


|  |   |   |
|--|---|---|
| <p><b>Nama:</b><br/>Aini Rihhadatul Aisy</p> <p><b>NIM:</b><br/>064102400024</p> |  <p><b>Praktikum Algoritma &amp; Pemrograman</b></p> | <p><b>MODUL 2</b></p> <p><b>Nama Dosen:</b><br/>Binti solihah, S.T,<br/>M.KOM</p>   |
| <p><b>Hari/Tanggal:</b><br/>Jumat, 27 September 2024</p>                         |   | <p><b>Nama Asisten Labratorium:</b></p> <ol style="list-style-type: none"> <li>1. Yustianas Rombon -<br/>064002300015</li> <li>2. Vira Aditya<br/>Kurniawan -<br/>065002300012</li> </ol> |

## Operasi Aritmatika pada Python

### 1. Teori Singkat

#### Operasi Aritmatika

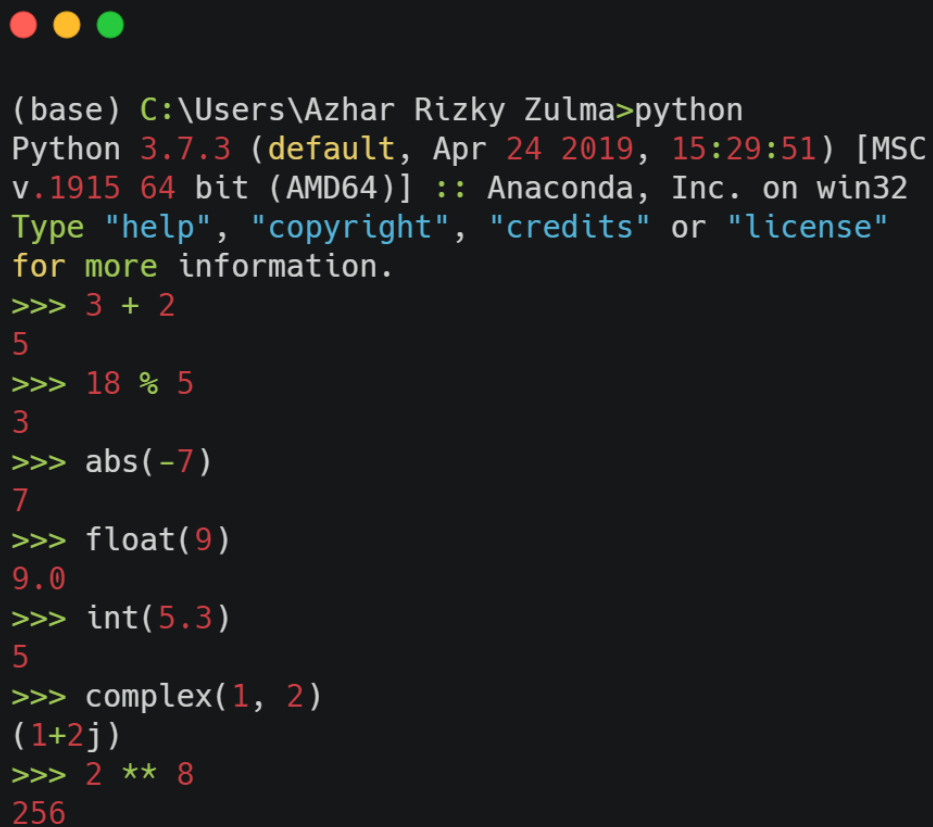
Operasi Aritmatika adalah bagian dari pengolahan bilangan dari sebuah komputer untuk melakukan operasi hitung. Selain melakukan operasi hitung, operasi aritmatika juga bisa dilakukan untuk operasi logika. Dasar melakukan operasi hitung dalam aritmatika komputer adalah penjumlahan atau yang disebut adder

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

A screenshot of the Anaconda Prompt terminal window. The window has a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The terminal shows the following text:

```
(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 mutlak 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   | <code>=</code>   |
| Penjumlahan | <code>+=</code>  |
| Pengurangan | <code>-=</code>  |
| Perkalian   | <code>*=</code>  |
| Pembagian   | <code>/=</code>  |
| Sisa Bagi   | <code>%=</code>  |
| Pemangkatan | <code>**=</code> |

Karena pada python tidak terdapat operator increment & decrement maka operator tersebut digantikan oleh operator penugasan dengan menginputkan seperti ini `i += 1`.

## 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



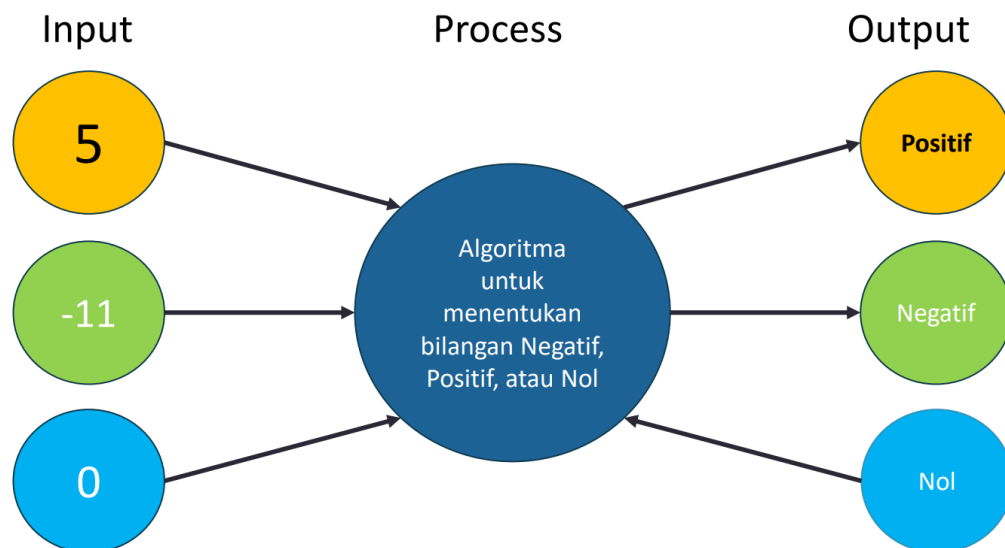
menggunakan fungsi dari package lain. Dalam praktikum kali ini kita akan belajar operasi aritmatika dan belajar tentang menggunakan package untuk memanggil fungsi operasi trigonometri yang packagenya sudah disediakan pada Python yaitu dalam package *Math*:

## 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) :



**Input:**

1. a: Bilangan bulat yang dimasukkan oleh pengguna (input dari pengguna).
2. b: Bilangan bulat yang dimasukkan oleh pengguna (input dari pengguna).

**Proses:**

1. Hitung Penjumlahan sebagai hasil dari  $a+b$ .
2. Hitung Pengurangan sebagai hasil dari  $a-b$ .
3. Hitung Perkalian sebagai hasil dari  $a \times b$ .
4. Hitung Pembagian sebagai hasil dari  $a/b$ .
5. Hitung sisa\_bagi sebagai hasil dari  $a \text{ mod } \% b$
6. Hitung perpangkatan sebagai hasil dari  $a^{**}b$ .
7. Hitung loga sebagai logarithm basis 10 dari a menggunakan  $\text{math.log10}(a)$ .

**Output:**

1. Tampilkan hasil Penjumlahan.
2. Tampilkan hasil Pengurangan.
3. Tampilkan hasil Perkalian.
4. Tampilkan hasil Pembagian.
5. Tampilkan hasil sisa\_bagi.
6. Tampilkan hasil loga.
7. Tampilkan hasil perpangkatan.

Pseudocode :



```
Masukkan import math
Masukkan Nilai Variabel a
Nilai Variabel a akan tersimpan di a
Masukkan Nilai Variabel b
Nilai Variabel a akan tersimpan di b
Hitung Nilai Penjumlahan dari (Variabel a) + (Variabel b)
Hitung Nilai Pengurangan dari (Variabel a) - (Variabel b)
Hitung Nilai Perkalian dari (Variabel a) * (Variabel b)
Hitung Nilai Pembagian dari (Variabel a) / (Variabel b)
Hitung Nilai sisa_bagi sari (Variabel a) % (Variabel b)
Hitung Nilai Perpangkatan dari (Variabel a) ** (Variabel b)
Hitung Nilai Log a dari Variabel a dengan rumus (math.log10(a))
Tampilkan Hasil Penjumlahan
Tampilkan Hasil Pengurangan
Tampilkan Hasil Perkalian
Tampilkan Hasil Pembagian
Tampilkan Hasil sisa_bagi
Tampilkan Hasil Loga
Tampilkan Hasil Pepangkatan
```

### Source Code



```
✓ 7s #Operasi Aritmatika
import math
a = int(input("Masukkan Variabel a : "))
b = int(input("Masukkan Variabel b : "))

Penjumlahan = (a + b)
Pengurangan = (a - b)
Perkalian = (a * b)
Pembagian = (a / b)
sisabagi = (a % b)
perpangkatan = (a ** b)
loga = math.log10(a)

print("Hasil Penjumlahan: " , Penjumlahan)
print("Hasil Pengurangan : " , Pengurangan)
print("Hasil Perkalian : " , Perkalian)
print("Hasil Pembagian : " , Pembagian)
print("Hasil Sisa Bagi : " , sisabagi)
print("Hasil Loga : " , loga)
print("Hasil Perpangkatan : " , perpangkatan)
```

### Output





```

➡ Masukkan Variabel a : 20
Masukkan Variabel b : 10
Hasil Penjumlahan: 30
Hasil Pengurangan : 10
Hasil Perkalian : 200
Hasil Pembagian : 2.0
Hasil Sisa Bagi : 0
Hasil Loga : 1.3010299956639813
Hasil Perpangkatan : 10240000000000

```

#### 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

Lintang1 = math.radians(float(input("Lintang Kota 1: ")))
Bujur1 = math.radians(float(input("Bujur Kota 1: ")))
Lintang2 = math.radians(float(input("Lintang kota 2: ")))
Bujur2 = math.radians(float(input("Bujur kota 2: ")))

R = 6371
Lat = Lintang2 - Lintang1
Long = Bujur2 - Bujur1

a = math.sin(Lat/2)**2 + math.cos(Lintang1) * math.cos(Lintang2) * math.sin(Long/2)**2
C = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))
d = R * C
print("jarak antara dua titik tersebut adalah ",d,"kilometer")

```



### Output

```
➡ Lintang Kota 1: 20  
  Bujur Kota 1: 30  
  Lintang kota 2: 10  
  Bujur kota 2: 4  
  jarak antara dua titik tersebut adalah 3000.247748329652 kilometer
```

#### 4. File Praktikum

Github Repository:

```
print("Taruh link github repository praktikum kalian disini yaa...")
```

#### 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. **Presedensi** adalah aturan yang menentukan urutan eksekusi operasi. Tanda kurung memiliki presedensi tertinggi, sehingga operasi di dalamnya harus diselesaikan terlebih dahulu. Presedensi yang Tertinggi Menggunakan tanda kurung guna untuk membantu memperjelas dan mengontrol hasil perhitungan dalam ekspresi yang kompleks.
2. Definisikan R sebagai jari-jari Bumi (6371 km)  
Tampilkan "Lintang Kota 1:"  
Baca Lintang1 dari input  
Konversi Lintang1 ke radian



Tampilkan "Bujur Kota 1:"  
 Baca Bujur1 dari input  
 Konversi Bujur1 ke radian  
 Tampilkan "Lintang Kota 2:"  
 Baca Lintang2 dari input  
 Konversi Lintang2 ke radian  
 Tampilkan "Bujur Kota 2:"  
 Baca Bujur2 dari input  
 Konversi Bujur2 ke radian  
 Hitung Lat sebagai selisih Lintang2 dan Lintang1  
 Hitung Long sebagai selisih Bujur2 dan Bujur1  
 Hitung a menggunakan rumus:  $a = \sin(\text{Lat}/2)^2 + \cos(\text{Lintang1}) * \cos(\text{Lintang2}) * \sin(\text{Long}/2)^2$   
 Hitung C menggunakan rumus:  $C = 2 * \text{atan2}(\text{sqrt}(a), \text{sqrt}(1 - a))$   
 Hitung jarak d sebagai:  $d = R * C$   
 Tampilkan "jarak antara dua titik tersebut adalah ", d, "kilometer"

## 6. Kesimpulan

- Dalam pengerjaan program dengan bahasa pemrograman Python, kita harus benar-benar teliti dalam menginputkan suatu fungsi untuk menampilkan suatu keluaran pada layar dengan sesuai.
- Kita dapat mengetahui cara mengoperasikan Aritmatika dengan menggunakan bahasa python, seperti operasi penjumlahan, pengurangan, perkalian, dan lain sebagainya. Kita juga diminta untuk membuat source code untuk menghitung jarak antara dua titik bumi dan menurut saya cukup lumayan sulit karena pakai rumusnya susah.

## 7. Cek List (✓)

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

## 8. Formulir Umpan Balik

| No | Elemen Kompetensi | Waktu Pengerjaan | Kriteria |
|----|-------------------|------------------|----------|
| 1. | Latihan Pertama   | 60 Menit         | Menarik  |
| 2. | Latihan Kedua     | 60 Menit         | Menarik  |

Keterangan:

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