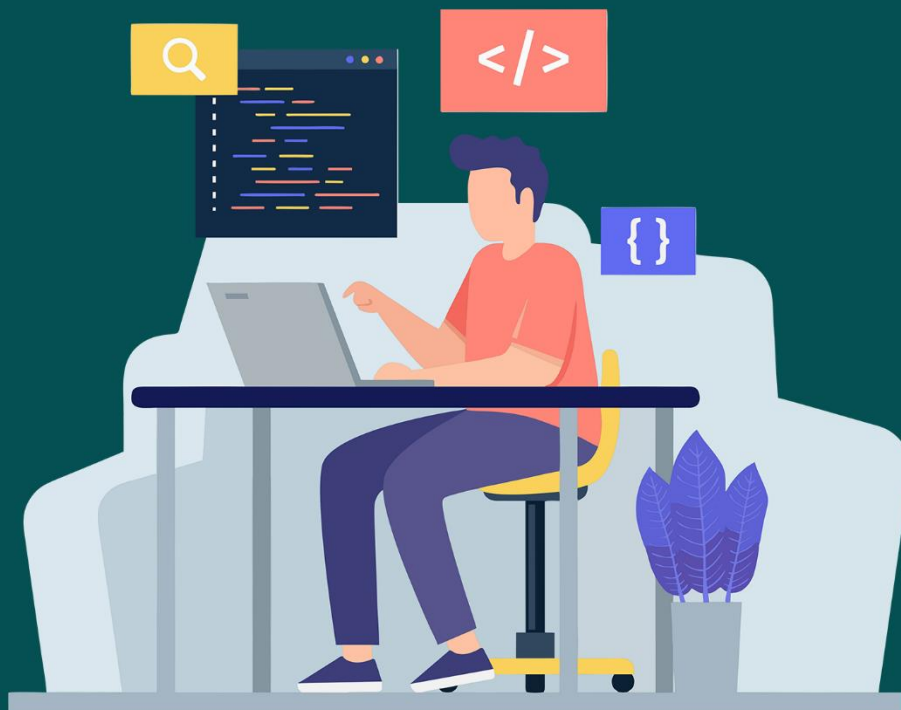


Modul Praktikum  
Algoritma dan Pemrograman I

## Perulangan For



TIM ASISTEN PEMROGRAMAN  
ANGKATAN 12

Departemen Pendidikan Ilmu Komputer  
Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam  
Universitas Pendidikan Indonesia  
2021

## Perulangan

**Perulangan** atau yang sering disebut dengan **Looping**, merupakan proses yang dilakukan secara berulang-ulang dalam batas yang telah ditentukan. Dalam hal ini kita sangat diuntungkan jika terdapat suatu **statement sama yang memiliki jumlah banyak** kita hanya perlu **mengetikkan satu statement** tersebut dan meletakkannya didalam struktur looping.

Secara umum, looping dibagi menjadi dua kelompok, yaitu

*counted loop* dan *uncounted loop*.

Perbedaanya:

- **Counted Loop** merupakan perulangan yang jelas dan sudah tentu banyak kali perulangannya.
- Sedangkan **Uncounted Loop**, merupakan perulangan yang tidak jelas berapa kali ia harus mengulang.

Dan ada 3 macam Looping di bahasa C, yaitu **for**, **do...while**, dan **while**.

## Perulangan For

Kali ini kita akan membahas **perulangan for**. **Perulangan for** merupakan perulangan yang termasuk dalam *counted loop*, karena sudah jelas berapa kali ia akan mengulang.

```
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
printf("Hello World\n");
```

```
int i;
for(i = 0 ; i < 10 ; i++){
    printf("Hello World\n");
}
```

### Output

```
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
```

## Struktur Perulangan For

```
for(/*Inisialisasi*/;/*Kondisi*/;/*Interasi*/){  
    /*  
        Block Program/Proses  
    */  
}
```

### 1. Inisialisasi

Inisialisasi adalah kondisi pada saat awal perulangan. Biasanya kondisi awal ini berisi perintah untuk memberikan nilai kepada **variable control**. Bagian ini hanya dieksekusi satu kali di awal perulangan.

```
int i, n;  
n = 10;  
for(i=n; /*Kondisi*/; /*Interasi*/){  
    /*  
        Block Program/Proses  
    */  
}  
  
for(i=0; /*Kondisi*/; /*Interasi*/){  
    /*  
        Block Program/Proses  
    */  
}
```

### 2. Kondisi

Bagian dimana kita mendeklarasikan kondisi yang harus terpenuhi agar perulangan dijalankan. Jika kondisi bernilai benar, maka Block Program/Proses akan dijalankan. Jika kondisi bernilai salah, maka Block Program/Proses tidak dijalankan dan proses perulangan for tersebut akan berhenti.

```
for(/*Inisialisasi*/; i < 10; /*Interasi*/){  
    /*  
        Block Program/Proses  
    */  
}  
  
n=10;  
for(/*Inisialisasi*/; i < n; /*Interasi*/){  
    /*  
        Block Program/Proses  
    */  
}
```

```
i < n;  
i > n;  
i <= n;  
i >= n;
```

### 3. Iterasi

Bagian ini dipakai untuk **mengupdate variable counter** agar bisa memenuhi kondisi akhir perulangan. Bagian ini selalu dieksekusi di setiap perulangan.

```
for(i=0; i < 10;i++){  
    /*  
        BLock Program/Proses  
    */  
}
```

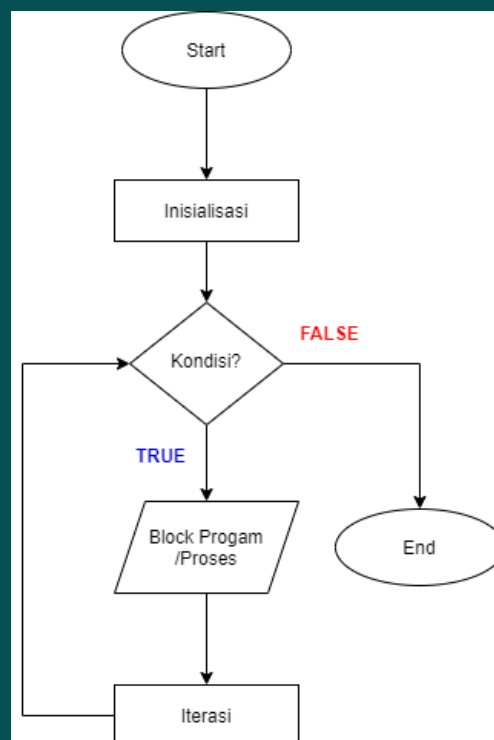
a. Menaikan Nilai  
(Increment)

```
//menaikan nilai 1  
i++  
i = i + 1  
//menaikan nilai 2  
i = i + 2  
i+= 2  
//menaikan nilai 3  
i+= 3
```

b. Menurunkan Nilai  
(Decrement)

```
//menurunkan nilai 1  
i--  
i = i - 1  
//menurunkan nilai 2  
i = i - 2  
i-= 2  
//menurunkan nilai 3  
i-= 3
```

### Bentuk Flowchart For



## Contoh Program

### 1. Menampilkan Angka Berurutan

Program:

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int i, n;
6      scanf("%d", &n);
7      for(i=1; i<=n; i++){
8          printf("Perulangan Ke-%d\n", i);
9      }
10
11     return 0;
12 }
13
```

Output:

```
5
Perulangan Ke-1
Perulangan Ke-2
Perulangan Ke-3
Perulangan Ke-4
Perulangan Ke-5
```

### 2. Menghitung Faktorial

Program:

```
#include <stdio.h>

int main()
{
    int i, n, hasil = 1;
    scanf("%d", &n);
    for(i=n; i>0; i--){
        hasil *= i;
    }
    printf("Hasil dari %d! adalah %d", n, hasil);

    return 0;
}
```

Output:

```
4
Hasil dari 4! adalah 24
```

3. Menghitung Jumlah Deret Bilangan dari 1 sampai n

Program:

```
#include <stdio.h>

int main()
{
    int i, n, total = 0;
    scanf("%d", &n);
    for(i=1; i<=n; i++){
        total += i;
    }
    printf("%d", total);

    return 0;
}
```

Output:

```
5
15
```

4. Menampilkan hasil Perkalian

Program:

```
#include <stdio.h>

int main()
{
    int i, n, num;
    printf("Masukkan Angka: ");
    scanf("%d", &num);
    printf("Mau dikalikan sampai berapa? ");
    scanf("%d", &n);
    for(i=1; i<=n; i++){
        printf("%d x %d = %d\n", num, i, num*i);
    }

    return 0;
}
```

Output:

```
Masukkan Angka: 5
Mau dikalikan sampai berapa? 10
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
```

#### 5. Membuat Pola

Program:

```
#include <stdio.h>

int main()
{
    int i;
    int n;
    scanf("%d", &n);
    for(i=0;i<n;i++){
        printf("*");
    }
    printf("\n");

    return 0;
}
```

Output:

```
5
*****
```

## 6. Pola Lain

Program:

```
#include <stdio.h>

int main()
{
    int i;
    int n;
    scanf("%d", &n);
    for(i=0;i<n;i++){
        printf("*\n");
    }
    return 0;
}
```

Output:

```
5
*
*
*
*
*
*
```



## Perulangan Bersarang

Program:

```
#include <stdio.h>

int main()
{
    int i, j;
    int n;
    scanf("%d", &n);
    for(i = 0; i < n; i++){
        for(j = 0; j < n; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

Output:

```
5
*****
*****
*****
*****
*****
```

## Pola Lain

Program:

```
#include <stdio.h>

int main()
{
    int i, j;
    int n;
    scanf("%d", &n);
    for(i = 0; i < n; i++){
        for(j = 0; j <= i; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

Output:

```
5
*
**
***
****
*****
```

## Latihan

Buatlah Pola – Pola dibawah ini menggunakan perulangan **For**

1.

```
5
*****
*****
*****
*****
*****
```

2.

```
5
      *****
      *****
      *****
      *****
      *****
```

3.

```
5
      *****
      *****
      *****
      *****
      *****
      *****
      *****
      *****
      *****
```

4.

```
5
10000
01000
00100
00010
00001
```



## Akhir Kata

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua elemen yang telah mendukung berjalannya sesi praktikum pada mata kuliah Algoritma dan Pemrograman I ini. Semoga apa yang telah kita laksanakan dapat menjadi berkah dan memberikan manfaat bagi kita semua dikemudian hari.

## Daftar Pustaka

Tim Asisten Praktikum Algoritma dan Pemrograman I Angkatan 11. (2020). *Perulangan For*. Modul Praktikum Algoritma dan Pemrograman I. Bandung: Universitas Pendidikan Indonesia.

Muhardian, Ahmad. (2019). **Belajar Pemrograman C #8: Memahami Blok Perulangan pada C**. <https://www.petanikode.com/c-perulangan/>