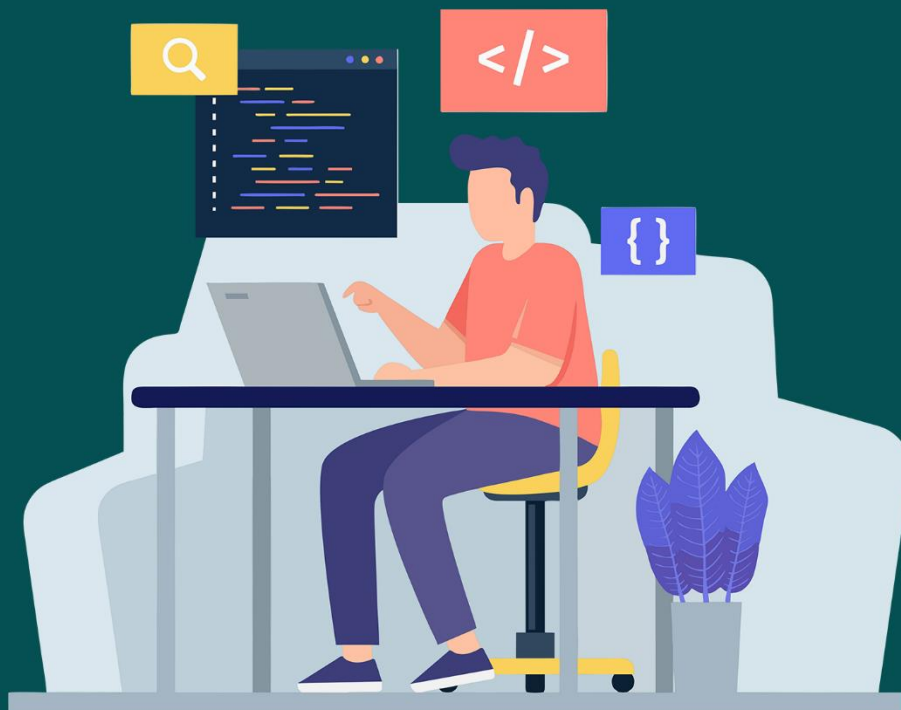


Modul Praktikum  
Algoritma dan Pemrograman I

## Pemilihan Kondisi (If) / Percabangan



TIM ASISTEN PEMROGRAMAN  
ANGKATAN 12

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



## Ucapan Terimakasih

Penulis mengucapkan terimakasih atas bantuan dari berbagai pihak baik secara langsung maupun tidak langsung dalam proses

## Pendahuluan

Dalam pembuatan suatu program, ada kalanya kita membutuhkan suatu **percabangan**. Kita dapat menganggap **percabangan** seperti sebuah **gerbang**, yang apabila suatu kondisi terpenuhi maka program akan menjalankan **kode program / perintah** yang ada di dalam gerbang tersebut. Dan jika tidak, maka gerbang akan tetap tertutup dan program akan menjalankan kode program / perintah yang lain. Dalam bahasa C, konsep ini dibuat dengan struktur **IF**.

## Apa Yang Akan Dipelajari?

Materi yang akan dipelajari adalah sebagai berikut :

1. Operator Pembandingan dan Operator Logika
2. Pemilihan 1 Kondisi
3. Pemilihan 2 Kondisi
4. Pemilihan Lebih dari Dua Kondisi
5. Pemilihan Beruntun
6. Pemilihan Bersarang

## Operator Pembandingan dan Operator Logika

### A. Operator Pembandingan

Merupakan operator yang digunakan untuk membandingkan 2 variabel.

Simbol	Arti Simbol
$a == b$	a sama dengan b
$a != b$	a tidak sama dengan b
$a < b$	a lebih kecil dari b
$a > b$	a lebih besar dari b
$a <= b$	a lebih kecil atau sama dengan b
$a >= b$	a lebih besar atau sama dengan b

### B. Operator Logika

#### 1) Operator AND (“&&”)

Kondisi yang dihubungkan operator AND akan bernilai TRUE jika semua syarat bernilai TRUE

a	b	a && b
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
FALSE	FALSE	FALSE

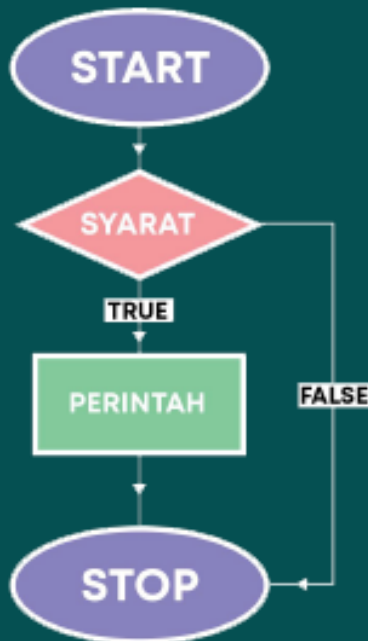
#### 2) Operator OR (“||”)

Kondisi yang dihubungkan operator OR akan bernilai TRUE jika salah satu syarat bernilai TRUE

a	b	a    b
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE

## Pemilihan Satu Kondisi

Cara Penulisan :



```
if(/*Kondisi atau syarat*/)
{
    /*Perintah yang akan program kerjakan
    apabila syarat terpenuhi*/
}
```

Ketika **syarat terpenuhi**, maka perintah yang berada **di dalam bracket** (“{””) **akan dijalankan** oleh program. Namun apabila syarat tidak terpenuhi, maka perintah tersebut tidak akan dijalankan.

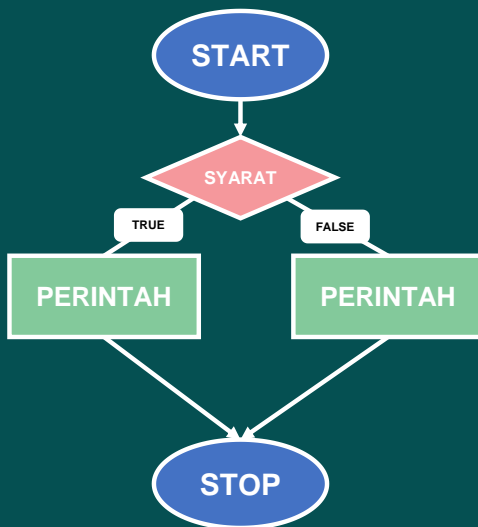
Contoh Program :

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int stones;
6
7      printf("\nMasukkan jumlah infinity stones : ");
8      scanf("%d", &stones);
9
10     if(stones >= 6)
11     {
12         printf("THANOS WINS\n");
13     }
14
15     return 0;
16 }
17
```

Output :

```
Masukkan jumlah infinity stones : 6
THANOS WINS
```

## Pemilihan Dua Kondisi



### Cara Penulisan :

```
if(/*Kondisi atau syarat*/)
{
    /*Perintah yang akan program kerjakan
    apabila syarat terpenuhi*/
}
else
{
    /*Perintah yang akan program kerjakan
    apabila syarat tidak terpenuhi*/
}
```

Ketika **syarat terpenuhi**, maka perintah yang **berada di dalam "if"** akan dijalankan. Jika tidak, maka program akan menjalankan perintah yang **berada di dalam "else"**.

### Contoh Program :

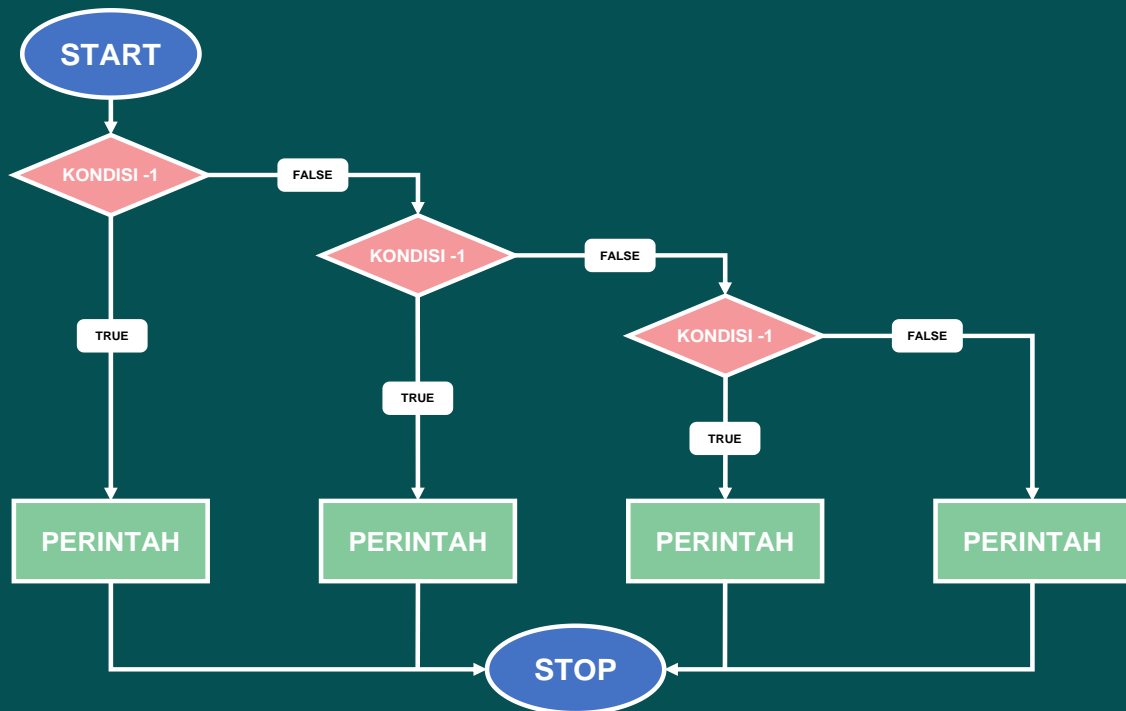
```
1  #include <stdio.h>
2
3  int main()
4  {
5      int stones;
6
7      printf("\nMasukkan jumlah infinity stones : ");
8      scanf("%d", &stones);
9
10     if(stones >= 6)
11     {
12         printf("THANOS WINS\n");
13     }
14     else
15     {
16         printf("AVENGERS WINS\n");
17     }
18
19     return 0;
20 }
21
```

### Output :

```
Masukkan jumlah infinity stones : 6
THANOS WINS
```

```
Masukkan jumlah infinity stones : 2
AVENGERS WINS
```

## Pemilihan Lebih dari Dua Kondisi



Cara Penulisan :

```
if(/*Kondisi atau syarat 1*/)
{
    /*Perintah yang akan program kerjakan
    apabila syarat terpenuhi*/
}
else if(/*Kondisi atau syarat 2*/)
{
    /*Perintah yang akan program kerjakan
    apabila syarat terpenuhi*/
}
else if(/*Kondisi atau syarat 3*/)
{
    /*Perintah yang akan program kerjakan
    apabila syarat terpenuhi*/
}
else
{
    /*Perintah yang akan program kerjakan
    apabila syarat tidak terpenuhi*/
}
```

Program di atas akan mengecek kondisi atau syarat **sesuai urutan penempatannya**. Apabila salah satu syarat terpenuhi maka program akan menjalankan perintah yang sudah ditulis dan menghentikan proses pengecekan.

### Contoh Program :

```
1  #include <stdio.h>
2
3  int main()
4  {
5
6      int angka;
7
8      printf("Masukkan angka : \n");
9      scanf("%d", &angka);
10
11     if(angka < 5)
12     {
13         printf("Angka kurang dari 5!\n");
14     }
15     else if(angka < 10)
16     {
17         printf("Angka kurang dari 10!\n");
18     }
19     else
20     {
21         printf("Angka lebih dari 10!\n");
22     }
23
24
25     return 0;
26 }
27
```

### Output :

```
Masukkan angka : 5
Angka kurang dari 10!
```

```
Masukkan angka : 8
Angka kurang dari 10!
```

```
Masukkan angka : 11
Angka lebih dari 10!
```

## Pemilihan Beruntun

Cara Penulisan :

```
if( /*Kondisi atau syarat*/ )
{
    /*Perintah yang akan dijalankan program apabila
    syarat terpenuhi */
}
if( /*Kondisi atau syarat*/ )
{
    /*Perintah yang akan dijalankan program apabila
    syarat terpenuhi */
}
if( /*Kondisi atau syarat*/ )
{
    /*Perintah yang akan dijalankan program apabila
    syarat terpenuhi */
}
```

Pada penulisan di atas, program akan tetap **mengecek semua kondisi** pada **semua** “if statement” satu persatu dan menjalankan perintah di dalam bracket apabila syarat terpenuhi.

Contoh Program :

```
1  #include <stdio.h>
2
3  int main(){
4      // Deklarasi variabel penampung bilangan
5      int bilangan;
6
7      // Meminta user memasukan sebuah bilangan
8      printf("Masukan bilangan : ");
9      scanf("%d", &bilangan);
10
11     // Syarat Pertama
12     if(bilangan > 10){
13         printf("Bilangan lebih besar dari 10\n");
14     }
15     // Syarat Kedua
16     if(bilangan % 5 == 0){
17         printf("Bilangan merupakan kelipatan 5\n");
18     }
19     // Syarat Ketiga
20     if(bilangan % 2 == 0){
21         printf("Bilangan merupakan bilangan genap\n");
22     }
23     return 0;
24 }
```

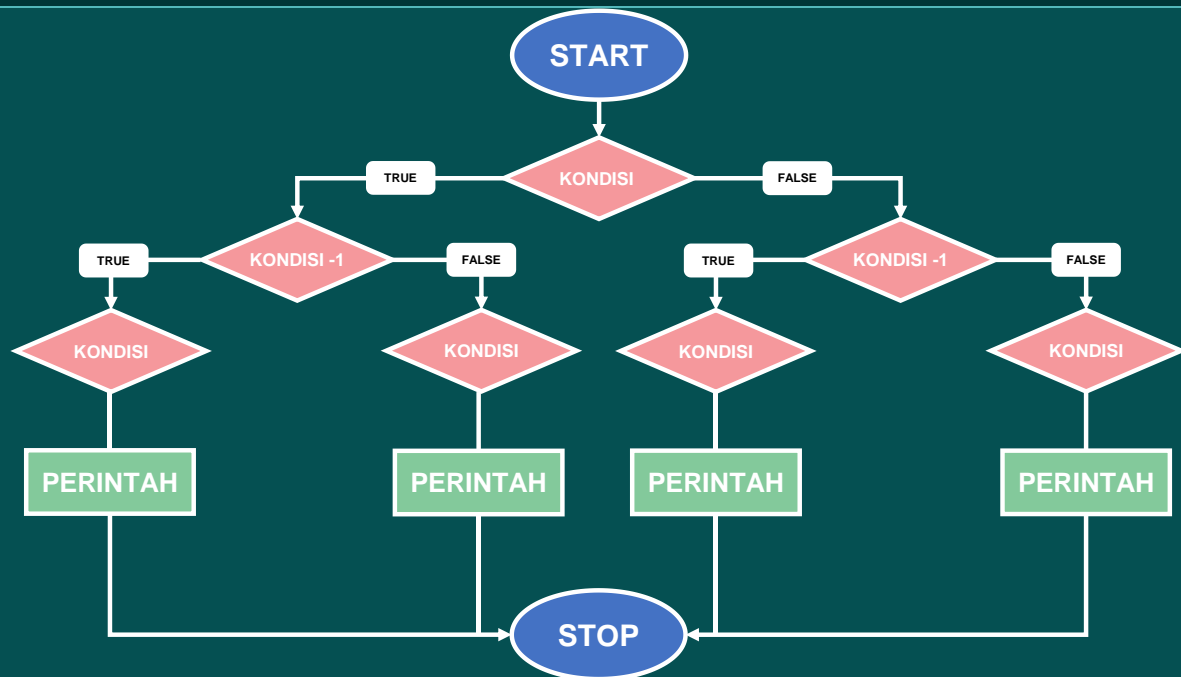


Output :

```
Masukan bilangan : 50
Bilangan lebih besar dari 10
Bilangan merupakan kelipatan 5
Bilangan merupakan bilangan genap
```

```
Masukan bilangan : 4
Bilangan merupakan bilangan genap
```

### Pemilihan Bersarang



Cara Penulisan :

```
if( /* Syarat 1 */ ){
    /* Perintah yg akan dijalankan jika syarat 1 terpenuhi */
    if( /* Syarat 2 */ ){
        /* Perintah yg akan dijalankan jika syarat 2 terpenuhi */
    }
    else{
        /* Perintah yg akan dijalankan jika syarat 2 TIDAK terpenuhi */
    }
}
else{
    /* Perintah yg akan dijalankan jika syarat 1 TIDAK terpenuhi */
    if( /* Syarat 3 */ ){
        /* Perintah yg akan dijalankan jika syarat 3 terpenuhi */
    }
    else{
        /* Perintah yg akan dijalankan jika syarat 3 TIDAK terpenuhi */
    }
}
```

### Contoh Program :

```
1  #include <stdio.h>
2
3  int main(){
4      // Deklarasi variabel penampung bilangan
5      int bilangan;
6
7      // Meminta user memasukan sebuah bilangan
8      printf("Masukan bilangan : ");
9      scanf("%d", &bilangan);
10
11     // Syarat Pertama
12     if(bilangan >= 20){
13         // Jika syarat pertama terpenuhi
14         if(bilangan <= 50){
15             printf("Merupakan bilangan kurang dari 50\n");
16         }
17         else{
18             printf("Merupakan bilangan lebih dari 50");
19         }
20     }
21     else{
22         // Jika syarat pertama TIDAK terpenuhi
23         if(bilangan > 10){
24             printf("Merupakan bilangan belasan\n");
25         }
26         else{
27             printf("Merupakan bilangan satuan\n");
28         }
29     }
30
31
32     return 0;
33 }
```

### Output :

```
Masukan bilangan : 68
Merupakan bilangan lebih dari 50
Masukan bilangan : 3
Merupakan bilangan satuan
Masukan bilangan : 14
Merupakan bilangan belasan
```



## Latihan

1. Peter mendapat kiriman 3 buah kotak penuh apel dari Happy, dia ingin mengetahui kotak mana yang memiliki apel **terbanyak** dan apakah jumlah apel itu **genap atau ganjil**. Buatlah program untuk membantu Peter!

Contoh input :

Kotak A : 23

Kotak B : 33

Kotak C : 9

Contoh Output :

Apel terbanyak terdapat pada kotak B dan jumlahnya ganjil



## Akhir Kata

*“Experience is the name everyone gives to their own mistakes.”*

*-Oscar Wilde*

## 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.