

Pertemuan 5

Dasar Pemrograman Komputer

Struktur Perulangan :
WHILE, FOR dan DO WHILE



Tujuan

- ❖ Memberikan pemahaman mengenai struktur perulangan dan implementasinya dalam bahasa C menggunakan while, for dan do while, sehingga mahasiswa dapat memanfaatkan struktur perulangan tersebut dalam menyelesaikan persoalan pemrograman.



Struktur Perulangan

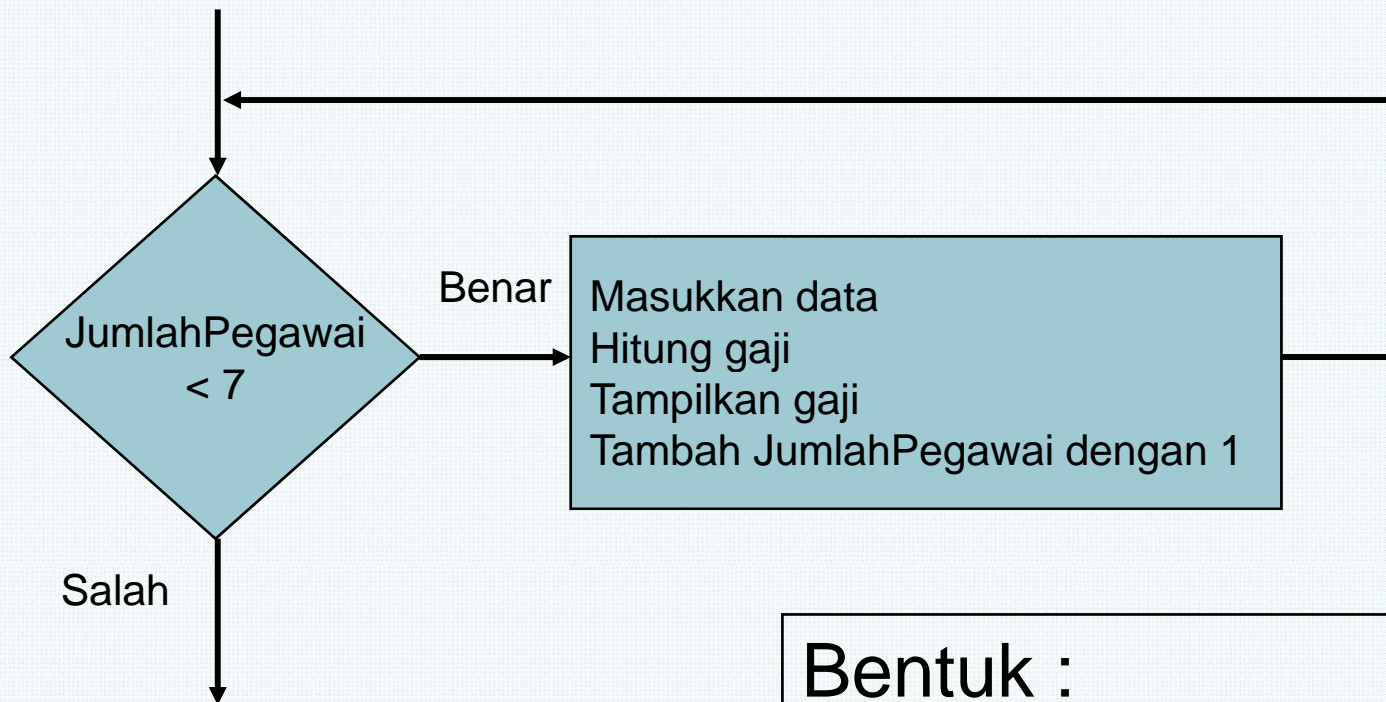
- ❖ Struktur Perulangan \Rightarrow struktur kontrol yang mengulang eksekusi sekumpulan langkah dalam program.
- ❖ Kumpulan langkah yang dikerjakan berulang-ulang disebut juga dengan *loop body*.



Struktur Perulangan

- ❖ 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?

Statement WHILE



Bentuk :

while (kondisi perulangan)
statement

Statement WHILE

Contoh :

```
JumlahPegawai=0; //initalisasi counter
while (JumlahPegawai < 7) {
    printf("Jumlah jam > "); scanf("%d", &jam);
    printf("Honor per jam > "); scanf("%lf", &honor);
    gaji = jam * honor;
    printf("Gaji adalah Rp. %6.2f \n", gaji);
    JumlahPegawai= JumlahPegawai +1; //
    penambahan cunter
}
```

A topographic map of a region, possibly in Southeast Asia, showing terrain, rivers, and some place names. The map is partially obscured by a blue gradient overlay at the top of the slide.

Statement WHILE

- ❖ Jika kondisi perulangan bernilai :
 - Benar : statement dalam “loop body” dijalankan
 - Salah : statement dalam “loop body” tidak dijalankan
- ❖ Variable Pengontrol Perulangan (pada contoh sebelumnya adalah JumlahPegawai) harus :
 - Diinisialisasi \Rightarrow contoh: JumlahPegawai=0;
 - Ditest \Rightarrow contoh: JumlahPegawai < 7
 - Diubah \Rightarrow contoh: JumlahPegawai += 1;



Statement FOR

Bentuk :

for (inisialisasi variabel perulangan;
test variabel perulangan;
ubah variabel perulangan)

Statement;

For (JumlahPegawai=0; JumlahPegawai<7; JumlahPegawai++)

S t a t e m e n t F O R

Contoh :

```
TotalGaji=0;
```

```
for (JumlahPegawai=0; JumlahPegawai < 7; JumlahPegawai += 1)
```

```
{
```

```
    printf("Jumlah jam > "); scanf("%d", &jam);
```

```
    printf("Honor per jam > "); scanf("%lf", &honor);
```

```
    gaji = jam * honor;
```

```
    printf("Gaji adalah Rp. %6.2f \n", gaji);
```

```
    JumlahPegawai= JumlahPegawai +1;
```

```
    TotalGaji=TotalGaji+ gaji;
```

```
}
```

```
printf("Semua pegawai telah diproses\n");
```

```
printf("Total gai adalah Rp %8.2f \n", TotalGaji);
```



Statement DO WHILE

- ❖ Satu bentuk perulangan di mana statement dalam “body loop” dieksekusi paling tidak satu kali atau lebih.

Bentuk :

```
do {  
    Statement;  
}  
while (kondisi perulangan);
```

A topographic map of a region, possibly in Southeast Asia, showing various geographical features like rivers, mountains, and cities. The map is overlaid with a semi-transparent blue banner at the top.

Statement DO WHILE

Contoh :

```
do {  
    printf("Masukkan huruf antara A sampai E > ");  
    scanf("%c", % HurufPilihan);  
} while (HurufPilihan < 'A' || HurufPilihan > 'E');
```



Latihan

1. Tuliskan program yang menghasilkan output sebagai berikut :

0	1
1	2
2	4
3	8
4	16
5	32
6	64



Latihan

2. Perhatikan potongan program berikut!
Tunjukkan nilai odd dan sum pada setiap iterasi, dengan nilai $n=8$!

```
sum=0;
for (odd=1; odd<n; odd += 2)
    sum=sum+odd;
printf("Jumlah dari bilangan ganjil positif yang kurang dari
%d adalah %d.\n", n, sum);
```



Latihan

3. Buatlah program yang menampilkan tabel perkalian untuk bilangan 0 sampai dengan 9!
4. Tuliskan perulangan yang menampilkan output berikut :

```
0
0 1
0 1 2
0 1 2 3
0 1 2
0 1
0
```



Referensi

- ❖ Bab 5, “Repetition and Loop Statements”, *Problem Solving and Program Design in C*, Jeri R. Hanly dan Elliot B. Koffman, Addison Wesley, 2002