

Modul Praktikum
Algoritma dan Pemrograman I

While & Array Tipe Data Terstruktur



TIM ASISTEN PEMROGRAMAN
ANGKATAN 12

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

Pendahuluan

Programmer yang profesional harus mampu untuk menciptakan sebuah **program yang efektif dan efisien**. Salah satu yang dapat mewujudkan program demikian adalah dengan penggunaan perulangan for dan tipe data terstruktur yang sebelumnya telah dibahas.

Dalam beberapa kondisi **penggunaan for kurang tepat** misalnya perulangan tidak harus sampai selesai. Bahkan **tidak dapat digunakan** misalnya ketika jumlah iterasi untuk perulangan tidak tetap (tergantung pada kondisi yang bukan angka) sehingga diperlukan bentuk perulangan yang lain. Salah satu bentuk perulangan yang dapat **menggantikan perulangan for adalah perulangan while**.

Dalam pendeklarasian variabel dengan tipe data terstruktur timbul suatu permasalahan yaitu **bagaimana jika deklarasi variabel dengan tipe data terstrukturnya sangat banyak?** Permasalahan tersebut dapat dipecahkan dengan cara menggunakan **array tipe data terstruktur**. Maka dari itu, pada kali ini kita akan mempelajari penggunaan Perulangan While dan Array Tipe Data Terstruktur.

Topik yang Dipelajari

1. While
2. Array Tipe Data Terstruktur

While

While merupakan salah satu perulangan. While setidaknya memiliki 3 syarat:

- Inisialisasi = kondisi awal
- Kondisi = pernyataan yang akan di cek. **Dapat beberapa statement.**
- Iterasi = bagaimana suatu kondisi berjalan yaitu bertambah atau berkurang

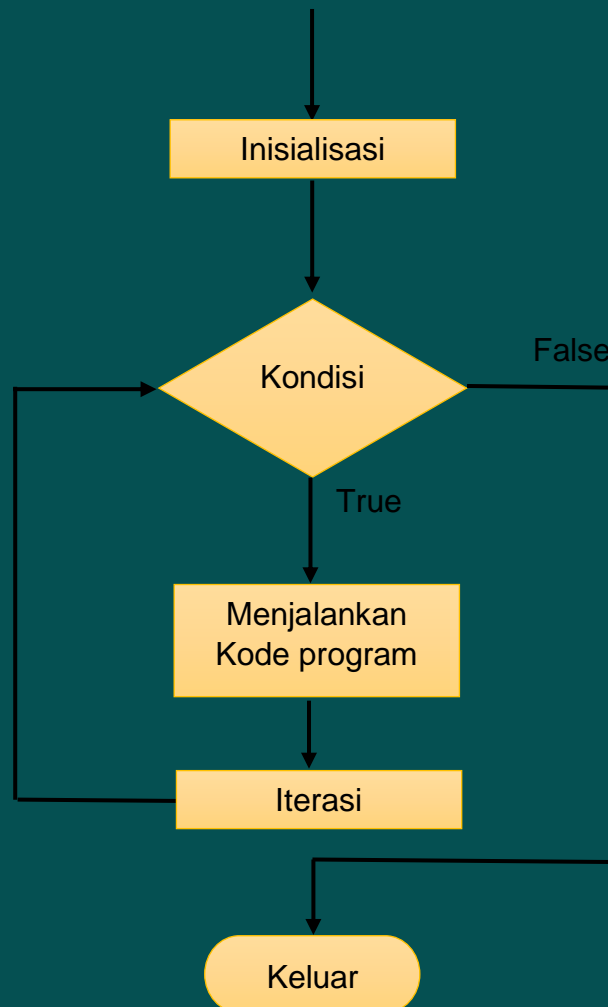
Format penulisan:

```
inisialisasi;
while(kondisi){
    //kode program
    //kode program
    iterasi;
}
```

While dapat digunakan untuk **jumlah perulangannya tidak tentu** dan dapat **berhenti di tengah-tengah perulangan** ketika syarat terpenuhi.

Cara Kerja While

Berikut skema cara kerja while:



Cara kerja while ini adalah berhenti ketika kondisi bernilai salah. Berikut penjelasan dari skema di atas:

Kondisi adalah variabel atau pernyataan yang akan dicek/menentukan apakah perulangan akan dilakukan atau dihentikan. Kondisi akan di cek setiap awal iterasi.

- Jika kondisi bernilai **true** maka kode program akan **dilakukan/dijalankan**
- Jika kondisi bernilai **false** maka perulangan **berhenti**.

Kondisi harus diperhatikan agar perulangan berhenti demi menghindari perulangan tak terbatas.

Contoh Program

1. While berhenti di tengah-tengah perulangan ketika syarat terpenuhi.
Mencari adakah bilangan kelipatan 3 atau tidak dari barisan aritmatika dengan $a = 4$ dan $b = 2$

Keluaran:

```
1  #include<stdio.h>
2
3  int main(){
4      int i, angka[5], kel3 = 0;;
5
6      //Misal nilai array adalah arimatika dengan a = 4 dan b = 2
7      for(i = 0; i < 5; i++){
8          angka[i] = 4 + (i)*2;
9      }
10
11     //pengecekan ada bilangan kelipatan 3 atau tidak
12     i = 0;
13     while ((i < 5) && (kel3 = 0)){ //kondisi
14         if((angka[i] % 2 == 1) ){
15             kel3 = 1;
16         }
17         i++; //iterasi
18     }
19
20     //keluaran
21     for(i=0; i<5; i++){
22         printf("%d\n", angka[i]);
23     }
24     if(kel3 = 1){
25         printf("Ada angka kelipatan 3\n");
26     }else{
27         printf("Tidak ada angka kelipatan 3\n");
28     }
29
30     return 0;
```

```
4
6
8
10
12
Ada angka kelipatan 3
```

2. While ketika jumlah perulangannya tidak tentu
Memasukkan huruf dan berhenti ketika masukan '*'

Keluaran:

```
1  #include<stdio.h>
2
3  int main(){
4      int i, n;
5      char huruf[100];
6
7      printf("Masukkan huruf\n");
8      printf("(jika selesai masukan '*')\n");
9
10     //inisialisasi
11     scanf(" %c", &huruf[0]);
12     n = 0; i = 0;
13     while ((huruf[i] != '*') && (i < 100)){ //kondisi
14         i++;
15         scanf(" %c", &huruf[i]);
16         n++; //iterasi
17     }
18
19     printf("Masukan sebelumnya:\n");
20     i = 0; //inisialisasi
21     while ((i < n)){ //kondisi
22         printf("%c\n", huruf[i]); //kode program
23         i++; //iterasi
24     }
25
26     return 0;
27 }
```

```
Masukkan huruf
(jika selesai masukan '*')
A
L
P
R
R
O
1
*
Masukan sebelumnya:
A
L
P
R
R
O
1
```

3. While ketika jumlah perulangannya tidak tentu

Melakukan pembelian barang dan menampilkan jumlah yang harus dibayar

```
1  #include<stdio.h>
2
3  int main(){
4
5      printf("=====\n");
6      printf("|          Toko ABC          |\n");
7      printf("=====\n");
8      printf("|          |\n");
9      printf("|      --Daftar Harga--      |\n");
10     printf("|          |\n");
11     printf("| (1) Buku       : Rp 4.000 |\n");
12     printf("| (2) Pensil    : Rp 1.000 |\n");
13     printf("| (3) Pulpen    : Rp 3.000 |\n");
14     printf("| (4) Lem       : Rp 5.000 |\n");
15     printf("| (5) Penghapus : Rp 1.000 |\n");
16     printf("|          |\n");
17     printf("=====\n");
18
19     int bayar = 0;
20     int beli;
21
22     printf("Barang yang akan dibeli (Masukan 0 jika selesai):\n");
23     scanf ("%d", &beli);
24
25     while(beli != 0){
26         if(beli == 1){
27             bayar += 4000;
28         }
29         else if(beli == 2){
30             bayar += 1000;
31         }
32         else if(beli == 3){
33             bayar += 3000;
34         }
35         else if(beli == 4){
36             bayar += 5000;
37         }
38         else if(beli == 5){
39             bayar += 1000;
40         }
41         scanf ("%d", &beli);
42     }
43
44     printf("Yang harus dibayar: Rp %d\n", bayar);
45
46     return 0;
47 }
```

Keluaran:

```
=====
|          Toko ABC          |
|          |
|      --Daftar Harga--      |
|          |
| (1) Buku       : Rp 4.000 |
| (2) Pensil    : Rp 1.000 |
| (3) Pulpen    : Rp 3.000 |
| (4) Lem       : Rp 5.000 |
| (5) Penghapus : Rp 1.000 |
|          |
=====
Barang yang akan dibeli (Masukan 0 jika selesai):
1
2
3
4
5
6
0
Yang harus dibayar: Rp 15000
```

Array Tipe Data Terstruktur

Array tipe data terstruktur hanyalah sebuah **tipe data terstruktur** yang dibuat menjadi **Array**. Ini merupakan solusi dari permasalahan jika variabel dengan tipe data terstruktur yang sama itu sangat banyak.

Misal akan membuat data mahasiswa yang terdiri dari no absen, nim, kelas dan usia secara manual.

```
1  #include<stdio.h>
2
3  //deklarasi bungkusan
4  typedef struct{
5      int nim, usia;
6      char kelas;
7  }data_mhs;
8
9  int main(){
10
11      int i;
12      data_mhs depilkom[2];
13
14      //data mahasiswa pertama
15      depilkom[0].nim = 20211;
16      depilkom[0].kelas = 'A';
17      depilkom[0].usia = 19;
18
19      //data mahasiswa kedua
20      depilkom[1].nim = 20212;
21      depilkom[1].kelas = 'B';
22      depilkom[1].usia = 20;
23
24      //Menampilkan depilkom
25      for(i = 0; i < 2; i++){
26          printf("No. Absen : %d\n", i+1);
27          printf("Nim      : %d\n", depilkom[i].nim);
28          printf("Kelas   : %c\n", depilkom[i].kelas);
29          printf("Usia     : %d\n\n", depilkom[i].usia);
30      }
31      return 0;
32  }
```

Keluaran:

```
No. Absen : 1
Nim       : 20211
Kelas    : A
Usia      : 19

No. Absen : 2
Nim       : 20212
Kelas    : B
Usia      : 20
```

Jika secara dinamis sesuai masukkan user:

```
1  #include<stdio.h>
2
3  //deklarasi bungkusan
4  typedef struct{
5      int nim;
6      char kelas;
7      int usia;
8  }data_mhs;
9
10 int main(){
11
12     int i, n;
13     scanf("%d", &n); //masukan user untuk jumlah data yang akan diinput
14
15     //deklarasi bungkusan data_mhs dengan nama depilkom sebanyak n
16     data_mhs depilkom[n];
17
18     //masukan user untuk data nim, kelas, dan usia
19     for(i = 0; i < n; i++){
20         scanf("%d %c %d", &depilkom[i].nim, &depilkom[i].kelas, &depilkom[i].usia);
21     }
22
23     //Menampilkan depilkom sesuai masukan user
24     for(i = 0; i < n; i++){
25         printf("No. Absen : %d\n", i+1);
26         printf("Nim      : %d\n", depilkom[i].nim);
27         printf("Kelas   : %c\n", depilkom[i].kelas);
28         printf("Usia      : %d\n\n", depilkom[i].usia);
29     }
30
31     return 0;
32 }
```

Latihan

1. Diberikan array bertipe character. Buat program untuk mencari apakah ada huruf A dalam array tersebut.

```
5
alpro
Ada huruf a, yang pertama di indeks ke-0
```

```
8
tipedata
Ada huruf a, yang pertama di indeks ke-5
```

```
5
while
Tidak ada huruf a
```

2. Diberikan masukan array tipe data terstruktur yang berisi nim, nilai dan kelas. Jika ada nilai yang 100 maka tampilkan “Ada orang Istimewa”, jika tidak ada tampilkan “Semua orang biasa”.

```
3
20211 A 80
20212 B 90
20213 C 100
Ada orang istimewa
```




Penutup

Terima kasih atas kerja sama kepada seluruh pihak yang telah mendukung dan membantu dalam penyusunan modul ini. Semoga apa yang telah didapatkan dapat menjadi berkah dan memberikan manfaat di kemudian hari.

Daftar Pustaka

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