

# MINGGU KE- 2

## KONSEP ALGORITMA & TIPE DATA



# KONSEP ALGORITMA

## 1. ALGORITMA PE-UBAH

Adalah Variabel yang nilainya BUKAN konstanta (selalu berubah – sesuai dengan kondisi Variabel ter**KINI**)

Sintaks :  $P = Q$

Algoritma :  $P \leftarrow Q$

Arti : Bahwa Nilai P diberi harga Nilai Q  
Nilai P akan SAMA DENGAN nilai Q, & Nilai Q TETAP

## 2. ALGORITMA PERTUKARAN

Berfungsi mempertukarkan masing-masing isi Variabel sedemikian sehingga Nilai dari tiap Variabel akan berubah/bertukar

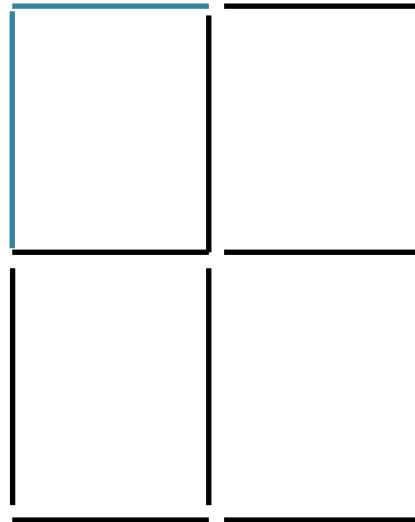
# Studi Kasus

1. Diketahui  $P=10$ ,  $Q=15$  dan  $R=5$ .  
Diberikan Algoritma  $P=Q, Q=R$ , mk Nilai  $P, Q, R$  sekarang?
2. Diketahui Algoritma  $P=11$ ,  $P=P+1$  dan  $Q = P$   
Berapakan Nilai  $P$  dan  $Q$  ? .....
3. Diketahui 3 variabel peubah  $P, Q$  dan  $R$ . Agar isi  $Q$  ditaruh di  $P$ , isi  $R$  ditaruh di  $Q$  dan isi  $P$  ditaruh di  $R$ , maka Algoritma yang dapat ditulis adalah : .....
4. Diketahui 2 peubah  $K = 15$  dan  $L = 25$ . Buat Algoritma untuk mempertukarkan isi  $K$  dan  $L$  : .....

- 
- A diagram consisting of a 2x3 grid of empty squares. The top row has three squares, and the bottom row has three squares. All squares are empty and have black outlines.

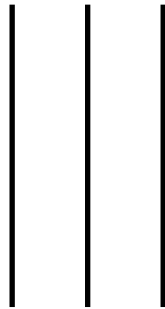
# ANALISA LANJUTAN

Jawab: dengan memindahkan dua buah lidi yang ada pada bagian bawah, seperti dibawah ini



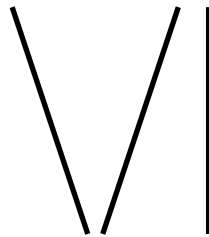
# ANALISA LANJUTAN

2. Ada tiga batang lidi dibawah ini, bagaimana caranya untuk membentuk angka 6 tanpa mematahkannya



# ANALISA LANJUTAN

Jawab: Ketiga buah lidi tersebut akan membentuk angka  
6 romawi



# ANALISA LANJUTAN

3. Budi tidak pernah bolos dalam kelasnya, tetapi dia tidak pernah mengerjakan tugas selama setahun ini. Kerjanya cuma bicara dan Budi juga tidak pernah mengikuti ujian semester, Budi juga bukan murid yang berprestasi. Kenapa Budi tidak pernah mendapat peringatan dari pihak sekolah?  
(menurut Anda apa jawabannya)



# ANALISA LANJUTAN

Jawab: Karena Budi adalah Seorang guru.

Penjelasan:

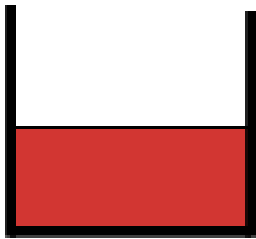
Budi tidak pernah mengerjakan tugas namun membuat tugas, kerjanya cuma bicara menjelaskan materi pelajaran dalam kelas sehingga Budi tidak akan pernah mengikuti ujian semester.

## 4. Algoritma Pertukaran Isi Bejana

**Untuk Latihan Uji Coba Pertukaran Mahasiswa Membawa 2 Gelas air yang berbeda warnanya dan 1 gelas Kosong**

Diberikan dua buah bejana, A dan B; bejana A berisi larutan berwarna merah, bejana B berisi larutan berwarna biru.

**Buatlah pseudocode** untuk menukarkan isi kedua bejana itu sedemikian sehingga bejana A berisi larutan berwarna biru dan bejana B berisi larutan berwarna merah.



Bejana A



Bejana B

# LANJUTAN

**Keadaan Awal Sebelum Pertukaran:**



Bejana A



Bejana B



Bejana C



**PROSES**



**Keadaan Akhir Setelah Pertukaran:**



Bejana A



Bejana B



Bejana C

# Tipe Data Pada Python

Tipe Data	Keterangan
Boolean	Mempunyai dua nilai yaitu true bernilai 1 dan false bernilai 0
String	Terdiri dari karakter/kalimat berupa huruf, angka, dll (diapit tanda " atau ')
Integer	Menyatakan bilangan bulat
Float	Menyatakan bilangan yang mempunyai koma
Complex	Menyatakan pasangan angka real dan imajiner
List	Data untaian yang menyimpan berbagai tipe data, isinya dapat berubah-ubah
Tuple	Data untaian yang menyimpan berbagai tipe data, tapi isinya tidak dapat berubah-ubah
Hexadecimal	Menyatakan bilangan dalam format heksa
Dictionary	Data untaian yang menyimpan berbagai tipe data berupa pasangan penunjuk dan nilai

# Tipe Data Pada Python

```
#tipe data Boolean  
print(True)
```

```
#tipe data String  
print("Belajar Python menyenangkan")
```

```
#tipe data Integer  
print(20)
```

```
#tipe data Float  
print(3.14)
```

```
#tipe data Complex  
print(5j)
```

01

## Hasil Running:

True

Belajar Python menyenangkan

20

3.14

5j

02



## Tipe Data List

List adalah sebuah array yang berisi kumpulan tipe yang tidak sejenis.

```
#tipe data list
kata = ["Belajar", "Python", "di", "Kampus UBSI"]
angka = [10, 50, 100, 1000]
campur = ["Belajar", 100, 7.99, True]
#cetak
print(kata)
print(angka)
print(campur)
```

Hasil Running:

```
['Belajar', 'Python', 'di', 'Kampus UBSI']
[10, 50, 100, 1000]
['Belajar', 100, 7.99, True]
```



## Tipe Data Tuple

Pada tuple anggotanya tidak bisa diubah setelah dideklarasikan, penulisannya menggunakan kurung biasa dan dipisahkan dengan koma.

```
#tipe data tuple
```

```
kata = ("Belajar", "Python", "di", "Kampus UBSI")
```

```
angka = (10, 50, 100, 1000)
```

```
campur = ("Belajar", 100, 7.99, True)
```

```
#cetak
```

```
print(kata)
```

```
print(angka)
```

```
print(campur)
```

### Hasil Running:

```
('Belajar', 'Python', 'di', 'Kampus UBSI')
```

```
(10, 50, 100, 1000)
```

```
('Belajar', 100, 7.99, True)
```

# Tipe Data Dictionary



Bentuk umum tipe data dictionary

```
Nama_variabel = {"key1": "value1", "key2":  
"value2", "key3": "value3" }
```

#Tipe data dictionary

```
data = {1:"Belajar", 2: ["C++", "Python"],  
"Di Kampus": "UBSI",  
"menyerah" : False, "Tahun": 2021}  
print(data)
```

**Hasil Running:**

```
{1: 'Belajar', 2: ['C++', 'Python'], 'Di Kampus':  
'UBSI', 'menyerah': False, 'Tahun': 2021}
```



# OPERATOR ARITMATIKA

Nama	Contoh	Penjelasan
Penjumlahan (+)	$1 + 5 = 6$	Menjumlahkan nilai dari masing-masing operan atau bilangan
Pengurangan (-)	$6 - 2 = 4$	Mengurangi nilai operan di sebelah kiri menggunakan operan di sebelah kanan
Perkalian (*)	$3 * 5 = 15$	Mengalikan operan/bilangan
Pembagian (/)	$15 / 3 = 5.0$	Untuk membagi operan di sebelah kiri menggunakan operan di sebelah kanan
Modulus atau Sisa Bagi (%)	$15 \% 2 = 1$	Mendapatkan sisa pembagian dari operan di sebelah kiri operator ketika dibagi oleh operan di sebelah kanan
Pangkat (**)	$8 ** 2 = 64$	Memangkatkan operan disebelah kiri operator dengan operan di sebelah kanan operator
Pembagian bulat (//)	$17 // 3 = 5$	Sama seperti pembagian. Hanya saja angka dibelakang koma dihilangkan

# OPERATOR ARITMATIKA

## Source Program

## #Penjumlahan

$$a = 1 + 5$$

```
print('1 + 5 =',a)
```

## #Pengurangan

$$a = 6 - 2$$

```
print('6 - 2 =',a)
```

## #Perkalian

$$a = 3 * 5$$

```
print('3 * 5 =',a)
```

## #Pembagian

$$a = 15 / 3$$

```
print('15 / 3 =',a)
```

## #Modulus

$$a = 15 \% 2$$

```
print('15 % 2 =',a)
```

## #Pangkat

a = 8 \*\* 2

```
print('8 ** 2 =',a)
```

## #Pembagian bulat

a = 17 // 3

```
print('17 // 3 =',a)
```

## Output Program

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python> & C:/Users/Lenovo/AppData/Local/
ads/bahan ajar gnp 2324/python/aritmatika1.py"
1 + 5 = 6
6 - 2 = 4
3 * 5 = 15
15 / 3 = 5.0
15 % 2 = 1
8 ** 2 = 64
17 // 3 = 5
PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python>

```

# OPERATOR PERBANDINGAN

Operator	Contoh	Penjelasan
Sama dengan (==)	$1 == 1$	bernilai True karena masing-masing operan memiliki nilai yang sama.
Tidak sama dengan (!=)	$2 != 2$	bernilai False karena operan memiliki nilai yang sama
Lebih besar dari (>)	$5 > 3$	bernilai True karena nilai operan kiri lebih besar dari nilai operan kanan
Lebih kecil dari (<)	$6 < 4$	bernilai False karena nilai operan kiri lebih besar dari nilai operan kanan
Lebih besar dari atau sama dengan (>=)	$5 >= 3$	bernilai True karena nilai operan kiri lebih besar dari nilai operan kanan, atau sama dengan.
Lebih kecil dari atau sama dengan (<=)	$5 <= 3$	bernilai false karena nilai operan kiri lebih besar dari nilai operan kanan, atau sama dengan.



# OPERATOR LOGIKA

Digunakan untuk membuat operasi logika seperti logika AND, OR dan NOT

Nama Operator	Simbol Operator
Logika AND	AND
Logika OR	OR
Negasi/kebalikan	NOT

# LOGIKA AND

Logika AND akan bernilai false (0) apabila nilai semua operandnya atau salah satu bernilai false (0), dan akan bernilai true (1) apabila kedua operand bernilai true (1).

Operand 1	Operand 2	Output
false (0)	false (0)	false (0)
false (0)	true (1)	false (0)
true (1)	false (0)	false (0)
true (1)	true (1)	true (1)

# LOGIKA AND

## Source Program

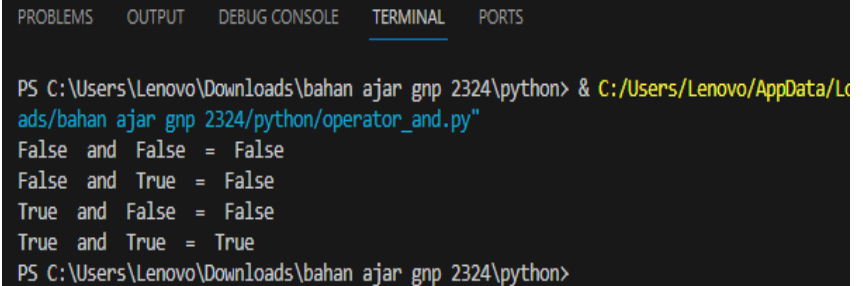
```
# a = false, b = false
a = False
b = False
c = a and b
print ("%r and %r = %r" % (a,b,c))

# a = false, b = true
a = False
b = True
c = a and b
print ("%r and %r = %r" % (a,b,c))

# a = true, b = false
a = True
b = False
c = a and b
print ("%r and %r = %r" % (a,b,c))

# a = false, b = false
a = True
b = True
c = a and b
print ("%r and %r = %r" % (a,b,c))
```

## Output Program



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python> & C:/Users/Lenovo/AppData/Local/Programs/Python/Python39-64/python operator_and.py
False and False = False
False and True = False
True and False = False
True and True = True
PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python>
```

# LOGIKA OR

Logika Or akan menghasilkan output:

Jika salah satu operand atau kedua operand bernilai true (1) akan menghasilkan output true (1), jika kedua operand bernilai false (0) maka akan menghasilkan output false (0).

Operand1	Operand2	Output
false (0)	false (0)	false (0)
false (0)	true (1)	true (1)
true (1)	false (0)	true (1)
true (1)	true (1)	true (1)



# LOGIKA OR

## Source Program

```
# a = false, b = false
a = False
b = False
c = a or b
print ("%r OR %r = %r" % (a,b,c))

# a = false, b = true
a = False
b = True
c = a or b
print ("%r or %r = %r" % (a,b,c))

# a = true, b = false
a = True
b = False
c = a or b
print ("%r or %r = %r" % (a,b,c))

# a = false, b = false
a = True
b = True
c = a or b
print ("%r or %r = %r" % (a,b,c))
```

## Output Program

```
PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python> & C:/Users/Lenovo/AppData/
ads/bahan ajar gnp 2324/python/operator_or.py"
False OR False = False
False or True = True
True or False = True
True or True = True
PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python>
```

# LOGIKA NOT

Adalah kebalikan dari nilai yang ada pada operandnya

Misal:



A = False maka not A adalah True

A = True maka not A adalah False






# OPERATOR PENUGASAN (ASSIGNMENT)

- Digunakan untuk memberikan nilai ke dalam sebuah variabel

Operator	Contoh	Penjelasan
Sama dengan (=)	$a = 5$	Artinya memberikan nilai pada variabel 5
Tambah sama dengan (+=)	$a += 5$  $a = a + 5$	artinya memberikan nilai pada variabel a dengan nilai a sebelumnya, lalu ditambah 5.
Kurang sama dengan (-=)	$a -= 3$  $a = a - 3$	artinya memberikan nilai pada variable a dengan nilai a sebelumnya dikurangi 3

# OPERATOR PENUGASAN (ASSIGNMENT)

Operator	Contoh	Penjelasan
Kali sama dengan (*=)	$a *= 10$  $a = a * 10$	Artinya memberikan nilai pada variabel a dengan nilai a sebelumnya dikali 10
Bagi sama dengan (/=)	$a /= 4$  $a = a / 4$	Artinya memberikan nilai pada variabel a dengan nilai a sebelumnya dibagi 4
Pangkat sama dengan (**=)	$a **= 2$  $a = a ** 2$	Artinya memberikan nilai pada variabel a dengan nilai a sebelumnya dipangkatkan 2

# OPERATOR PENUGASAN (ASSIGNMENT)

## Source Python

```
# input variabel a untuk mengisi nilai
```

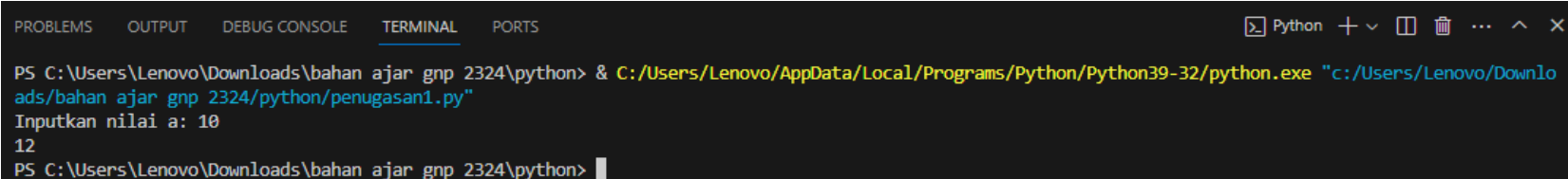
```
a = int(input("Inputkan nilai a: "))
```

```
# tambahkan dengan 2
```

```
a += 2
```

```
print(a)
```

## Output:



```
PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python> & C:/Users/Lenovo/AppData/Local/Programs/Python/Python39-32/python.exe "c:/Users/Lenovo/Downlo
ads/bahan ajar gnp 2324/python/penugasan1.py"
Inputkan nilai a: 10
12
PS C:\Users\Lenovo\Downloads\bahan ajar gnp 2324\python> |
```

# Menggabungkan Nilai string

```
#Penggabungan dua string
kata1 = "Belajar Bahasa Pemrograman Python "
kata2 = "Sangat Menyenangkan"
print("Kata1: ",kata1)
Print("Kata2: ",kata2)
#kata pertama dan kedua digabungkan
gabung = kata1 + kata2
print("Hasil Penggabungan kata1 dan kata2")
print(gabung)
```

Hasil Running:

Belajar Bahasa Pemrograman Python Sangat Menyenangkan

# Fungsi Len

Untuk menghitung jumlah karakter digunakan fungsi **len()**

```
#Fungsi Len  
#Untuk Menghitung Panjang Karakter  
kata = "Belajar Bahasa Pemrograman Python"  
jumlah_karakter=len(kata)  
print(jumlah_karakter)
```

**Hasil Running:**  
33



# Fungsi index()

untuk mengetahui posisi karakter dalam kalimat.

#fungsi index

kata = 'Aisah Zahra'

#dimana posisi karakter Z

```
print (kata.index('Z'))
```

#dimana posisi karakter r

```
print (kata.index('r'))
```

**Hasil Running:**

6

9

# Membuat Dokumentasi Kode Program

## **Dokumentasi kode program adalah:**

Bentuk hard copy dari modul program yang telah ditulis termasuk property dari kode program tersebut (tanggal dibuat, versi, nama modul, pembuat, nomor dsb).

## **Tahapan membuat Dokumentasi Kode Program :**

1. Mengidentifikasi kode program
2. Membuat dokumentasi program dengan memberikan penjelasan pada setiap baris program
3. Melakukan generate dokumentasi

# Tugas Mandiri

Ibu pergi ke pasar membeli telur sebanyak 5 kilogram untuk membuat kue, harga 1 kilo gram telur adalah 26000 perkilogram. Untuk pergi ke pasar ibu harus naik angkot pp (pulang pergi) dengan tarip Rp 3500 sekali naik angkot. Pertanyaan: Berapakah sisa uang jika ibu membawa uang sebesar Rp 200.000,-

## Identifikasi masalah

**Input:** berat telur(brt), harga telur (hrg), transport(ongkos)  
uang ibu(uang)

**Output:** sisa uang(sisa)

Buatlah Programnya dengan Python!

## Latihan Pertemuan 2

Seorang pedagang mangga menjual dagangannya yang setiap kg mangga dihargai dengan harga tertentu. Setiap pembeli membayar harga mangga yang dibeli nya berdasarkan berat.

Buatlah algoritma untuk menentukan harga yang harus dibayar pembeli.

- **Identifikasi masalah**
- **Input:** harga per kg(hrg), berat pembelian(brt)
- **Output:** harga yang dibayar pembeli(byr)