



BAB VII

PERULANGAN FOR

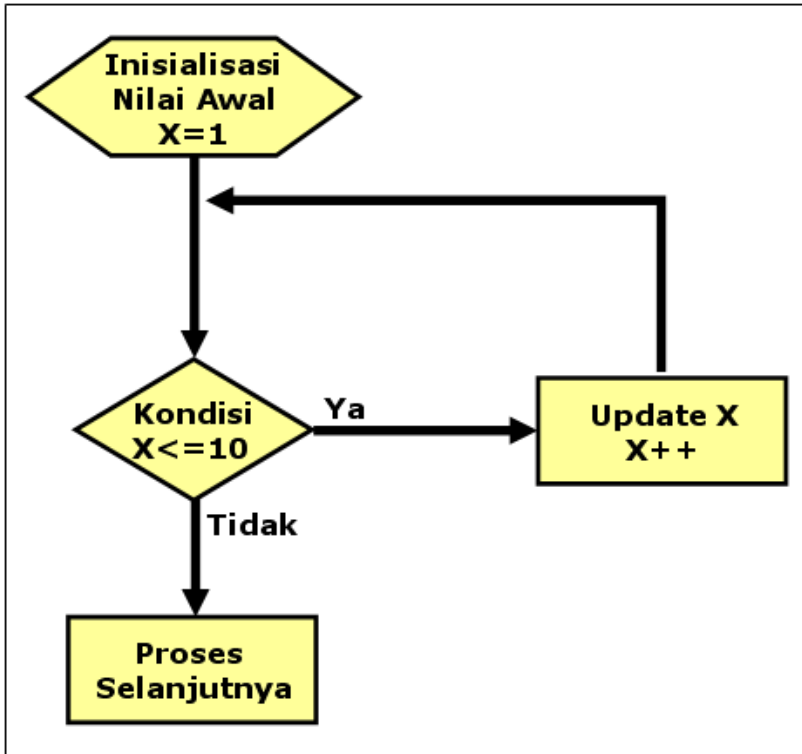
7.1 STRUKTUR PERULANGAN “FOR”

Struktur perulangan for biasa digunakan untuk mengulang suatu proses yang telah diketahui jumlah perulangannya. Dari segi penulisannya, struktur perulangan for tampaknya lebih efisien karena susunannya lebih simpel dan sederhana. Bentuk umum perulangan for adalah sebagai berikut :

```
for(inisialisasi; syarat; penambahan)  
{  
    pernyataan;  
}
```

Keterangan :

- **inisialisasi** : pernyataan untuk menyatakan keadaan awal dari variabel kontrol.
- **syarat** : ekspresi relasi yang menyatakan kondisi untuk keluar dari perulangan.
- **counter** : pengatur perubahan nilai variabel kontrol.



Gambar 7.1 Struktur Perulangan For

Contoh :

```

#include <stdio.h>
#include <conio.h>

main()
{
    int x;
    for(x=1; x<=10; x++)
    {
        printf("%d BAHASA C\n", x);
    }
    getch();
}
  
```

Program 7.1 Contoh Perulangan For

Pada contoh program 7.1 kita dapat mengamati bahwa inisialisasi: $x=1$, artinya kita memberi nilai awal mulai dari 1. Jika nilai x kita ganti menjadi $x=2$, maka kita akan memulai dari nilai 2.

Syarat: $x \leq 10$, artinya perulangan akan terus dilakukan jika kondisinya masih sesuai. Ketika nilai x tidak lagi kurang dari atau sama dengan 10, maka perulangan dihentikan.

Counter: $x++$, artinya x bertambah 1 angka setiap perulangan. Penulisan $x++$ dapat diganti dengan $x=x+1$. Jika kita menginginkan pertambahan nilai x adalah 2 angka setiap perulangan maka kita dapat menuliskannya $x=x+2$.

7.2 PERULANGAN FOR-NEGATIF

Secara umum, perulangan dibagi menjadi 2 jenis, yaitu; **1) Perulangan Positif** dan **2) Perulangan Negatif**. Perulangan Positif artinya setiap terjadi perulangan, maka nilai dari inisialisasinya selalu bertambah. Pada program 7.1 merupakan contoh perulangan positif.

Perulangan Negatif kebalikan dari perulangan positif. Perulangan negatif artinya setiap perulangan nilai dari inisialisasinya selalu berkurang.

Untuk membuat perulangan positif maka yang harus diingat adalah :

1. Nilai inisialisasinya harus lebih besar dibandingkan syaratnya.
2. counter setiap perulangan harus berkurang.

Contoh:

PERULANGAN-NEGATIF	KETERANGAN
<code>for (x=10; x>=1; x--)</code>	BENAR
<code>for (x=2; x>=-8; x=x-1)</code>	BENAR
<code>for (x=10; x<=1; x--)</code>	SALAH
<code>for (x=1; x>=10; x--)</code>	SALAH

Contoh :

```
#include <stdio.h>
#include <conio.h>

main()
{
    int x;

    clrscr();
    for(x=9; x>=1; x--)
    {
        printf("%d BAHASA C\n", x);
    }

    printf("-----\n");

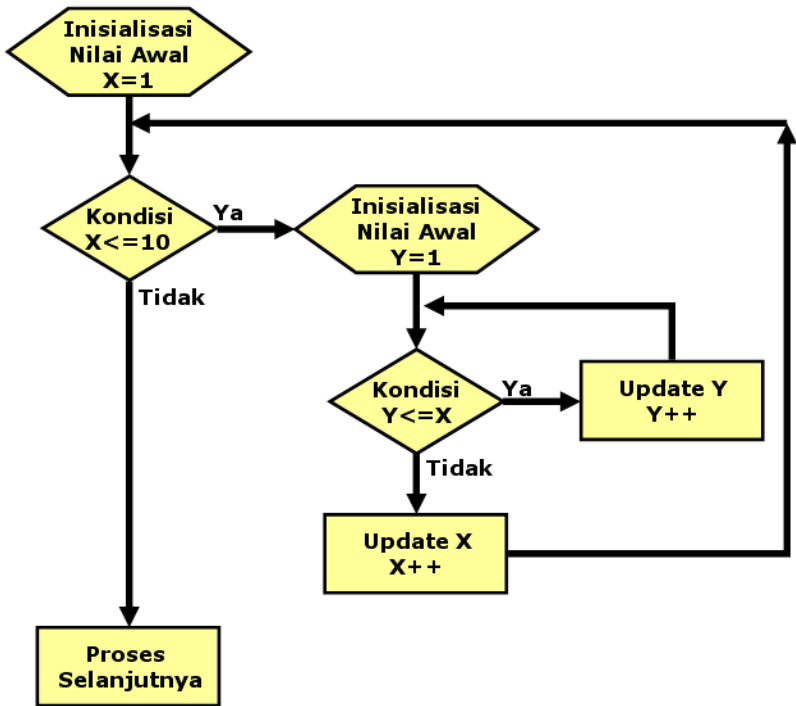
    for(x=2; x>=-16; x=x-2)
    {
        printf("%d BAHASA C\n", x);
    }

    getch();
}
```

Program 7.2 Contoh Perulangan For Negatif

7.3 PERULANGAN FOR TERSARANG

Perulangan for tersarang pun merupakan bentuk pengembangan, dimana anda akan menemukan perulangan didalam perulangan. Perulangan for tersarang sering digunakan dalam algoritma sorting atau pengurutan, namun dalam buku ini kita akan menggunakan contoh kasus yang mudah.



Gambar 7.2 Struktur Perulangan For Tersarang

Contoh :

```

#include <stdio.h>
#include <conio.h>

main()
{
    int x,y;

    clrscr();
    for(x=1; x<=10; x++)
    {
        for(y=1; y<=x; y++)
        {
            printf("%d", y);
        }
    }
}
  
```

6

```
    }  
    printf("\n");  
}  
  
getch();  
}
```

Program 7.3 Contoh Perulangan For Tersarang



L A T I H A N - 7

Lat 7.1 :

Apa yang terjadi jika program berikut ini di running ?

```
#include <stdio.h>
#include <conio.h>

main()
{
    int x;
    for(x=1; x>=10; x++)
    {
        printf("%i \n", x);
    }

    getch();
}
```

Lat 7.2 :

Apa penyebab terjadinya keanehan pada p(rogram Lat 7.1, jelaskan ?

Lat 7.3 :

Bualah program untuk menampilkan bilangan ganjil diatara 1 samapi dengan 100 !

8

Lat 7.3 :

Bualah program untuk menampilkan bilangan prima di antara 1 samapi dengan 100 !

Lat 7.4 :

Tuliskan Kode Program yang dapat menampilkan tampilan sebagai berikut, dengan perulangan for ?

```
#  
##  
###  
####  
#####  
#####  
#####
```

Lat 7.5:

Tuliskan Kode Program yang dapat menampilkan tampilan sebagai berikut, dengan perulangan for ?

```
  #  
  ##  
  ###  
  ####  
  #####  
  #####  
  #####  
 #####
```