

PERULANGAN WHILE BAHASA C++

PENGERTIAN STRUKTUR PERULANGAN WHILE BAHASA C++

Dalam tutorial sebelumnya tentang [perulangan FOR](#), kita telah bahas bahwa sebuah perulangan harus memiliki 3 syarat: kondisi awal perulangan, kondisi pada saat perulangan, dan kondisi akhir perulangan.

Dalam perulangan **FOR**, ketiga syarat ini ditulis dalam 1 baris perintah seperti `for (i = 1; i < 5; i++)`. Di dalam perulangan **WHILE**, ketiga kondisi ini saling terpisah.

Berikut format dasar struktur perulangan **WHILE** dalam bahasa C++:

```
1 | start;  
2 | while (condition)  
3 | {  
4 |     // kode program  
5 |     // kode program  
6 |     increment;  
7 | }
```

Di bagian **start** biasanya ditulis perintah inisialisasi variabel counter, misalnya `i = 0`. Di bagian **condition** terdapat kondisi yang harus dipenuhi agar perulangan berjalan, misalnya `i < 5`. Kemudian perintah **increment** harus berada di dalam block perulangan agar bisa menaikkan nilai variabel counter, misalnya dengan perintah `i++`.

CONTOH KODE PROGRAM PERULANGAN WHILE BAHASA C++

Sebagai praktek pertama, berikut kode program perulangan WHILE untuk menampilkan teks "Hello World" sebanyak 5 kali:

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int i = 1;
8      while (i <= 5){
9          cout << "Hello World" << endl;
10         i++;
11     }
12
13     return 0;
14 }
```

Di baris 7 terdapat perintah untuk menginput angka 1 ke dalam variabel *i*. Nantinya, variabel *i* ini akan menjadi *variabel counter* yang dipakai untuk menentukan jumlah perulangan. Proses perulangan di mulai di baris 8. Perintah **while (i <= 5)** artinya, selama nilai variabel *i* kurang atau sama dengan 5, maka jalankan perulangan.

Di dalam blok perulangan terdapat perintah **cout << "Hello World" << endl** di baris 9. Ini dipakai untuk menampilkan teks "Hello World". Kemudian di baris 10 terdapat perintah *increment*, yakni **i++**. Perintah ini akan menaikkan nilai variabel *i* sebanyak 1 angka dalam setiap iterasi.

Perulangan while akan di ulang sebanyak 5 kali, mulai dari *i* = 1, *i* = 2, *i* = 3, *i* = 4, hingga *i* = 5. Ketika nilai *variabel counter* *i* sudah mencapai 6, maka kondisi **while (i <= 5)** tidak terpenuhi lagi (**false**), sehingga perulangan berhenti.

Sama seperti perulangan **FOR**, di dalam block perulangan **WHILE** kita juga bisa mengakses nilai dari variabel counter **i**:

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int i = 1;
8      while (i <= 5){
9          cout << "Hello World " << i <<endl;
10         i++;
11     }
12
13     return 0;
14 }
```

Bagaimana dengan perulangan menurun? tidak masalah. Kita tinggal mengatur kondisi awal, kondisi akhir, serta proses decrement:

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int i = 10;
8      while (i > 5){
9          cout << "Hello World " << i <<endl;
10         i--;
11     }
12
13     return 0;
14 }
```



Sebagai latihan, silahkan anda coba rancang kode program perulangan WHILE untuk membuat deret berikut:

3 6 9 12 15 18 21 24 27 30

Deret ini sudah pernah kita bahas dalam tutorial perulangan FOR, sekarang tinggal mengkonversinya menjadi perulangan WHILE.

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int i = 0;
8      while (i <= 10){
9          cout << i*3 <<endl;
10         i++;
11     }
12
13     return 0;
14 }
```