

# Pertemuan 4

# Pseudocode dan Flowchart

Tim Ajar Dasar Pemrograman 2023

# Tujuan

- Mahasiswa memahami dan mampu menjelaskan tentang Pseudo-code
- Mahasiswa menjabarkan dan mampu menjelaskan tentang flowchart dan symbol-simbolnya
- Mahasiswa memahami dan mampu menggunakan pseudocode dan flowchart untuk menyelesaikan studi kasus

# PSEUDOCODE

# Pseudocode

- Pseudocode adalah deskripsi tingkat tinggi dari algoritma pemrograman komputer dengan **menggunakan notasi** yang **mirip** dengan **kode pemrograman**, tetapi tidak terikat oleh aturan baku dalam suatu bahasa pemrograman tertentu.
- Pseudocode digunakan untuk menjelaskan langkah-langkah penyelesaian suatu masalah secara jelas dan mudah dipahami oleh manusia, bukan oleh mesin.
- Ditulis dalam bahasa Inggris atau Indonesia sederhana
- Sintaks digunakan untuk menjelaskan **perintah** (misalnya: “jika”, “ulangi”, “sampai”, “if”, “repeat”, “until”)

# Menulis Pseudocode

Sintaks dikelompokkan dalam 6 operasi dasar berikut ini :

1. Menerima data (Input)
2. Menampilkan data (Output)
3. Menghitung data (Compute)
4. Menyimpan data (Store)
5. Membandingkan data (Compare)
6. Melakukan pengulangan (Loop)



# 1. Menerima data (Input)

- Sewaktu komputer menerima data atau *input*, maka perintah yang biasa digunakan adalah **“Read”, “Get”, atau “Baca”**
- Contoh:
  - Read Bilangan
  - Get kode\_pajak
  - Baca nama\_mahasiswa



## 2. Menampilkan data (Output)

- Sewaktu komputer menampilkan data ataupun *output*, maka perintah yang biasa digunakan adalah **“Print”**, **“Write”**, **“Put”**, **“Output”**, **“Display”** atau **“Cetak”**
- Contoh:
  - Print “Universitas Bina Nusantara”
  - Cetak “Metode Perancangan Program”
  - Output Total

### 3. Menghitung data (Compute)

- Untuk melakukan operasi aritmetika digunakan :
  - + Untuk penjumlahan (add)
  - Untuk pengurangan (subtract)
  - \* Untuk perkalian (multiply)
  - / Untuk pembagian (divide)
  - () Untuk kurung
- Statement **“Compute”**, **“Calculate”** ataupun **“Hitung”** juga dapat digunakan.
- Contoh:  
$$\text{Compute } X = (1+2) * 3$$
$$\text{Total} = \text{Total} + 10$$



## 4. Menyimpan data (Store)

- Ada tiga cara untuk menyimpan data :
  - Memberikan **nilai awal**, menggunakan *statement* “**Initialize**” atau “**Set**”
  - Memberikan **nilai sebagai hasil dari suatu proses**, maka tanda “=” digunakan
  - Untuk **menyimpan suatu nilai** maka *statement* “**Save**” atau “**Store**” digunakan
- Contoh:
  - Set Umur to 17
  - Harga = 1000
  - Store 50 to Discount



## 5. Membandingkan data (Compare)

- Salah satu operasi terpenting yang dapat dilakukan komputer adalah membandingkan dan memilih salah satu alternatif solusi.
- Perintah yang digunakan : **“IF”, “THEN” dan “ELSE” (atau “JIKA”, “MAKA”, “SELAIN ITU”)**
- Contoh  
IF Pilih=1 THEN  
    Discount = 0.1 \* harga  
ELSE  
    Discount = 0.2 \* harga  
ENDIF



## 6. Melakukan pengulangan (Loop)

- Jika ada beberapa perintah yang harus diulang, maka dapat digunakan perintah **“DOWHILE”** dan **“ENDDO”**.

- Contoh

Bil = 0

DOWHILE bil < 10

  cetak bil

  bil = bil + 1

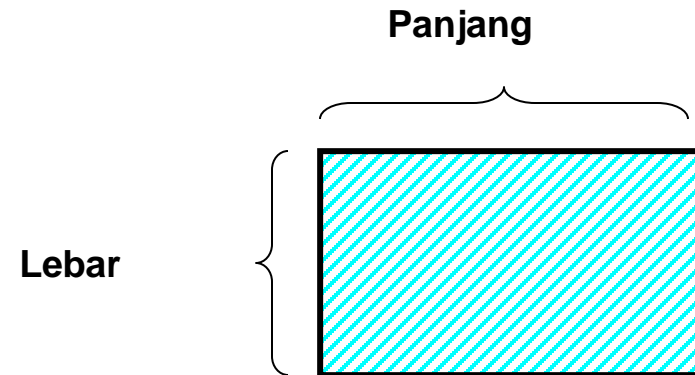
ENDDO

```
literasi 1  
bil = 0  
bil = 0 + 1 = 1  
literasi 2  
bil = 1  
bil = 1 + 1 = 2  
literasi 3  
bil = 2  
bil = 2 + 1 = 3  
.  
.  
.  
literasi 9  
bil = 8  
bil = 8 + 1 = 9
```

# Contoh

Buatlah algoritma menggunakan pseudocode untuk menghitung luas persegi panjang, dengan rumus :

$$\text{Luas} = \text{Panjang} * \text{Lebar}$$





# Jawaban - Pseudocode

Algoritma: hitung\_luas\_persegi

{dibaca Panjang dan lebar dari piranti masukan. Hitunglah luas persegi Panjang tersebut}

Deklarasi :

panjang, lebar, luas: int

Deskripsi :

print "Input & Hitung Luas Persegi Panjang"

print "Masukkan Panjang = "

read panjang;

print "Masukkan Lebar = "

read lebar

luas = panjang \* lebar

print "Luas Persegi Panjang = "

print luas

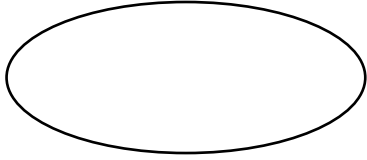
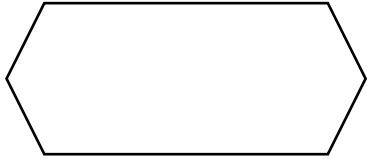


# FLOWCHART



# Flowchart


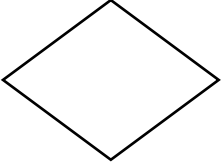
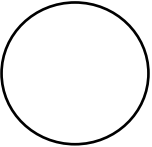
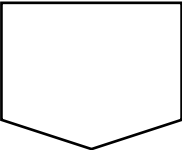

- Flowchart merupakan sebuah bagan dengan **simbol-simbol** tertentu yang digunakan untuk **menjelaskan urutan proses** dan hubungan antar proses lainnya pada **sebuah program**.
- Merupakan cara penyajian dari suatu algoritma.
- Ada 2 macam Flowchart :
  1. System Flowchart → urutan proses dalam system dengan menunjukkan alat media input, output serta jenis media penyimpanan dalam proses pengolahan data.
  2. Program Flowchart → urutan instruksi yang digambarkan dengan symbol tertentu untuk memecahkan masalah dalam suatu program.

# Symbol-symbol Flowchart

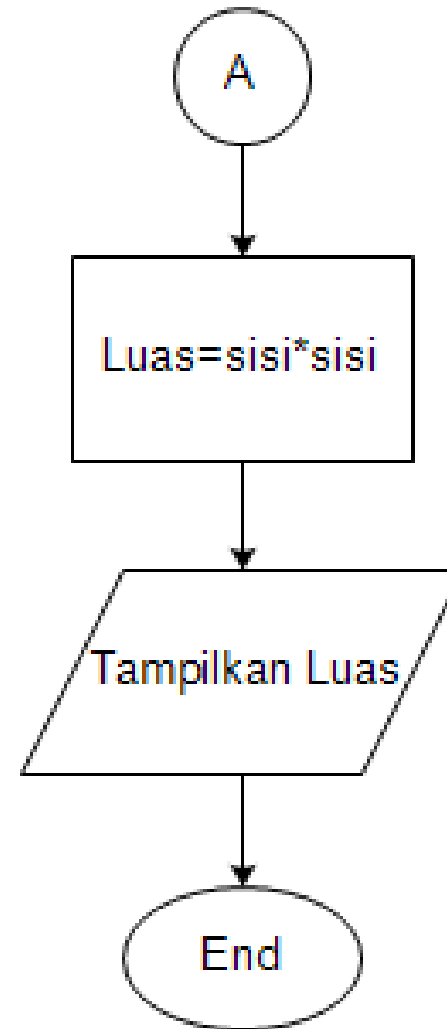
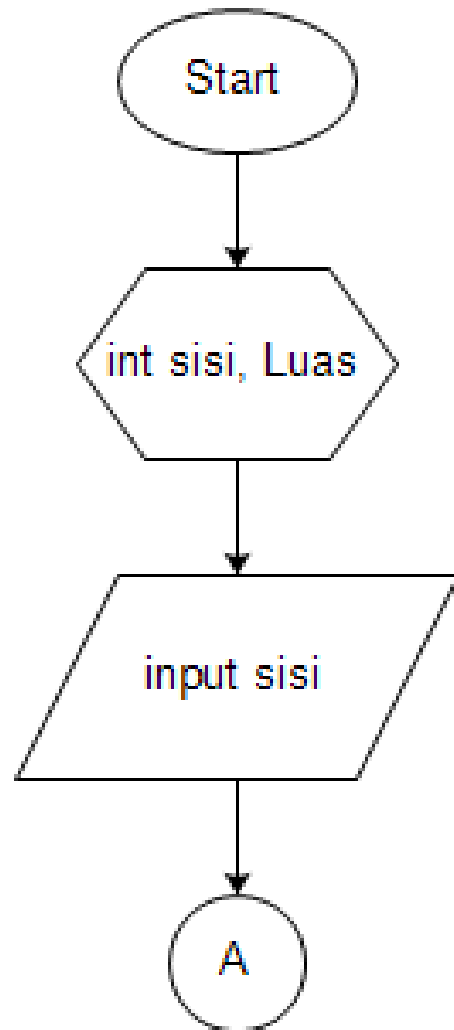
Symbol	Nama	Deskripsi
	Terminator	Simbol untuk permulaan (start) dan akhir (end) dalam sebuah proses
	Preparation <small>deklarasi variabel</small>	Simbol untuk mempersiapkan penyimpanan yang akan digunakan
	Input-output	Simbol yang menyatakan proses input dan output
	Flow Line (Garis Alir)	Arah aliran program



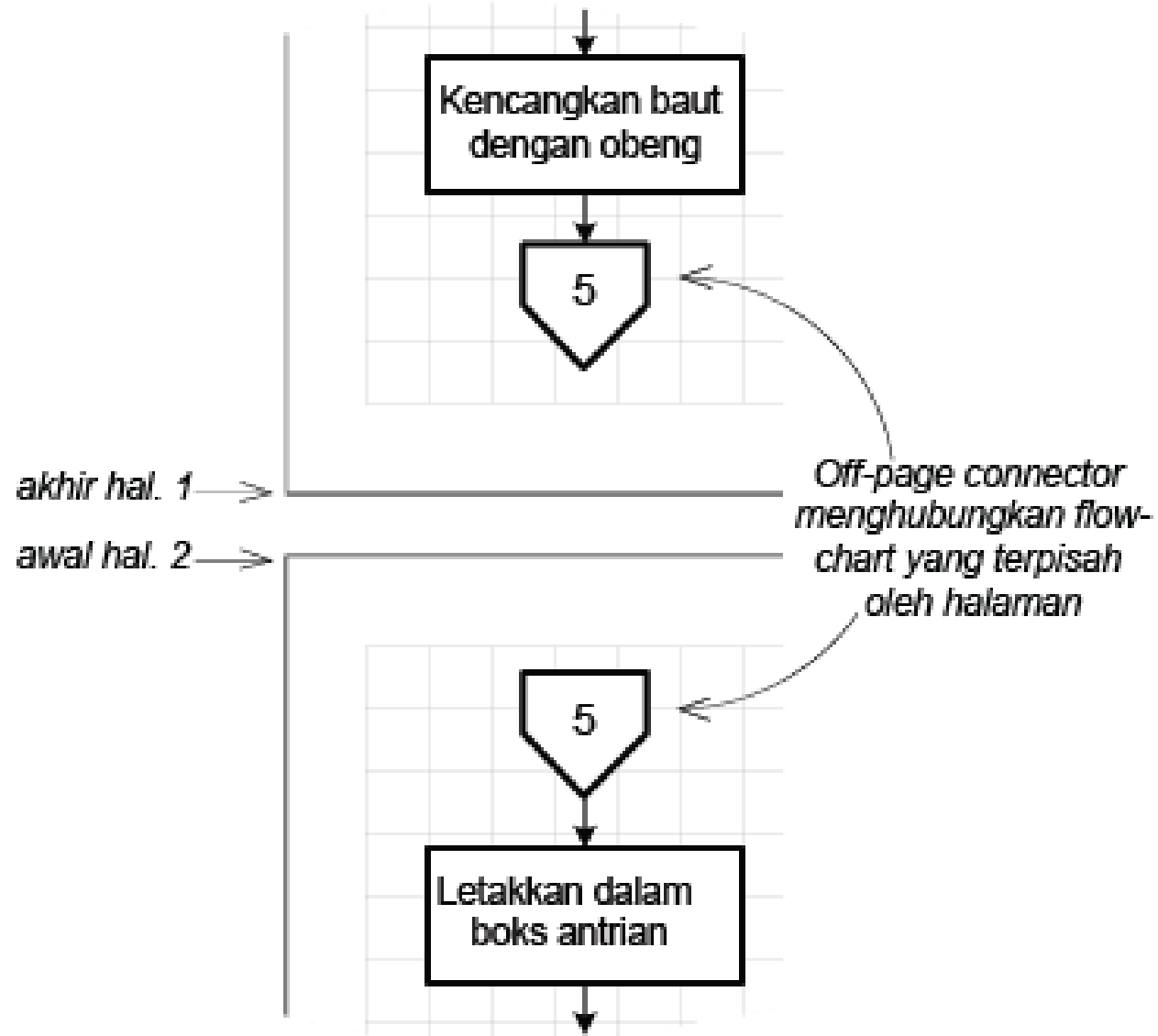
# Symbol-symbol Flowchart

Symbol	Nama	Deskripsi
	Processing simbol	Simbol yang menunjukkan pemrosesan oleh komputer
	Percabangan	Simbol yang menunjukan terdapat dua pilihan output
	On Page Connector	Penghubung bagian-bagian flowchart yang ada pada satu halaman
	Off Page Connector	Penghubung bagian-bagian flowchart yang berada pada halaman yang berbeda
	Predefined Process	Simbol untuk pelaksanaan suatu bagian (sub-program)/fungsi/prosedur

# Contoh Penggunaan On Page Connector



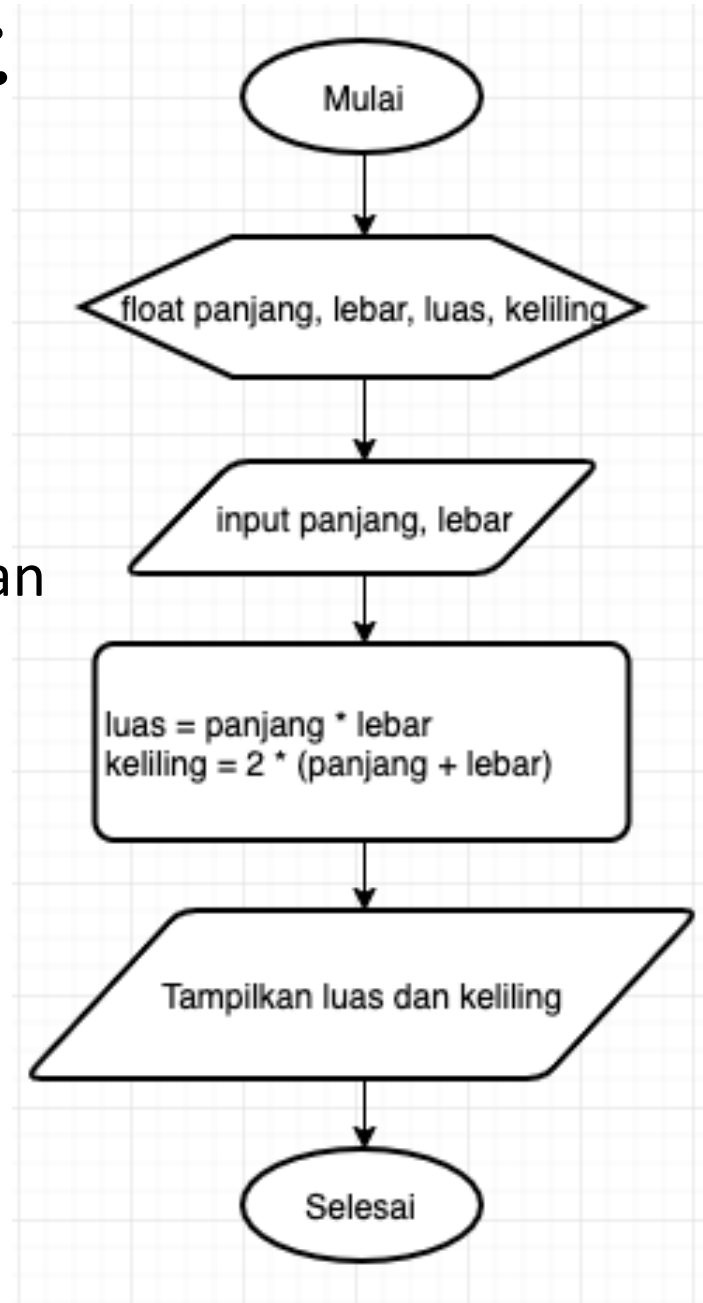
# Contoh Penggunaan Off Page Connector



# Contoh Flowchart

## Contoh Studi Kasus:

Pak Adi mempunyai sebuah kebun berbentuk persegi panjang, buatlah flowchart untuk menghitung luas dan keliling kebun pak Adi!



# Contoh Studi Kasus

Pak Adi mempunyai sebuah kebun berbentuk persegi Panjang. Pak Adi ingin membuat pagar kayu untuk mengelilingi kebun tersebut.

Sebelum membuat program untuk membantu pak adi menghitung keliling kebunnya, maka bantulah pak adi untuk mengidentifikasi variable dan tipe data beserta algoritmanya!

# Contoh Studi Kasus

## 1. Menentukan Algoritma

Input: panjang, lebar

Output: keliling

Proses:

1. input panjang, lebar
2. keliling = 2 x (panjang + lebar)
3. Output keliling

## 2. Mengidentifikasi variable dan jenis tipe data berdasarkan algoritma

Variabel	Tipe data
panjang	int
lebar	int
keliling	int

### PSEUDOCODE

Algoritma : hitung\_keliling\_persegi

{dibaca Panjang dan lebar dari piranti masukan.  
Hitunglah keliling persegi}!

Deklarasi :

panjang, lebar, keliling: int

Deskripsi :

read panjang

read lebar

keliling = panjang \* lebar

print keliling

# Contoh Studi Kasus

## PSEUDOCODE

Algoritma : hitung\_keliling\_persegi

{dibaca Panjang dan lebar dari piranti masukan.  
Hitunglah keliling persegi}!

Deklarasi :

panjang, lebar, keliling: int

**Baris 8,9,10**

Deskripsi :

read panjang

**Baris 12,13**

read lebar

keliling = panjang \* lebar

**Baris 15**

print keliling

**Baris 17**

```
1  import java.util.Scanner;
2
3  public class Coba {
4      Run | Debug
5      public static void main (String [] args){
6
7          Scanner input =new Scanner(System.in);
8
9          int panjang;
10         int lebar;
11         int keliling;
12
13         panjang = input.nextInt();
14         lebar = input.nextInt();
15
16         keliling = 2 * (panjang + lebar);
17
18         System.out.println(keliling);
19     }
20 }
```

# Contoh Studi Kasus

```
1  import java.util.Scanner;
2
3  public class Coba {
4      Run | Debug
5      public static void main (String [] args){
6
7          Scanner input =new Scanner(System.in);
8
9          int panjang;
10         int lebar;
11         int keliling;
12
13         panjang = input.nextInt();
14         lebar = input.nextInt();
15
16         keliling = 2 * (panjang + lebar);
17
18         System.out.println(keliling);
19     }
20 }
```

Baris 8,9,10

Baris 12,13

Baris 15

Baris 17

