

# Struktur Dasar Program

**Bahan Kuliah ST1101\_2C1 Algoritma**

**Sevi Nurafni**

**Fakultas Sains dan Teknologi**

**Universitas Koperasi Indonesia 2024**

# Python



- Bahasa programming tingkat tinggi, direlease oleh Guido van Rossum pada tahun 1991
- Mendukung berbagai paradigma pemrograman. Dalam kuliah ini, hanya akan menggunakan paradigma procedural.
- Interpreter yg tersedia pada beragam sistem operasi:
  - Indentasi untuk menandai blok program
  - case sensitive → perbedaan huruf besar dan kecil berpengaruh
- Python adalah bahasa pemrograman yang loosely typed
  - Tidak perlu mendeklarasikan secara eksplisit tipe data dari variabel

# Struktur Dasar Algoritma

<p><b>Program</b> &lt;JudulProgram&gt; { Spesifikasi Program }</p>
<p><b>KAMUS</b> { Deklarasi type, variabel, konstanta, fungsi, prosedur }</p>
<p><b>ALGORITMA</b> { Deretan langkah algoritmik untuk penyelesaian persoalan } { Ditulis dengan pseudocode atau flowchart }</p>

# Struktur Dasar Program Python

```
# Program <JudulProgram>
```

```
# Spesifikasi Program
```

```
# KAMUS
```

```
# Penjelasan dalam bentuk komentar
```

```
# Deklarasi type, variabel, konstanta, fungsi, prosedur
```

```
# ALGORITMA
```

```
# Deretan langkah algoritmik untuk penyelesaian # persoalan
```

# Program Pertama

- Buatlah program untuk menuliskan “Hello, World!” Ke layar.

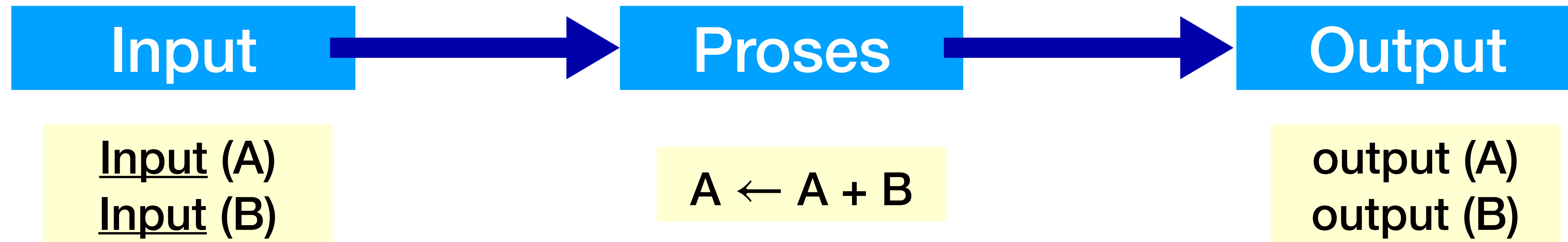
Print adalah perintah untuk mencetak teks ke layar/monitor

```
# Program HelloWorld
# Mencetak Hello, World! ke layar

# KAMUS
# belum diperlukan

# ALGORITMA
print('Hello, World!')
```

# Input - Proses - Output



## Python

```
A = int(input(""))  
B = int(input(""))
```

```
A = A + B
```

```
print(A)  
print(B)
```

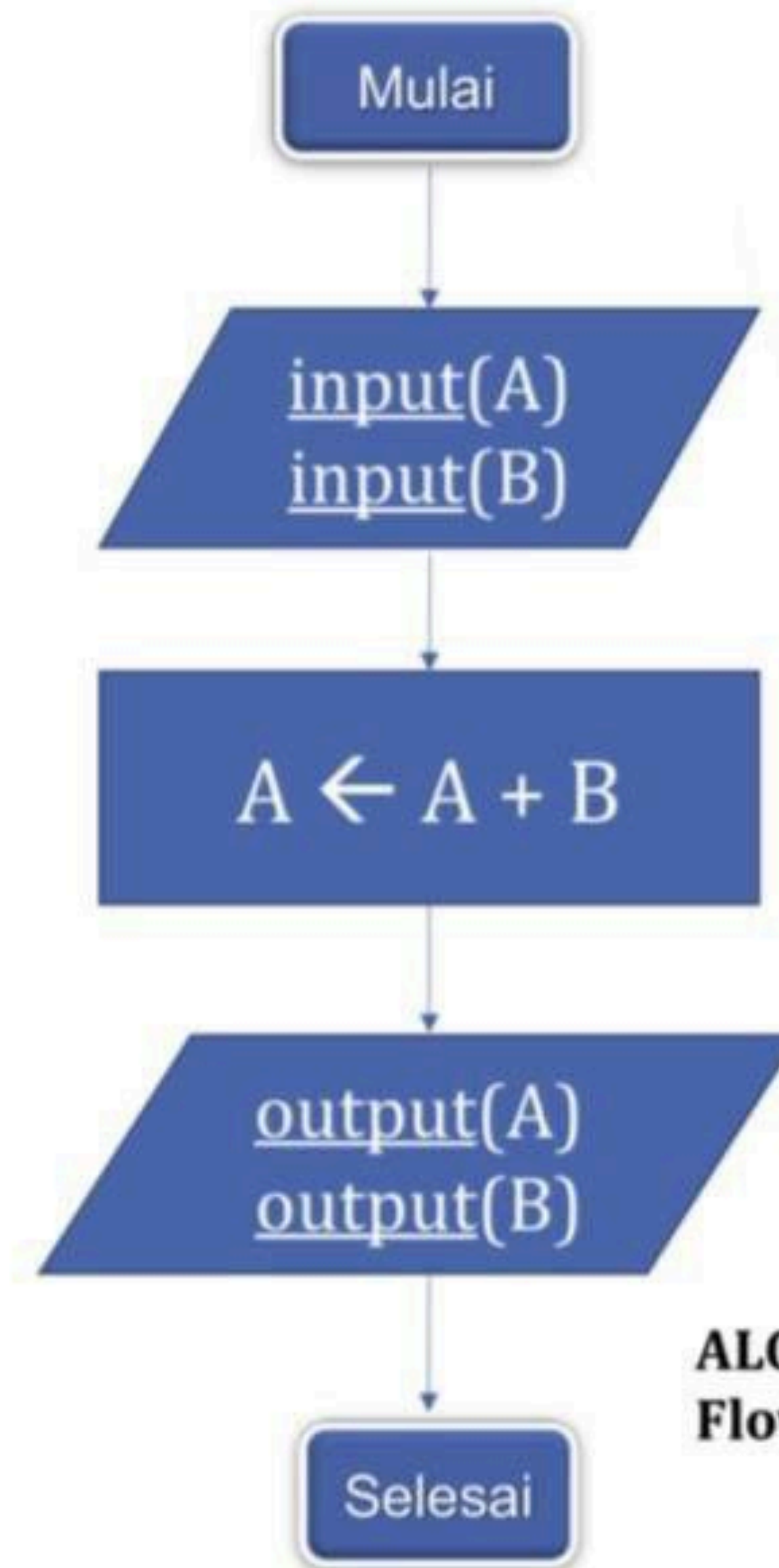


# Struktur Dasar Program

**Program Test**  
{ Spesifikasi Program: menghitung  $A + B$  }

**KAMUS**  
{ Deklarasi variabel }  
A, B : integer

**ALGORITMA - Notasi Algoritmik**  
input(A)  
input(B)  
 $A \leftarrow A + B$   
output(A)  
output(B)



**ALGORITMA -  
Flowchart**

# Contoh Program Python

```
# Program Test
# Spesifikasi : Menghitung nilai A dan B

# KAMUS
# A : int
# B : int

# ALGORITMA
A = int(input())          # input
B = int(input())

A = A + B                 # proses

print(A)                  #output
print(B)
```



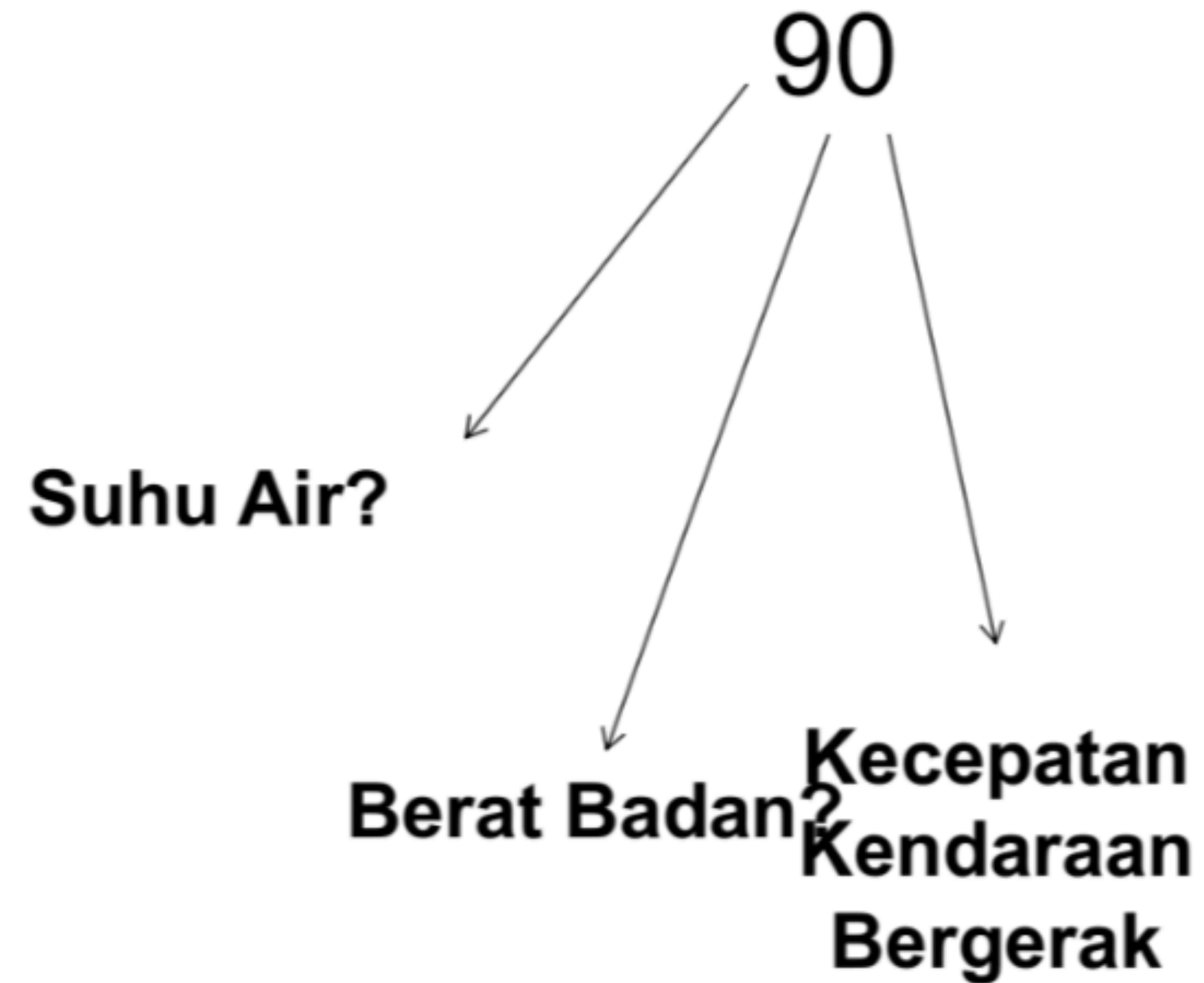
# Komentar



- Dalam bahasa pemrograman komentar adalah bagian program yang tidak dieksekusi
  - Bagian ini hanya digunakan untuk memberikan penjelasan suatu langkah, rumus ataupun bisa hanya berupa keterangan
- Dalam Python komentar dituliskan per baris diawali dengan #
- Contoh:  
**# ini komentar**

# Abstraksi Data

- Kemampuan kita untuk menginterpretasikan suatu data dengan konteks masalahnya.



# Bagian Kamus



- Bagian **Kamus** dipakai untuk mendeklarasikan nama-nama yang digunakan dalam program
- Nama-nama merepresentasikan **data** yang digunakan dalam program
- Python adalah bahasa pemrograman yang ***loosely typed***
  - Tidak perlu mendeklarasikan secara eksplisit tipe data dari variabel
- Namun demikian...
  - Dalam menggunakan variabel tetap harus diketahui dengan baik tipe data apa didefinisikan terhadap variabel tersebut
  - Untuk itu, bagian KAMUS tetap harus dinyatakan walaupun hanya dalam bagian komentar

# Tipe Data Dasar/Primitif

- Disediakan oleh bahasa pemrograman

Python	Domain Nilai
<b>Bool</b>	Nilai boolean: True; False
<b>Numbers</b>	Nilai-nilai numerik. Jenis nilai numerik: <ul style="list-style-type: none"><li>• int : integer/bilangan bulat bertanda (+/-). Contoh: 1; -144; 999; 0</li><li>• float : floating point (real). Contoh: 3.14; 4.01E+1</li><li>• complex : bilangan kompleks à tidak akan digunakan di kelas ini</li></ul>
<b>String</b>	Kumpulan karakter/huruf, ditandai dengan kutip tunggal atau kutip ganda. Contoh: 'xcxcx'
<b>Char</b>	Character: karakter/huruf, ditandai dengan kutip tunggal; Contoh: 'A'; '#'; 'b'

# Variabel

- Variabel digunakan menyimpan suatu nilai yang ber-"tipe data" tertentu sesuai dengan deklarasi
- Merepresentasikan suatu makna di dunia nyata yang ingin diolah dalam program, misalnya:
  - Sum : jumlah beberapa angka
  - Max : nilai maksimum
- Penggunaan variabel:
  - deklarasi (supaya nama dikenal dan diketahui tipe datanya),
  - inisialisasi dan manipulasi nilai

## Python

```
# KAMUS
# i : int
# A : int

# ALGORITMA
...
i = 100
A = i * 50
....
```

# Membuat Nama Variabel yang Benar dan “baik”



- Nama variabel harus dimulai dengan huruf dan dapat diikuti dengan huruf lagi dan angka
  - Tidak boleh ada karakter lain, kecuali: underscore (\_)
- Dalam nama variabel tidak boleh dipisahkan oleh spasi
- Cari nama variabel yang bisa dimengerti dan tidak membingungkan
  - Contoh: sum adalah untuk jumlah, bertipe integer. Jangan gunakan untuk data bertipe lain
- Python adalah bahasa yang **case sensitive**: Kesalahan penulisan huruf besar dan kecil menyebabkan error



# Assignment



- Suatu besaran (dengan tipe tertentu), misalnya variabel, yang telah dikenal dapat diberi nilai/harga
- Pemberian nilai:
  - Pemberian nilai langsung atau disebut sebagai **assignment**
    - Contoh: **A = 10**
  - Dibaca dari piranti masukan (perintah input)
    - Contoh: **A = input()**

# Input

- Perintah **input**: pemberian nilai **variabel** dari piranti masukan,
  - misal: keyboard → dibaca atas masukan dari pengguna
- Perintah di Python: **input(<perintah>)**  
    <perintah> dapat diganti dengan kalimat pengantar input
- Contoh:
  - **A = input()** # A bertipe string
  - **B = input('Masukkan angka =')** # B bertipe string
  - **C = int(input())** # C bertipe integer
  - **D = float(input('Masukkan angka ='))** # D bertipe float

# Type Conversion

- Beberapa fungsi type conversion yang penting diketahui:

No.	Function & Description
1	<b>int(x)</b> Mengkonversi x menjadi integer
2	<b>float(x)</b> Mengkonversi x menjadi nilai floating point (real)
3	<b>str(x)</b> Mengkonversi objek x menjadi representasi stringnya
4	<b>chr(x)</b> Mengkonversi sebuah integer x menjadi character
5	<b>ord(x)</b> Mengkonversi sebuah character x menjadi nilai integernya

# Output

- Perintah **output**: penulisan nilai (variabel/konstanta/hasil ekspresi) ke piranti keluaran, misal: monitor
- Perintah di python: **print**
- Contoh:
  - **print(A)** # menulis isi variabel A ke layar
  - **print('Hello')** # menulis Hello ke layar
  - **print(A \* 4)** # menulis hasil perkalian A\*4
  - **print("Hello World!" + str(a))** # menulis Hello World! <nilai a>

# Ekspresi



- **Ekspresi** adalah kombinasi dari satu atau lebih variabel, konstanta, operator, dan fungsi yang bermakna menurut aturan suatu bahasa pemrograman dan menghasilkan suatu nilai dalam suatu type tertentu
- **Operator** adalah suatu fungsi standar yang disediakan dalam bahasa pemrograman untuk melakukan beberapa hal dasar seperti perhitungan aritmatika, logika, dan relasional.
- Struktur umum ekspresi [biner]: <operan1> <operator> <operan2>
- Hasil dari operasi bergantung pada tipe data operan
- Operan dapat berupa nilai, variable, konstanta, atau ekspresi lain

# Jenis Ekspresi

- Jenis ekspresi menurut **arity** dari operator:
  - Ekspresi **biner**: bentuk dasarnya adalah operasi dengan 2 operan
    - Contoh:  $A + 5$
  - Ekspresi **uner**: bentuk dasarnya adalah operasi dengan 1 operan
    - Contoh: `not (found)`
- Jenis ekspresi menurut tipe data yang dihasilkan:
  - Ekspresi aritmatika: operan bertipe numerik (int/float) dan menghasilkan nilai numerik
  - Ekspresi relasional: operan bertipe numerik (int/float) dan menghasilkan nilai bool/logika
  - Ekspresi logika: operan bertipe bool/logika dan menghasilkan nilai bool/logika



# Aksi Sekuensial



- Aksi sekuensial: sederetan instruksi primitif dan/atau aksi yang akan dilaksanakan (dieksekusi) oleh komputer berdasarkan urutan penulisannya
- Setiap aksi akan mengubah status dari program
  - Jadi setiap aksi sekuensial harus ada awal dan akhir.
  - Dengan kata lain, suatu program harus dimulai dan suatu ketika harus berakhir
- Instruksi ditulis terurut sesuai penulisan per baris
- Perhatikan bahwa:
  - ada program yang akan berubah jika urutan baris instruksinya berubah
  - dan ada juga program yang tidak berubah jika urutan baris instruksinya berubah

# Urutan instruksi tidak mengubah hasil eksekusi

```
# Program Test  
# KAMUS  
# i : int  
# x : float
```

```
# ALGORITMA
```

```
i = int(input())  
x = 100.75
```

```
print(x)  
print(i*2)
```

```
# Program Test  
# KAMUS  
# i : int  
# x : float
```

```
# ALGORITMA
```

```
x = 100.75  
i = int(input())
```

```
print(x)  
print(i*2)
```

# Urutan instruksi mengubah hasil eksekusi

```
# Program Test  
# KAMUS  
# i : int  
# x : float  
  
# ALGORITMA  
i = int(input())  
x = 100.75
```

```
print(x)  
print(i*2)
```

```
# Program Test  
# KAMUS  
# i : int  
# x : float  
  
# ALGORITMA  
i = int(input())  
x = 100.75
```

```
print(i*2)  
print(x)
```

# Blok Program



- Sederetan instruksi yang dieksekusi secara sekuensial dikelompokkan dalam blok program
- Dalam Python, satu blok program ditandai dengan indentasi yang semakin menjorok ke dalam
- Dalam 1 blok program dimungkinkan ada blok program lain yang berada lebih di dalam (inner block)
- Jika instruksi berada dalam 1 blok, maka indentasi harus rapi. Jika tidak, akan error.



# Blok Program

- Sederetan instruksi yang dieksekusi secara sekuensial dikelompokkan dalam blok program
- Dalam Python, satu blok program ditandai dengan indentasi yang semakin menjorok ke dalam
- Dalam 1 blok program dimungkinkan ada blok program lain yang berada lebih di dalam (inner block)
- Jika instruksi berada dalam 1 blok, maka indentasi harus rapi. Jika tidak, akan error.

```
a = int(input("Masukkan angka = "))
if (a > 50):
    print ("Hello World!")
    print ("bye")
else: # a <= 50
    print ("Hello Darling!")
    print ("bye bye")
```

OK!

```
a = int(input("Masukkan angka = "))
if (a > 50):
    print ("Hello World!")
    print ("bye")
else: # a <= 50
    print ("Hello Darling!")
    print ("bye bye")
```

Error

# Contoh: Tinggi Rata-Rata

- Pak Guru menyeleksi 5 orang anak yang akan masuk ke tim basket sekolah. Ia ingin mengetahui tinggi badan rata-rata mereka.
- Buat program menghitung rata-rata dari tinggi badan 5 anak
  - Program akan menerima masukan data tinggi badan untuk 5 orang anak
  - Selanjutnya program menampilkan tinggi rata-rata dari ke lima anak tersebut

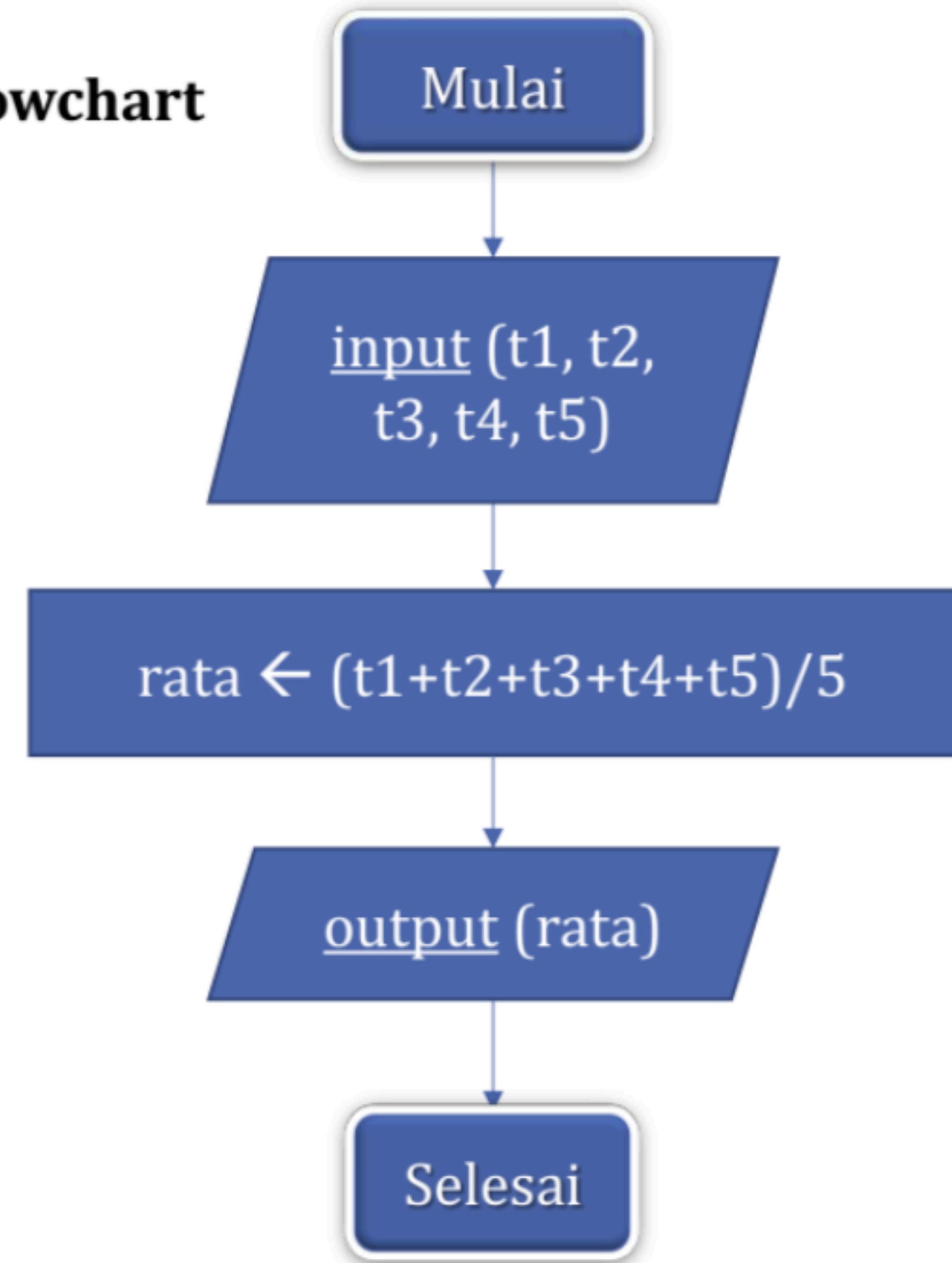


# Contoh: Pseudocode + Flowchart

## Pseudocode

```
input(t1, t2, t3, t4, t5)  
rata  $\leftarrow$  (t1+t2+t3+t4+t5)/5  
output(rata)
```

## Flowchart



# Contoh: Python

```
# Program TinggiRataRata
# Menerima tinggi 5 siswa dan menghitung rata-ratanya

# KAMUS
# t1, t2, t3, t4, t5 : float
# rata : float

# ALGORITMA
t1 = float(input())
t2 = float(input())
t3 = float(input())
t4 = float(input())
t5 = float(input())
rata = (t1 + t2 + t3 + t4 + t5)/5
print (rata)
```

# Latihan: Toko Kelereng

- Sebuah toko menjual kelereng. Berikut adalah tabel harga kelereng berdasarkan warnanya:

Warna kelereng	Harga 1 butir (dalam ratusan rupiah)
Merah	10
Hijau	15
Kuning	20

- Seorang anak membeli kelereng sejumlah  $m$  kelereng merah,  $h$  kelereng hijau, dan  $k$  kelereng kuning. Asumsikan  $m \geq 0$ ,  $h \geq 0$ ,  $k \geq 0$ .
- Hitunglah berapa yang harus dibayarkan anak itu.

**SELAMAT  
BELAJAR**