINNYA JIKA nilai = 10
    CETAK "Nilai sama dengan 10"
SELAINNYA
    CETAK "Nilai kurang dari 10"
AKHIR JIKA
```

**Fungsi atau Prosedur:**

```
FUNGSI tambah(a, b)
    KEMBALIKAN a + b
AKHIR FUNGSI
```

**Contoh Lengkap:**

```
DEKLARASI variabel n, bilangan, total, rata_rata FLOAT
MINTA "Masukkan jumlah bilangan: " SIMPAN
total = 0.0
```

```
UNTUK i DARI 1 SAMPAI n
    MINTA "Masukkan bilangan ke- " + i + ": "
    SIMPAN bilangan
    total = total + bilangan
END UNTUK

rata_rata = total / n
CETAK "Rata-rata adalah: " + rata_rata
```

## 2. Alat dan Bahan

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)

## 3. Elemen Kompetensi

### a. Latihan pertama

Buatlah sebuah program yang menerima dua bilangan bulat a dan b dari user. Program anda harus menampilkan hasil perhitungan:

- Jumlah a dan b
- Selisih antara b dengan a
- Hasil kali a dan b
- Sisa pembagian a dengan b
- Pembagian a dengan b
- Hasil dari  $\log(a)$
- $a$  pangkat b

Gunakan fungsi  $\log10()$  dari package math

Input Process Output (IPO) :



```
import math

# Menerima input dari pengguna

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

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

# Melakukan perhitungan

jumlah = a + b

selisih = b - a

kali = a * b

sisa_pembagian = a % b

pembagian = a / b if b != 0 else "Tak terdefinisi (b tidak boleh 0)"

log_a = math.log(a) if a > 0 else "Tak terdefinisi (a harus positif)"

pangkat = a ** b
```

Pseudocode :



```
#Menerima input dari pengguna
bilangan bulat a=
bilangan bulat b=
#melakukan perhitungan
jumlah=a+b
selisih=b-a
kali=a*b
sisa_pembagian=a%b
pembagian=a/b
log_a=math.log(a)
pangkat= a**b
print(f"Jumlah a dan b: {jumlah}")
print(f"Selisih b dengan a: {selisih}")
print(f"Hasil kali a dan b: {kali}")
print(f"Sisa pembagian a dengan b: {sisa_pembagian}")
print(f"Pembagian a dengan b: {pembagian}")
print(f"Hasil log(a): {log_a}")
print(f"Hasil a pangkat b: {pangkat}")
```

### Source Code



```
import math

# Menerima input dari pengguna
a = int(input("Masukkan bilangan bulat a: "))
b = int(input("Masukkan bilangan bulat b: "))

# Melakukan perhitungan
jumlah = a + b
selisih = b - a
kali = a * b
sisa_pembagian = a % b
pembagian = a / b if b != 0 else "Tak terdefinisi (b tidak boleh 0)"
log_a = math.log(a) if a > 0 else "Tak terdefinisi (a harus positif)"
pangkat = a ** b

# Menampilkan hasil perhitungan
print(f"Jumlah a dan b: {jumlah}")
print(f"Selisih b dengan a: {selisih}")
print(f"Hasil kali a dan b: {kali}")
print(f"Sisa pembagian a dengan b: {sisa_pembagian}")
print(f"Pembagian a dengan b: {pembagian}")
print(f"Hasil log(a): {log_a}")
print(f"Hasil a pangkat b: {pangkat}")
```



## Output

The screenshot shows a Google Colab notebook titled "Untitled2.ipynb - Colab". The code cell contains the following Python script:

```
# Melakukan perhitungan
jumlah = a + b
selisih = b - a
kali = a * b
sisa_pembagian = a % b
pembagian = a / b if b != 0 else "Tak terdefinisi (b tidak boleh 0)"
log_a = math.log(a) if a > 0 else "Tak terdefinisi (a harus positif)"
pangkat = a ** b

# Menampilkan hasil perhitungan
print(f"Jumlah a dan b: {jumlah}")
print(f"Selisih b dengan a: {selisih}")
print(f"Hasil kali a dan b: {kali}")
print(f"Sisa pembagian a dengan b: {sisa_pembagian}")
print(f"Pembagian a dengan b: {pembagian}")
print(f"Hasil log(a): {log_a}")
print(f"Hasil pangkat b: {pangkat}")

Masukkan bilangan bulat a: 10
Masukkan bilangan bulat b: 5
Jumlah a dan b: 15
Selisih b dengan a: -5
Hasil kali a dan b: 50
Sisa pembagian a dengan b: 0
Pembagian a dengan b: 2.0
Hasil log(a): 2.302585092994046
Hasil a pangkat b: 100000
```

The output cell shows the results of the calculations for inputs a=10 and b=5. The results are:

- Jumlah a dan b: 15
- Selisih b dengan a: -5
- Hasil kali a dan b: 50
- Sisa pembagian a dengan b: 0
- Pembagian a dengan b: 2.0
- Hasil log(a): 2.302585092994046
- Hasil a pangkat b: 100000

b. Latihan Kedua

Buatlah program untuk menghitung jarak antara dua titik di permukaan bumi menggunakan rumus dan fungsi trigonometri pada package math pada python.

### Source Code



```
import math

def haversine(lon1, lat1, lon2, lat2):
    """
    Calculate the great circle distance between two points
    on the earth (specified in decimal degrees)
    """
    # Convert decimal degrees to radians
    lon1, lat1, lon2, lat2 = map(math.radians, [lon1, lat1, lon2,
                                                lat2])

    # Haversine formula
    dlon = lon2 - lon1
    dlat = lat2 - lat1
    a = math.sin(dlat/2)**2 + math.cos(lat1) * math.cos(lat2) *
math.sin(dlon/2)**2
    c = 2 * math.asin(math.sqrt(a))
    r = 6371 # Radius of Earth in kilometers. Use 3956 for miles
    return c * r

# Contoh penggunaan:
# Koordinat titik A
lat1 = -6.2000
lon1 = 106.8167

# Koordinat titik B
lat2 = -6.1750
lon2 = 106.8244

distance = haversine(lon1, lat1, lon2, lat2)

print("Jarak antara kedua titik adalah", distance, "kilometer")
```

### Output



```
distance = haversine(lon1, lat1, lon2, lat2)

print("Jarak antara kedua titik adalah", distance, "kilometer")

Jarak antara kedua titik adalah 2.9072768383489245 kilometer
```

#### 4. File Praktikum

Github Repository:

<https://github.com/Stevenraja/Prak-Algo-2.git>

#### 5. Soal Latihan

Soal:

1. Dalam operasi aritmatika, presedensi yang tertinggi adalah ekspresi dalam tanda kurung. Jelaskan serta deskripsikan apa yang dimaksud dari presedensi yang tertinggi tersebut?
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. (Isi Dengan Jawaban Kalian)
2. (Isi Dengan Jawaban Kalian)

#### 6. Kesimpulan

- a. Dalam penggerjaan program dengan bahasa pemrograman Python, kita harus benar-benar teliti dalam menginputkan suatu fungsi untuk menampilkan suatu keluaran pada layar dengan sesuai.
- b. Kita dapat mengetahui... (Tolong Isi lebih dari dua baris!)



**7. Cek List (✓)**

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	✓	
2.	Latihan Kedua	✓	

**8. Formulir Umpam Balik**

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	15 Menit	Menarik
2.	Latihan Kedua	20 Menit	Menarik

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang

