

# Konsep Pemrograman

## 3. Pengambilan Keputusan 1

Umi Sa'adah

Entin Martiana Kusumaningtyas

Tri Hadiah Muliawati

2021



Politeknik Elektronika Negeri Surabaya  
Departemen Teknik Informatika dan Komputer

# Overview

- Kondisi & Operator Kondisi
  - Operator Relasi
  - Operator Logika
  - Prioritas Operator Relasi & Logika
- Pernyataan `if`
- Pernyataan `if else`

# Kondisi

- Banyak persoalan yang akan diselesaikan dengan komputer memerlukan sebuah pengambilan keputusan, misalnya menyimpulkan bahwa suatu bilangan adalah genap atau ganjil.
- C menyediakan beberapa jenis pernyataan, berupa
  - Pernyataan *if*
  - Pernyataan *if-else*
  - Pernyataan *switch*
- Pernyataan-pernyataan tersebut memerlukan suatu **KONDISI**, sebagai basis dalam pengambilan keputusan.




# Kondisi & Operator Kondisi

- Pernyataan **kondisi** adalah pernyataan yang bernilai TRUE (benar) atau FALSE (salah)
- Dibentuk oleh operator kondisi, yaitu operator relasi atau digabung dengan operator logika
- Pada C nilai hasil dari sebuah pernyataan kondisi jika dinyatakan dengan angka adalah:
  - FALSE → nilai = 0
  - TRUE → nilai != 0 (misalnya nilai = 1)

# Operator Relasi

- Operator relasi digunakan untuk membandingkan dua buah nilai.
- Hasil pembandingan berupa keadaan TRUE atau FALSE
- Tabel operator relasi

Operator	Makna
>	Lebih dari
>=	Lebih dari atau sama dengan
<	Kurang dari
<=	Kurang dari atau sama dengan
==	Sama dengan
!=	Tidak sama dengan

- 
- Operator relasi sama dengan (==) harap dibedakan dengan operator (=) yang merupakan operator penugasan (assignment).

# Operator Relasi

- Beberapa contoh pemakaian operator relasi

<u>Kondisi</u>	<u>Hasil</u>
----------------	--------------

- |                  |       |  |
|------------------|-------|--|
| • $1 > 2$        | FALSE | → dibaca : apakah 1 lebih dari 2 ?   |
| • $1 < 2$        | TRUE  | → apakah 1 kurang dari 2 ?   |
| • $A == 1$       |       | → apakah $A = 1$ ?<br>TRUE, jika A bernilai 1; FALSE, jika A tidak bernilai 1  |
| • $'A' < 'B'$    |       | → apakah 'A' kurang dari 'B' ?<br>TRUE, karena kode ASCII untuk karakter 'A' kurang dari kode ASCII untuk karakter 'B' |
| • $jawab == 'Y'$ |       | → apakah jawab == 'Y' ?<br>TRUE, jika kar berisi 'Y'; FALSE, jika kar tidak berisi 'Y'                                 |

# Operator Logika

- Operator logika dipakai untuk menghubungkan ekspresi relasi
- Tabel Operator Logika

Operator	Makna
& &     !	dan (AND) atau (OR) tidak (NOT)

- Bentuk pemakaian operator && dan || adalah  
**operand1 operator operand2**
- Baik **operand1** maupun **operand2** dapat berupa ekspresi relasi ataupun ekspresi logika.
- Hasil pernyataannya bisa bernilai TRUE atau FALSE.

# Operator Logika

- Tabel di bawah ini memberikan penjelasan hasil operasi ekspresi logika yang menggunakan operator && maupun || untuk berbagai kemungkinan keadaan operand-nya.

Operand1	Operand2	Hasil	
			&&
FALSE	FALSE	0	0
FALSE	TRUE	1	0
TRUE	FALSE	1	0
TRUE	TRUE	1	1

- Tampak bahwa operator **atau** (||) menghasilkan nilai 1 jika ada operand yang TRUE dan nilai 0 jika semua operand adalah salah.
- Adapun operator logika **dan** (&&) memberikan hasil 1 hanya jika kedua operand adalah benar.



# Operator Logika

Contoh ekspresi logika di antaranya :

- `(kar >= 'A') && (kar <= 'Z')`

Hasil operasi logika `&&` adalah TRUE hanya jika `kar > 'A'` dan `kar < 'Z'` (dalam hal ini yang diperbandingkan adalah kode ASCII dari karakter tsb).

- `(jawab == 'Y') || (jawab == 'y')`

Hasil operasi logika `||` adalah TRUE jika `jawab` berupa 'Y' atau 'y'

Catatan Kode ASCII:

- `0 → 48, 1 → 49 s/d 9 → 57`
- `A → 65, Z → 90, a → 97, z → 122`



# Prioritas Operator Logika dan Relasi

- Tabel prioritas operator logika dan relasi

<b>Tertinggi :</b>	!			
	>	>=	<	<=
	=	!=		
	&&			
<b>Terendah:</b>				

- Berdasarkan prioritas yang ditunjukkan pada tabel di atas, maka ekspresi seperti

`(kar > 'A') && (kar < 'Z')`

sama saja kalau ditulis menjadi

`kar > 'A' && kar < 'Z'`

Hanya saja penulisan dengan menggunakan tanda kurung akan lebih memberikan kejelasan.

# Pernyataan if

- Pernyataan if mempunyai bentuk umum :

```
if (kondisi )  
    pernyataan;
```

- **Kondisi** diuji dulu, nilainya TRUE ataukah FALSE
- Jika nilainya TRUE, maka pernyataan yang mengikutinya akan dieksekusi
- Jika nilai FALSE, maka pernyataan yang mengikutinya akan diabaikan

# Contoh

```
#include <stdio.h>
```

```
main(){  
    float total, diskon = 0.0f;  
  
    printf("Total belanja : ");  
    scanf("%f", &total);  
    if(total >= 200000)  
    {  
        printf("Selamat anda dapat diskon 10%%");  
        diskon = total * 0.1;        //disc 10%  
    }  
    printf("Total yang harus dibayar : %8.2f\n",  
        total - diskon);  
}
```



# Pernyataan *if*

- Untuk pernyataan *if* yang diikuti dengan pernyataan majemuk, bentuknya adalah sebagai berikut :

```
if (kondisi )  
{           //tanda awal pernyataan majemuk  
  pernyataan_1;  
  pernyataan_2;  
  .  
  .  
  pernyataan_n;  
}           //tanda akhir pernyataan majemuk
```

- 
- Pernyataan-pernyataan yang berada di antara tanda kurung { dan } akan dijalankan **hanya jika** kondisi *if* bernilai TRUE

# Pernyataan *if else*

- Pernyataan *if else* bentuk umumnya :

```
if (kondisi)
    pernyataan_1;
else
    pernyataan_2;
```

- **Kondisi** diuji dulu, nilainya TRUE ataukah FALSE
- Jika nilainya TRUE, maka pernyataan\_1 akan dieksekusi
- Jika nilai FALSE, maka pernyataan\_2 akan dieksekusi

# Pernyataan *if else*

- Pernyataan ***if else*** juga bisa diikuti dengan pernyataan majemuk :

```
if (kondisi )
{
    pernyataan_1;
    .
    .
    pernyataan_n;
} //tanda akhir blok if
else
{
    pernyataan_1;
    .
    .
    pernyataan_n;
} //tanda akhir blok else
```

- Pernyataan-pernyataan pada **blok *if*** akan dijalankan **hanya jika** kondisi *if* bernilai **TRUE**.
- Pernyataan-pernyataan pada **blok *else*** akan dijalankan **hanya jika** kondisi *if* bernilai **FALSE**.

# Contoh

```
#include <stdio.h>

main()
{
    float total, diskon = 0.0f;

    printf("Total belanja : ");
    scanf("%f", &total);
    if(total <= 200000)
        diskon = total * 0.1;          //disc 10%
    else
        diskon = total * 0.25;         //disc 25%
    printf("Total yang harus dibayar : %8.2f\n",
        total - diskon);
}
```



# Latihan

1. Buat program untuk menginputkan sebuah bilangan, kemudian cetak ke layar bilangan tersebut dan beri komentar apakah bilangan tersebut ganjil atau genap.

Contoh:        input   = 15

Output = Bilangan yang diinputkan adalah 15.

Bilangan tersebut adalah bilangan ganjil.

2. Dengan menggunakan if ... else dan operator logika OR buat program untuk mendefinisikan sebuah karakter yang diinputkan adalah vokal atau konsonan.
3. Buat program menggunakan pernyataan *if* adalah untuk menentukan besarnya potongan harga yang diterima oleh seorang pembeli, berdasarkan kriteria :
  - tidak ada potongan harga jika total pembelian kurang dari Rp. 100.000 (dalam hal ini potongan harga diinisialisasi dengan nol).
  - bila total pembelian lebih dari atau sama dengan Rp. 100.000, potongan harga yang diterima dirubah menjadi sebesar 5% dari total pembelian.

Contoh        input   = 50.000

Output = Total yang harus dibayar adalah Rp. 50.000



# Latihan

4. Gunakan pernyataan if...else untuk membuat program yang menerima 2 buah bilangan bulat masukan. Tampilkan hasil dari pembagian bilangan pertama dengan bilangan kedua, dengan ketelitian 3 desimal.

Input : bil1 dan bil2

Output : hasil bagi bil1 dengan bil2

Nilai tambah : program bisa mengecek pembagian dengan nol, yaitu jika bilangan kedua adalah nol, maka tidak dilakukan proses pembagian, namun ditampilkan pesan kesalahannya (*division by zero*).



# Latihan

5. Gunakan pernyataan if...else untuk membuat program yang menerima apakah sebuah tahun yang diinputkan adalah tahun kabisat atau bukan.

Input : 2010

Output : Bukan kabisat

Catatan : Tahun kabisat adalah tahun yang merupakan kelipatan 4, kelipatan 400, bukan kelipatan 100 atau selainnya.

# Referensi

1. Brian W. Kernighan, Dennis M. Ritchie (2012): The C Programming Language : Ansi C Version 2 Edition, PHI Learning
2. Byron Gottfried (2010) : Programming with C, Tata McGraw - Hill Education
3. [Kochan Stephen](#) (2004) : Programming in C, 3rd Edition, Sams
4. K. N. King (2008) : C Programming: A Modern Approach, 2nd Edition, W. W. Norton & Company
5. Abdul Kadir (2012) : Algoritma & Pemrograman Menggunakan C & C++, Andi Publisher, Yogyakarta
6. <http://www.gdsw.at/languages/c/programming-bbrown/>
7. <https://www.petanikode.com/tutorial/c/>
8. <http://www.cprogramming.com/tutorial/c-tutorial.html>



**bridge to the future**

<http://www.eepis-its.edu>