

BAB 7

STRUKTUR KONTROL (PERULANGAN)

Struktur perulangan

- Digunakan untuk melakukan sekelompok langkah/instruksi secara berulang-ulang. Sekelompok langkah/instruksi yang diulang ini disebut *loop body*.
- Untuk menentukan perlu tidaknya struktur loop, serta bentuk loop apa yang dipilih, perlu dijawab pertanyaan-pertanyaan berikut:
 1. Adakah langkah-langkah dalam program yang perlu dikerjakan berulang-ulang?
 2. Jika jawaban dari pertanyaan nomor 1 adalah ada, pertanyaan selanjutnya, apakah bisa diketahui sebelumnya berapa kali langkah-langkah tersebut harus dikerjakan?
 3. Jika jawaban dari pertanyaan nomor 2 adalah tidak, bagaimana kita bisa mengetahui berapa kali langkah-langkah tersebut harus dikerjakan?

Struktur perulangan

- Dalam setiap statemen perulangan, terdapat 3 struktur inti yakni:
 1. Inisialisasi kondisi, merupakan pemberian nilai awal pada kondisi. Apabila dalam statement perulangan tidak terdapat ini maka akan terjadi error pada saat pengecekan kondisi perulangan.
 2. Pengecekan kondisi merupakan pengecekan kondisi perulangan dan apabila bernilai benar maka pernyataan akan dijalankan, begitu pula sebaliknya.
 3. Step perulangan merupakan memberikan perubahan kepada nilai kondisi sehingga pada saat pengecekan kondisi, nilai dari kondisi akan bernilai salah. Apabila step perulangan tidak ada, maka perulangan tidak akan berhenti.

Struktur perulangan

- Ada 2 macam bentuk perulangan, yaitu :
 1. Perulangan Deterministik
Adalah bentuk perulangan dimana jumlah perulangan sudah dapat diketahui pasti.
Bentuk : **for**
 2. Perulangan Non Deterministik
Adalah bentuk perulangan dimana jumlah perulangan berdasarkan kondisi (belum dapat diketahui pasti).
Bentuk : **while** dan **do-while**

Struktur perulangan

KRITERIA	FOR	WHILE	DO-WHILE
Jumlah perulangan diketahui	Ya	Tidak	Tidak
Pengecekan kondisi	Ya	Ya, di awal	Ya, di akhir
Jumlah perulangan minimum	Sesuai jumlah perulangan	0	1

Struktur perulangan - Deterministik

Bentuk :

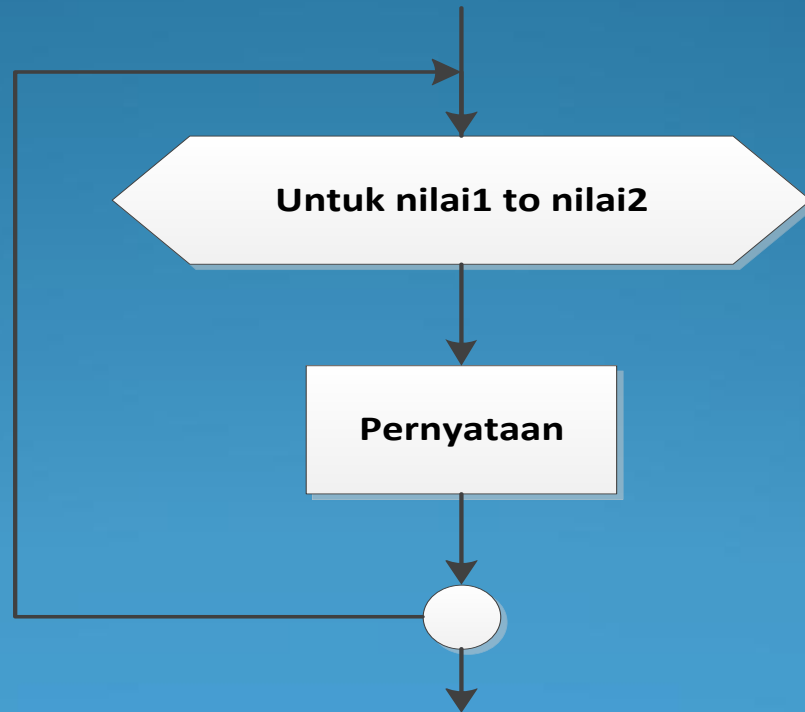
```
for (initialization_expression; loop_condition; step_expression)  
    pernyataan;
```

Ket:

- ~ initialization_expression = nilai awal , yaitu menginisialisasi variabel loop. Dieksekusi hanya sekali.
- ~ loop_condition = kondisi akhir perulangan, yaitu membandingkan variabel loop pada nilai batas tertentu.
- ~ step_expression = loop counter, yaitu melakukan update pada variabel loop (increment / penambahan atau decrement / pengurangan)

Struktur perulangan - Deterministik

Flowchart :



Struktur perulangan - Deterministik

- Contoh penggunaan for :

```
int i;  
for (i=0; i<=5; i++)  
    cout<<i;
```

Kondisi
awal

Kondisi
akhir

Increment/
decrement

- Pada contoh diatas, pernyataan `i=0` merupakan inisialisasi dari variabel loop. Selanjutnya, kondisi `i<=5` diperiksa. Jika kondisi bernilai true, maka pernyataan di dalam for (yaitu `cout<<i`) dieksekusi. Kemudian ekspresi `i++` dieksekusi, lalu kembali pada bagian pemeriksaan terhadap kondisi `i<=5` lagi. Kondisi ini akan dilakukan berulang-ulang sampai mencapai nilai yang salah (false).
- Sehingga output yang dihasilkan adalah : 0 1 2 3 4 5