

# ALGORITMA DAN PEMROGRAMAN

[Komang Aryasa | [Pertemuan 04]

# Outline



- Pengertian instrukti seleksi
- Seleksi if tunggal
- Contoh seleksi if-tunggal
- Seleksi if else
- Conroh seleksi if/else



**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI**  
**DIREKTORAT JENDERAL PENDIDIKAN TINGGI, RISET, DAN TEKNOLOGI**  
**DIREKTORAT PEMBELAJARAN DAN KEMAHASISWAAN**





# Instruksi Seleksi/Pemilihan

- Instruksi yang memungkinkan untuk memilih satu operasi dari beberapa operasi yang disediakan berdasarkan suatu persyaratan.
- Ada dua format utama dari instruksi seleksi yaitu:
  - If / then / else
  - case



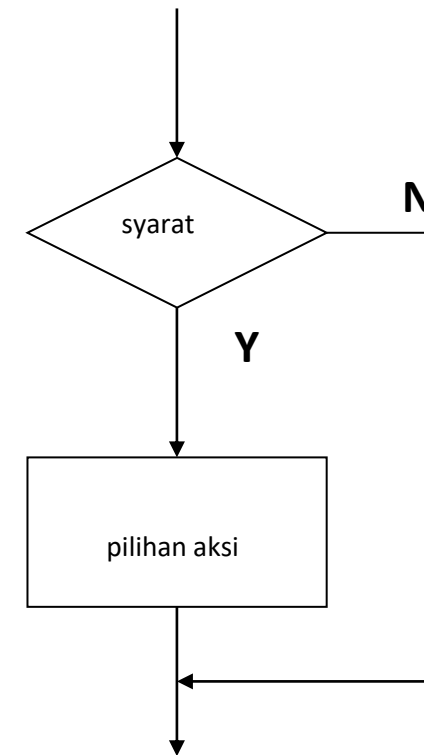
# If / then / else

- Instruksi memiliki format:
  - 1 pilihan
  - 2 pilihan
  - Lebih dari 2 pilihan
- Untuk 1 pilihan bentuknya sederhana, yaitu:
  - if (syarat) then pilihan aksi
  - berarti pilihan akan diproses apabila syarat di penuhi, dan tidak diproses bila syarat tidak terpenuhi.

```
if (syarat)  
  then aksi  
endif
```

contoh : **if** (  $x > 100$  )  
          **then**  $x \leftarrow x + 5$   
          **endif.**

**Kalu nilai x lebih dari 100 maka  
nilainya ditambah 5.**

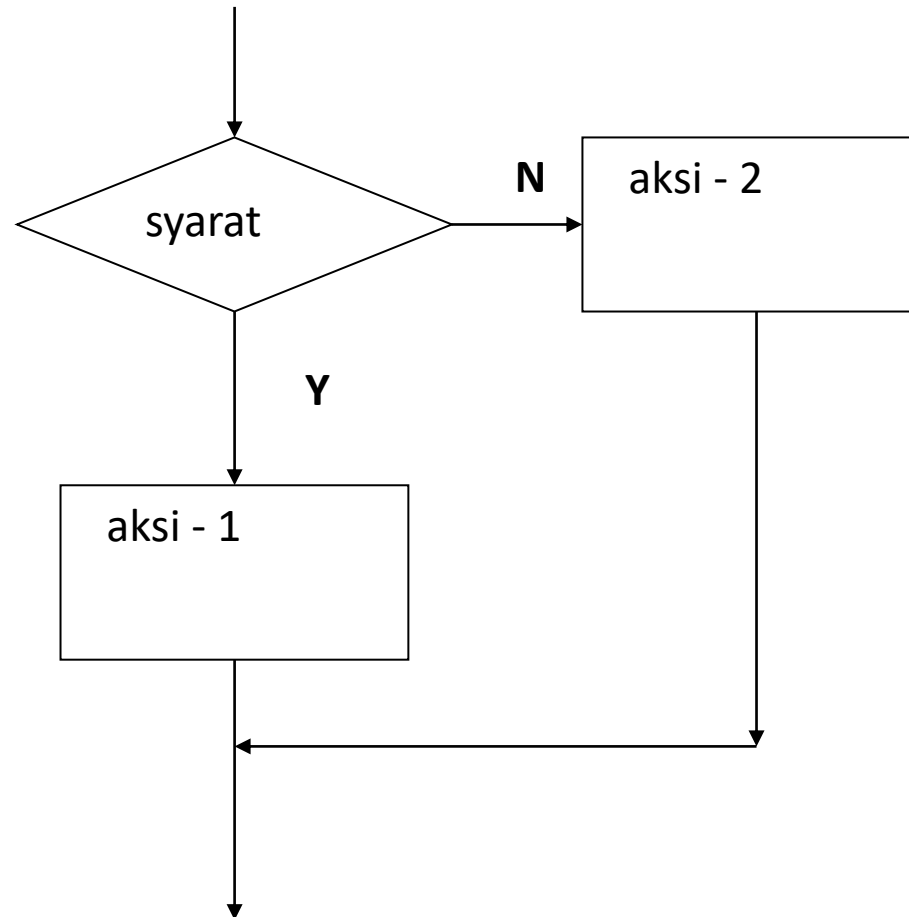


## Bentuk 2 pilihan:

```
if ( syarat )  
    then aksi-1  
    else aksi-2  
endif.
```

contoh :

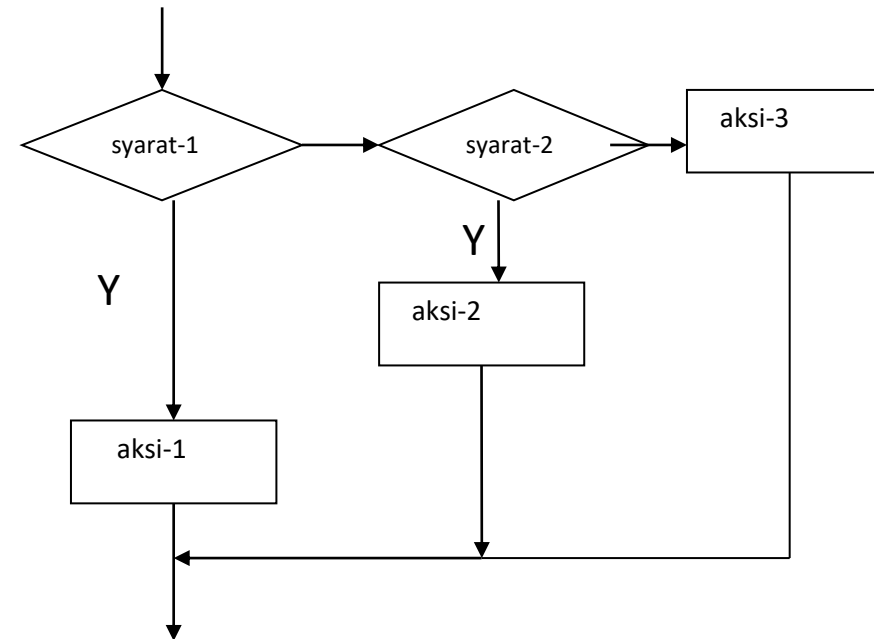
```
if ( a > 0 )  
    then write ("bilangan ini positif ")  
    else write ("bilangan ini negatif ")  
endif.
```





## Bentuk bersusun (lebih dari 2 pilihan) :

```
if ( syarat-1 )  
  then aksi-1  
  else if ( syarat-2 )  
    then aksi-2  
    else aksi-3  
  endif  
endif.
```





Bentuk yang lain:

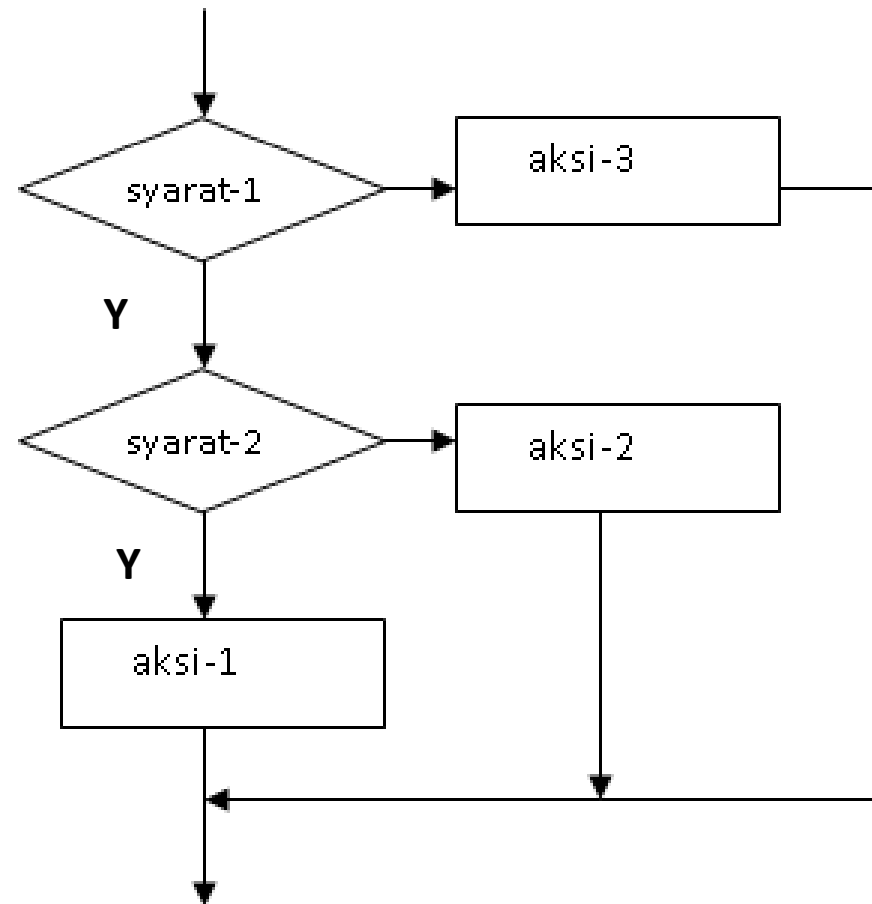
```
if ( syarat-1 )  
  then if ( syarat-2 )  
    then aksi-1  
    else aksi-2  
  endif  
  else aksi-3  
endif.
```

Contoh:

```
If (  $x > 5$  )  
  then if (  $y > 3$  )  
    then  $z = 4x + 5y$   
    else  $z = 2x + 2y$   
  else  $z = 3x + 3y$ 
```

Berapa z bila:

- $x = 3, y = 5$
- $x = 5, y = 3$





# if/then/else dalam C++

- Pada bahasa C / C++ format-nya sbb:

```
if (syarat) {  
    pilihan-1  
}  
else {  
    pilihan-2  
}
```

- **Perlu diperhatikan ! C/C++ tidak memakai THEN**



# Contoh-1

Algoritma berikut ini menunjukkan pemakaian instruksi pemilihan if/then/else untuk menentukan apakah bilangan bulat yang dimasukkan lewat keyboard adalah bilangan yang genap atau bilangan ganjil.

Langkah utama yang harus dikerjakan pada persoalan ini, adalah: memasukkan angka tersebut, kemudian angka ini dibagi dua, bila bersisa maka ia bilangan ganjil, bila tidak bersisa maka ia bilangan genap.



# Algoritma

## Algoritma Pemilihan\_1

{ contoh algoritma untuk menunjukkan pemakaian instruksi pemilihan.  
algoritma ini menerima satu bilangan bulat kemudian memeriksanya  
apakah bilangan genap atau bilangan ganjil }

## Definisi Variabel / DEKLARASI

**integer** bilangan;

## Rincian Langkah / DESKRIPSI

**write** ("masukkan satu bilangan bulat : ");

**read** (bilangan);

**if** ( bilangan % 2 == 0 )

**then write** ( "bilangan genap ! ");

**else write** ( "bilangan ganjil ! ");

**endif**



# Contoh-2

- Buat algoritma dan program untuk memilih bilangan terbesar dari tiga bilangan (A,B,C) yang di-input



# Algoritma



## Algoritma Pemilihan

{ contoh algoritma ini menerima 3 bilangan bulat kemudian menetapkan bilangan yang terbesar }

### Deklarasi

**integer** A, B, C, maks;

### Deskripsi

**write** ("masukkan bilangan 1 : ");

**read** ( A );

**write** ("masukkan bilangan 2 : ");

**read** ( B );

**write** ("masukkan bilangan 3 : ");

**read** ( C );

**if** ( A > B )

**then if** ( A > C )

**then write** (" maksimum = ", A );

**else write** (" maksimum = ", C );

**endif**

**else if** ( B > C )

**then write** (" maksimum = ", B );

**else write** (" maksimum = ", C );

**endif**

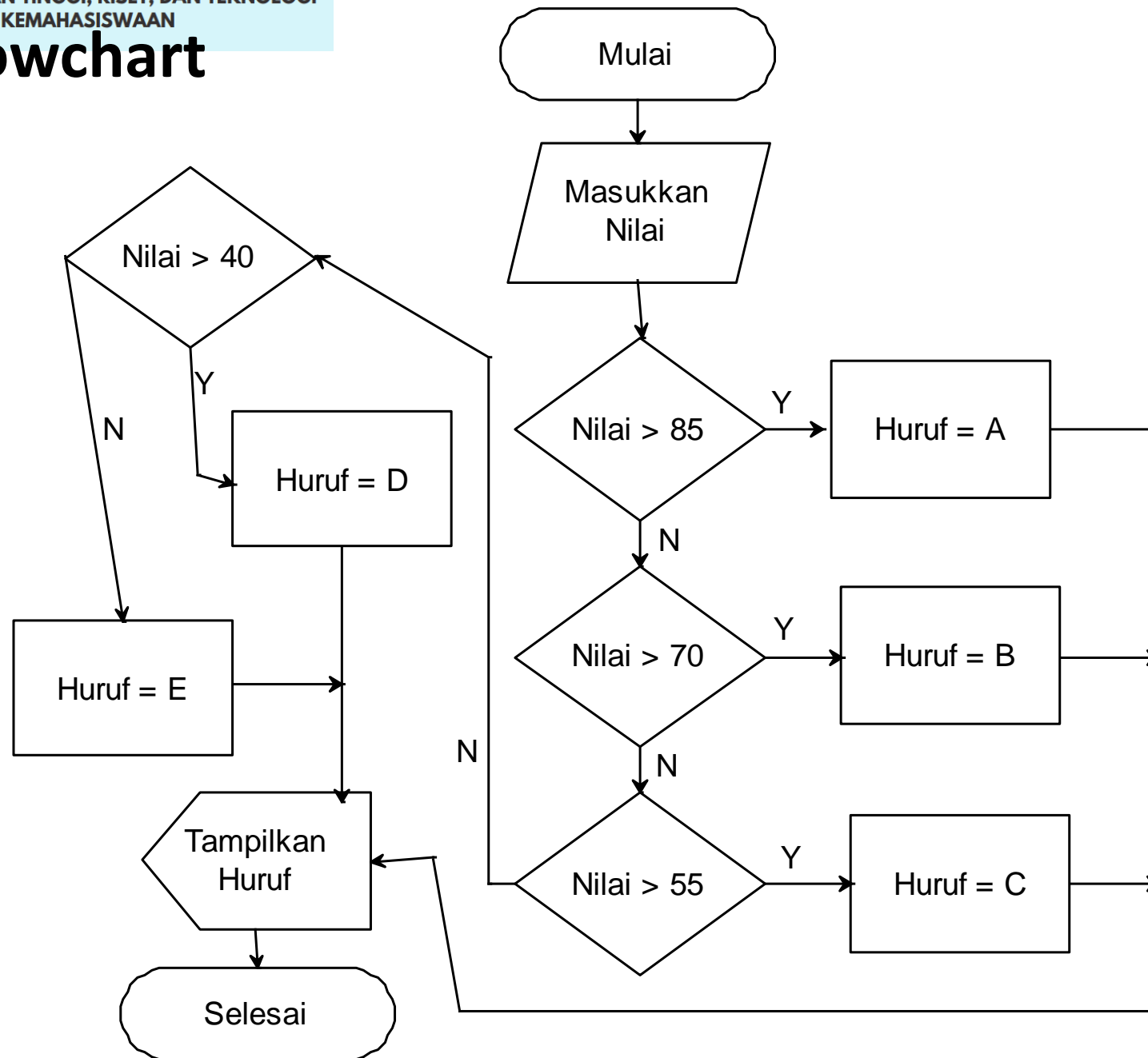
**endif**



# Contoh-3

- Buat sebuah program yang menerima sebuah nilai angka kemudian menetapkan nilai huruf berdasarkan aturan berikut:
  - Nilai A bila angka  $> 85$
  - Nilai B bila angka antara 71 sampai 85
  - Nilai C bila angka antara 56 sampai 70
  - Nilai D bila angka antara 41 sampai 55
  - Nilai E bila angka  $< 41$

# Flowchart







# Algoritma



## Algoritma Nilai Huruf

### Deklarasi

```
int Nilai;  
char Huruf;
```

### Deskripsi

```
write(`Masukkan Nilai : `);  
read(Nilai);  
if (Nilai > 85) then Huruf = `A`  
else if (Nilai > 70) then Huruf=`B`  
    else if (Nilai > 55) then Huruf = `C`  
        else if (Nilai > 40) then Huruf=`D`  
            else Huruf = `E`;  
write(`Nilai huruf = `,Huruf);
```



# Instruksi Case / Algoritma

Instruksi **case** digunakan sebagai instruksi pemilihan dimana aksi yang akan dilakukan bergantung pada nilai dari **satu macam** variabel saja, dengan kata lain variabel yang menentukan pilihan aksi mungkin memiliki banyak macam nilai dan setiap nilainya berkaitan dengan satu macam aksi.



# Bentuk case

**case** ( variable )

**nilai-1** : aksi-1;

**nilai-2** : aksi-2;

**nilai-3** : aksi-3;

.....

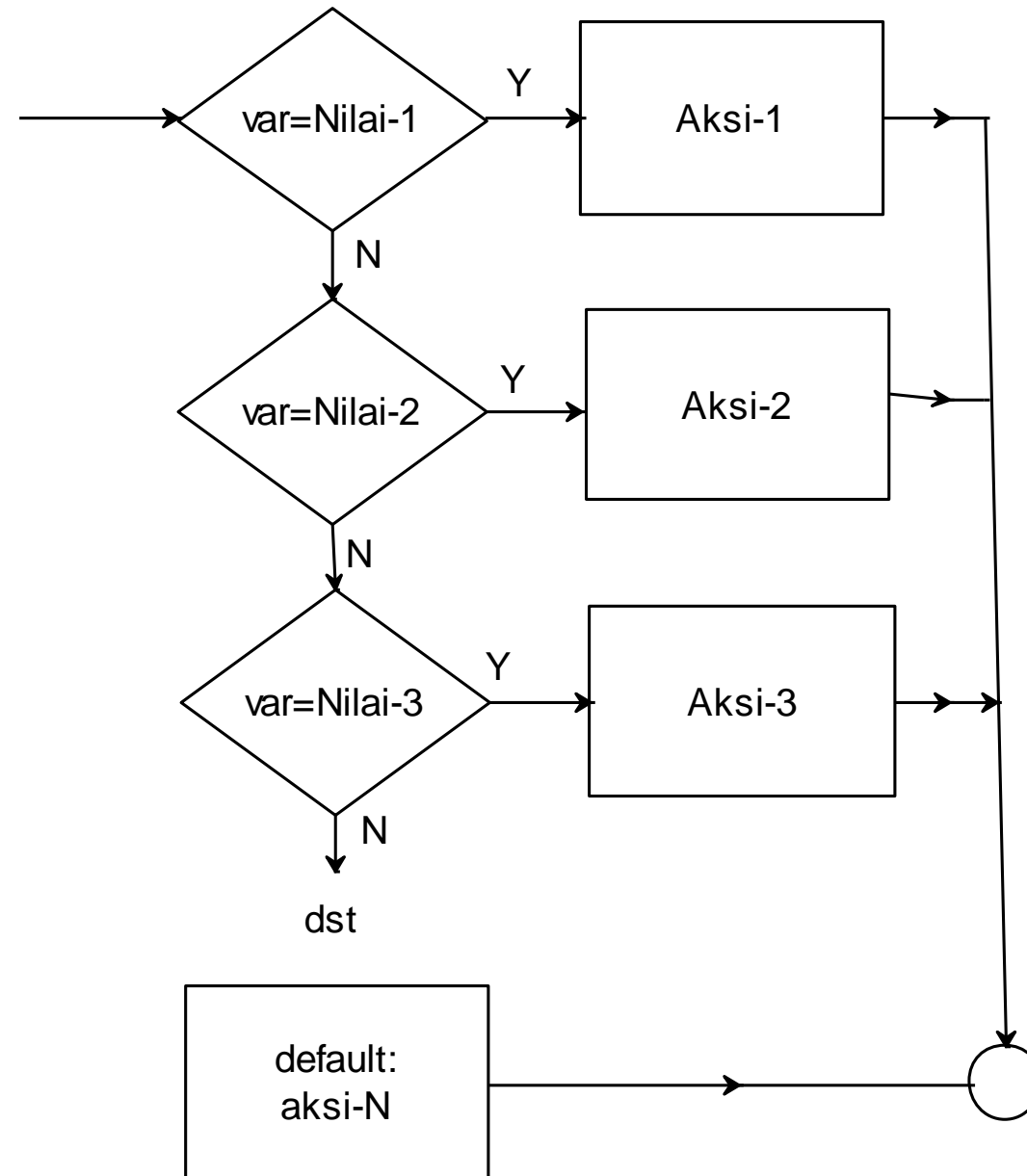
**default** : aksi-n;

**endcase.**



- Bentuk **case** diatas dapat diterjemahkan sebagai berikut:
  - dimungkinkan ada n-buah aksi, mulai dari aksi-1 hingga aksi-n
  - setiap aksi hanya dilakukan apabila suatu nilai variabel dicapai sesuai dengan persyaratan, misalnya: aksi-1 dilakukan hanya bila variabel bernilai nilai-1, aksi-2 dilaksanakan apabila variabel bernilai nilai-2, dan seterusnya.
  - apabila tidak ada satupun nilai variabel yang cocok maka aksi-n dikerjakan sebagai aksi "default" (hanya dikerjakan apabila tidak ada yang memenuhi syarat).

# Flowchart





# Contoh Soal

Gaji karyawan pada sebuah perusahaan di-dasarkan pada jam-kerja dalam satu bulan serta posisi atau golongannya dalam perusahaan itu. Upah perjam menurut golongan adalah sbb:

Golongan	Upah/jam (Rp)
<b>A</b>	<b>5000</b>
<b>B</b>	<b>6000</b>
<b>C</b>	<b>7500</b>
<b>D</b>	<b>9000</b>

Apabila karyawan bekerja lebih dari 150 jam perminggu, maka kelebihan jam kerja tersebut dihitung sebagai lembur dengan upah/jam 25% diatas upah reguler. Buat sebuah algoritma yang menerima nama, golongan, serta jam-kerja karyawan, kemudian menampilkan gaji total-nya dalam satu bulan.



# Algoritma

## Algoritma Gaji\_Karyawan

{ algoritma yang menerima nama, golongan serta jam-kerja kemudian menampilkan total gaji yang diterima karyawan }

### Definisi Variabel

**real** gaji, total, jamkerja, lembur, upah;  
**string** nama;  
**char** golongan;

### Rincian Langkah

```
write (" masukkan nama karyawan : ");  
read ( nama );  
write (" masukkan golongan-nya : ");  
read ( golongan );  
write (" masukkan jam kerjanya : ");  
read ( jamkerja );
```



**case ( golongan )**

**` A `** : upah  $\leftarrow$  5000;

**` B `** : upah  $\leftarrow$  6000;

**` C `** : upah  $\leftarrow$  7500;

**` D `** : upah  $\leftarrow$  9000;

**default : write ( " golongannya salah ! " );**

upah  $\leftarrow$  0;

**endcase.**

**if ( jamkerja > 150 )**

**then** lembur  $\leftarrow$  ( jamkerja - 150 ) \* upah \* 1.25;

gaji  $\leftarrow$  150 \* upah;

**else** lembur  $\leftarrow$  0;

gaji  $\leftarrow$  jamkerja \* upah;

**endif**

total  $\leftarrow$  gaji + lembur;

**write ( " Gaji yang diterima sdr : ", nama, " adalah = Rp. ",  
total);**