

Alif Catur Murti, M.Kom

UNIVERSITAS MURIAKUDUS
PROGRAM STUDI TEKNIK INFORMATIKA

2018

SEASON

Looping

----- C++ -----

Pengertian Perulangan atau Looping

Perulangan dapat diartikan sebagai berikut:

Perulangan (atau dalam bahasa Inggris disebut dengan loop) adalah instruksi program yang bertujuan untuk mengulang beberapa baris perintah. Fungsi paling sederhana yaitu untuk mempermudah melakukan suatu proses yang berulang-ulang, seperti mencetak angka dari 1 – 100. Ada 3 cara untuk melakukan perulangan di C++ yaitu: for, while, & do while.

A. 1 Perulangan dengan fungsi for

Penulisan dasar format perulangan for dalam C++ adalah sebagai berikut:

Source Code

>_

```
for (start; condition; increment)
{
    statement;
}
```

Start adalah kondisi pada saat awal perulangan.

Biasanya kondisi awal ini digunakan untuk membuat dan memberikan nilai kepada variabel yang digunakan untuk mengontrol perulangan. Misalkan, kita akan membuat variabel *i*, maka untuk kondisi start ini, kita juga harus memberikan nilai awal untuk variabel *i*, misalnya dengan 1, maka *i*=1. Misalkan kita ingin menghentikan perulangan jika variabel *i* telah mencapai nilai 10, maka pada bagian condition ini kita membuat perintah *i*≤10. Yang berarti selama nilai *i* kurang atau sama dengan 10, terus lakukan perulangan.

Increment adalah bagian yang digunakan untuk memproses variabel agar bisa memenuhi kondisi akhir perulangan.

Umumnya nilai variable tersebut bertambah (i++) / berkurang (i--) 1 (satu).

Condition adalah kondisi yang harus dipenuhi agar perulangan dijalankan. Selama kondisi ini terpenuhi, maka C++ akan terus melakukan perulangan.

Statement adalah bagian kode program yang akan diproses secara terus-menerus selama proses perulangan berlangsung. Kita membuat blok program di antara tanda kurung kurawal ({ dan }) sebagai penanda bahwa bagian di dalam kurung kurawal inilah yang akan dikenai proses perulangan.

Contoh looping dengan fungsi for pada program c++:

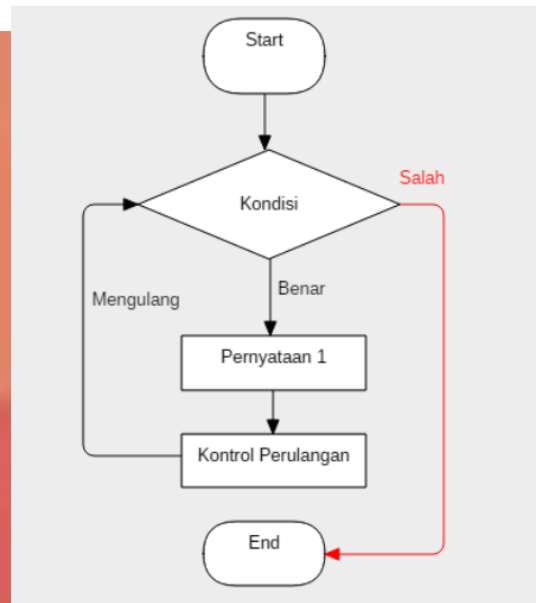
Source Code

```
#include <iostream>
using namespace std;

void CetakAngka (int i, int j){
    for (i = 1; i <= j; i++){
        cout << i << '\n';
    }
}

int main ()
{
    CetakAngka (1,10);
}
```

B. Perulangan dengan fungsi while-do



Dalam perulangan while, program akan terus melakukan perulangan dengan mengeksekusi pernyataan target selama kondisi tertentu bernilai benar.

Penulisan dasar format perulangan while dalam C++ adalah sebagai berikut:

Source Code

```
while(condition){  
    statement;  
}
```

Condition adalah kondisi yang harus dipenuhi agar perulangan berlangsung.

Kondisi ini mirip seperti dalam perulangan for. Condition ini akan diperiksa pada tiap perulangan, dan hanya jika hasilnya FALSE, maka proses perulangan berhenti. Artinya kita tidak tahu berapa banyaknya perulangan.

Karena, selama condition bernilai TRUE, maka perulangan akan terus dilakukan.

Statement adalah kode program yang akan diulang. Tanda kurung kurawal diperlukan untuk membatasi blok program yang akan diulang. Jika statement hanya terdiri dari 1 baris, maka tidak diperlukan.

Contoh perulangan dengan fungsi while dalam program c++:

Source Code

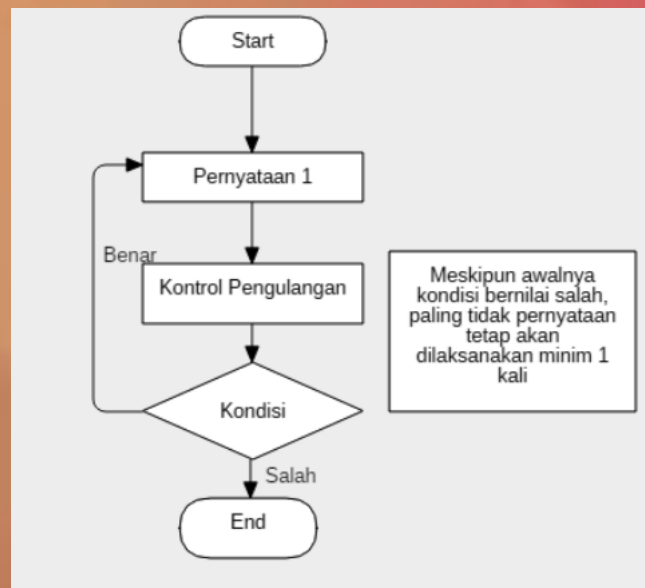
```
#include <iostream>
using namespace std;

int main () {
    // Local variable declaration:
    int a = 1;

    // while loop execution
    while( a < 10 ) {
        cout << "value of a: " << a << endl;
        a++;
    }

    return 0;
}
```

C. Perulangan dengan fungsi do while



Perulangan while dan do-while pada dasarnya hampir sama.

Source Code



```
do {  
    statement;  
} while (condition);
```

Perbedaan terletak pada 'lokasi' pengecekan kondisi perulangan.

Dalam struktur while, pengecekan untuk kondisi perulangan dilakukan di awal, sehingga jika kondisi tidak terpenuhi, maka perulangan tidak akan pernah dijalankan.

Namun pada perulangan do-while:

Pengecekan kondisi akan dilakukan di akhir perulangan, sehingga walaupun kondisi adalah FALSE, perulangan akan tetap berjalan minimal 1 kali. Statement adalah kode program yang akan diulang & condition adalah kondisi yang harus dipenuhi agar perulangan berlangsung.

Contoh program looping dengan fungsi do while dalam pemrograman c++:

Source Code



```
#include <iostream>  
using namespace std;  
  
int main () {  
    // Local variable declaration:  
    int a = 1;  
  
    // while loop execution  
    do {  
        cout << "value of a: " << a << endl;  
        a = a + 1;  
    } while ( a < 10 );  
  
    return 0;  
}
```

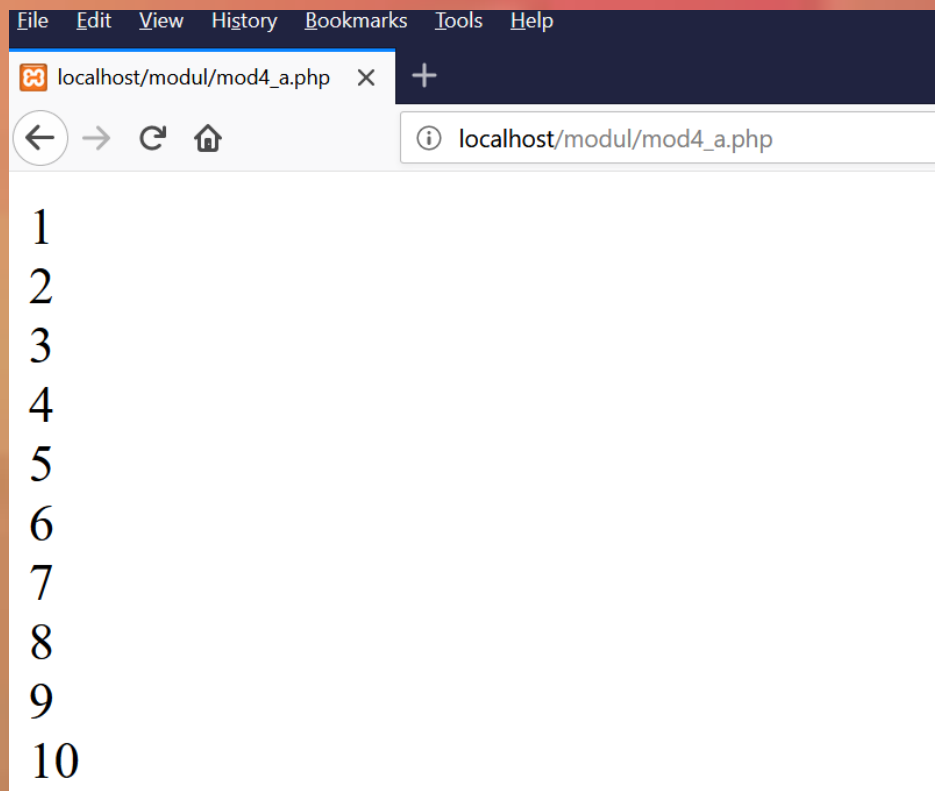
PHP

A. PERULANGAN FOR DALAM BAHASA PHP

Contoh :

```
mod4_a.php x
1 <?php
2 for ($i = 1; $i <= 10; $i++) {
3 echo $i."<br>";
4 }
5
6 ?>
7
```

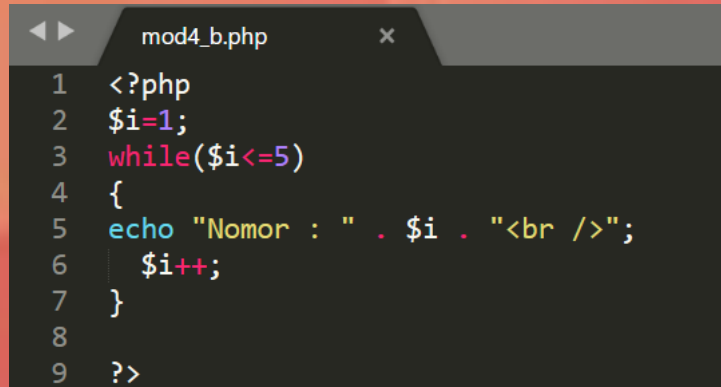
Menghasilkan :



1
2
3
4
5
6
7
8
9
10

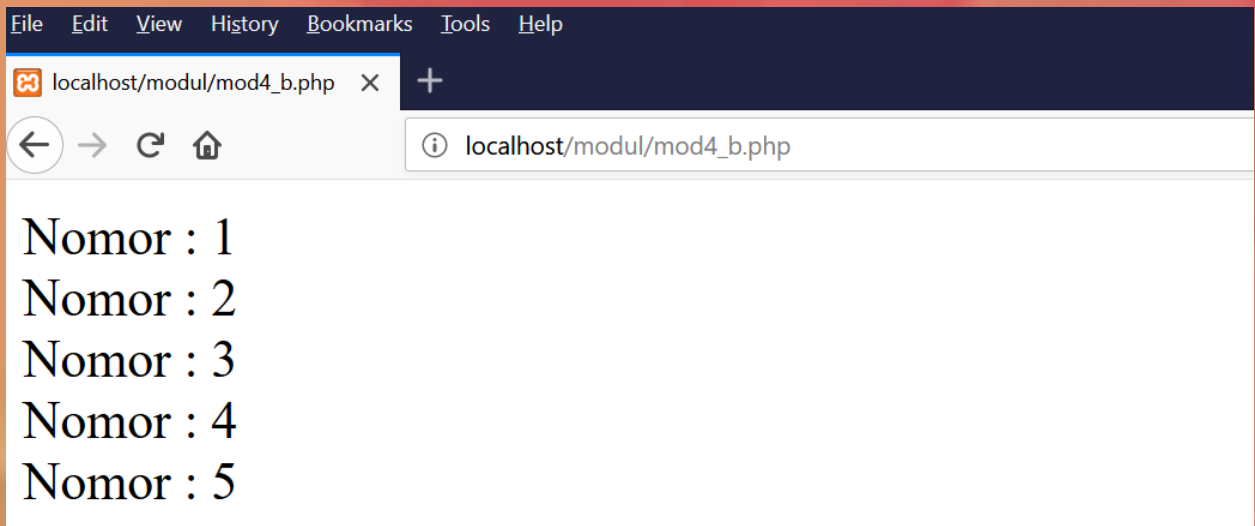
B. PERULANGAN WHILE-DO DALAM BAHASA PHP

Contoh :

A screenshot of a code editor window titled 'mod4_b.php'. The code is as follows:

```
1 <?php
2 $i=1;
3 while($i<=5)
4 {
5     echo "Nomor : " . $i . "<br />";
6     $i++;
7 }
8
9 ?>
```

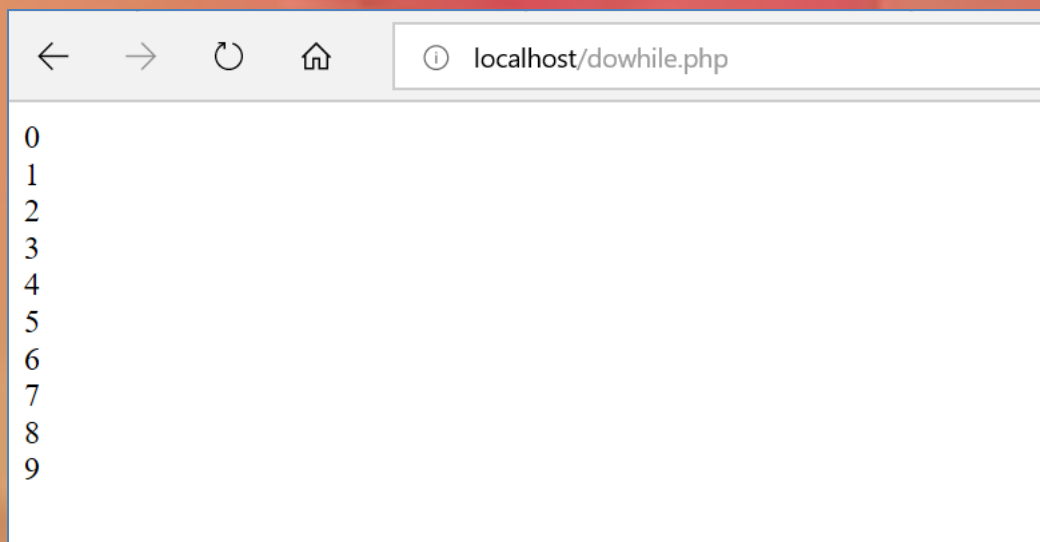
Menghasilkan :

A screenshot of a web browser window. The address bar shows 'localhost/modul/mod4_b.php'. The page content displays the output of the PHP script:

Nomor : 1
Nomor : 2
Nomor : 3
Nomor : 4
Nomor : 5

C. PERULANGAN DO-WHILE DALAM BAHASA PHP**Contoh:**

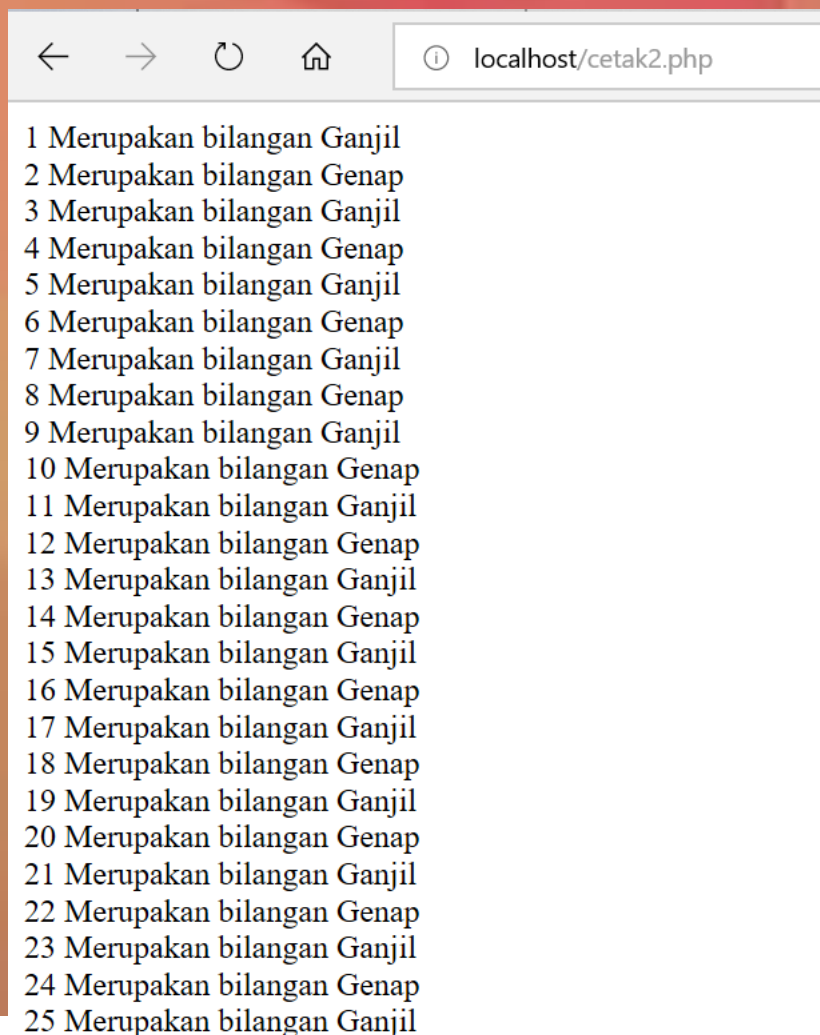
```
1  <?php
2  $i=0;
3
4  do {
5  echo $i ;
6  $i=$i+1;
7
8  } while ($i < 10);
9
10 ?>
11
```

Menghasilkan :

QUIS

**UNTUK 5 ORANG MAHASISWA YANG SELESAI
LEBIH CEPAT AKAN MENDAPATKAN POIN NILAI.**

**Buat program untuk menampilkan hasil seperti
berikut ini**



```
localhost/cetak2.php
1 Merupakan bilangan Ganjil
2 Merupakan bilangan Genap
3 Merupakan bilangan Ganjil
4 Merupakan bilangan Genap
5 Merupakan bilangan Ganjil
6 Merupakan bilangan Genap
7 Merupakan bilangan Ganjil
8 Merupakan bilangan Genap
9 Merupakan bilangan Ganjil
10 Merupakan bilangan Genap
11 Merupakan bilangan Ganjil
12 Merupakan bilangan Genap
13 Merupakan bilangan Ganjil
14 Merupakan bilangan Genap
15 Merupakan bilangan Ganjil
16 Merupakan bilangan Genap
17 Merupakan bilangan Ganjil
18 Merupakan bilangan Genap
19 Merupakan bilangan Ganjil
20 Merupakan bilangan Genap
21 Merupakan bilangan Ganjil
22 Merupakan bilangan Genap
23 Merupakan bilangan Ganjil
24 Merupakan bilangan Genap
25 Merupakan bilangan Ganjil
```