

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 9
“IF-THEN”



DISUSUN OLEH:
ICHYA ULUMIDDIIN
103112400076
S1 IF-12-01
DOSEN:
Yohani Setiya Rafika Nur, M. Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024/2025

DASAR TEORI

1. Pendahuluan

Dalam pemrograman Go, pernyataan `else if` adalah bagian penting dari struktur kontrol yang memungkinkan pengembang untuk mengevaluasi beberapa kondisi secara berurutan. Struktur ini memungkinkan program untuk mengeksekusi blok kode tertentu berdasarkan kondisi yang terpenuhi pertama kali.

1. Go memiliki beberapa karakteristik unik dalam penggunaan `else-if`:

- **Tidak Memerlukan Tanda Kurung:** Berbeda dengan beberapa bahasa pemrograman lain, Go tidak memerlukan tanda kurung di sekitar kondisi dalam pernyataan `if` atau `else if`.
- **Kurung Kurawal Wajib:** Penggunaan kurung kurawal `{}` wajib dalam Go, bahkan untuk pernyataan satu baris. Ini membantu mencegah kesalahan yang dapat terjadi di bahasa lain di mana kurung kurawal bersifat opsional.
- **Deklarasi Variabel Pendek:** Go memungkinkan deklarasi variabel pendek dalam pernyataan `if`, yang lingkungannya terbatas pada blok `if` tersebut.

CONTOH SOAL

1. Latihan 1

Buatlah program dengan bahasa Go yang digunakan untuk menentukan apakah seorang penduduk bisa membuat KTP atau tidak. Syarat utama membuat KTP adalah berusia minimal 17 tahun dan memiliki kartu keluarga.

Masukan terdiri dari dua baris. Baris pertama adalah bilangan bulat yang menyatakan usia, sedangkan baris kedua adalah boolean yang menyatakan memiliki KTP atau tidak.

Keluaran berupa teks "bisa membuat KTP" atau "belum bisa membuat KTP" sesuai dengan masukan yang diberikan.

Source Code:

```
package main

import "fmt"

func main() {
    var usia int
    var kk bool
    fmt.Scan(&usia, &kk)
    if usia >= 17 && kk {
        fmt.Println("bisa membuat KTP")
    } else {
        fmt.Println("belum bisa membuat KTP")
    }
}
```

Output:

```
17
true
bisa membuat KTP
```

Deskripsi Program:

Program tersebut bertujuan untuk menentukan apakah seseorang memenuhi syarat untuk membuat Kartu Tanda Penduduk (KTP) berdasarkan usia dan kepemilikan Kartu Keluarga (KK). Program menggunakan bahasa pemrograman Go dan membaca dua input dari pengguna: usia (dalam bilangan bulat) dan kk (dalam bentuk boolean, yang menunjukkan apakah pengguna memiliki KK).

2. Latihan 2

Buatlah suatu program Go yang digunakan untuk menentukan suatu alfabet yang diberikan adalah vokal atau konsonan.

Masukan terdiri dari suatu huruf.

Keluaran berupa teks yang menyatakan huruf adalah "vokal" atau "konsonan" atau "bukan huruf".

Source Code:

```
package main

import "fmt"

func main() {
    var x rune
    var huruf, vKecil, vBesar bool
    fmt.Scanf("%c", &x)
    huruf = (x >= 'a' && x <= 'z') || (x >= 'A' && x <= 'Z')
    vKecil = x == 'a' || x == 'i' || x == 'u' || x == 'e' || x == 'o'
    vBesar = x == 'A' || x == 'I' || x == 'U' || x == 'E' || x == 'O'
    if huruf && (vKecil || vBesar) {
        fmt.Println("vokal")
    } else if huruf && !(vKecil || vBesar) {
        fmt.Println("konsonan")
    } else {
        fmt.Println("bukan huruf")
    }
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_
ndom.go"
A
vokal
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_
ndom.go"
9
bukan huruf
```

Deskripsi Program:

Program tersebut bertujuan untuk mengidentifikasi karakter yang diinputkan oleh pengguna, apakah termasuk huruf vokal, huruf konsonan, atau bukan huruf. Program ditulis dalam bahasa pemrograman Go dan menerima input berupa satu karakter yang kemudian diperiksa menggunakan logika kondisional.

3. Latihan 3

Buatlah program untuk menentukan apakah digit dalam suatu bilangan terurut membesar, mengecil, atau tidak terurut. Bilangan hanya terdiri dari empat digit saja, atau lebih besar atau sama dengan 1000 dan kecil sama atau sama dengan 9999.

Source Code:

```
package main

import "fmt"

func main() {
    var bilangan, d1, d2, d3, d4 int
    var teks string
    fmt.Print("Bilangan: ")
    fmt.Scan(&bilangan)
    d4 = bilangan % 10
    d3 = (bilangan % 100) / 10
    d2 = (bilangan % 1000) / 100
    d1 = bilangan / 1000

    if d1 < d2 && d2 < d3 && d3 < d4 {
        teks = "terurut membesar"
    } else if d1 > d2 && d2 > d3 && d3 > d4 {
        teks = "terurut mengecil"
    } else {
        teks = "tidak terurut"
    }
    fmt.Println("Digit Pada bilangan", bilangan, teks)
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
odul10_pt2\coso3.go"
Bilangan: 4567
Digit Pada bilangan 4567 terurut membesar
```

Deskripsi Program:

Program tersebut bertujuan untuk memeriksa susunan digit dalam sebuah bilangan empat digit dan menentukan apakah digit-digitnya terurut membesar, terurut mengecil, atau tidak terurut. Program ditulis dalam bahasa pemrograman Go dan bekerja dengan cara menerima input berupa bilangan empat digit dari pengguna, kemudian memisahkan setiap digitnya untuk dianalisis.

SOAL LATIHAN

- 1) PT POS membutuhkan aplikasi perhitungan biaya kirim berdasarkan berat parcel. **Maka, buatlah program BiayaPos untuk menghitung biaya pengiriman tersebut dengan ketentuan sebagai berikut!**

Dari berat parcel (dalam gram), harus dihitung total berat dalam kg dan sisanya (dalam gram). Biaya jasa pengiriman adalah Rp. 10.000,- per kg. Jika sisa berat tidak kurang dari 500 gram, maka tambahan biaya kirim hanya Rp. 5,- per gram saja. Tetapi jika kurang dari 500 gram, maka tambahan biaya akan dibebankan sebesar Rp. 15,- per gram. Sisa berat (yang kurang dari 1kg) digratiskan biayanya apabila total berat ternyata lebih dari 10kg

Source Code

```
package main

import "fmt"

func main() {
    var beratParsel int
    var totalKg, sisaGram int
    var biayaKg, biayaSisa, totalBiaya int
    fmt.Print("Berat parcel (gram): ")
    fmt.Scan(&beratParsel)
    totalKg = beratParsel / 1000
    sisaGram = beratParsel % 1000
    biayaKg = totalKg * 10000
    if totalKg > 10 {
        biayaSisa = 0
    } else if sisaGram >= 500 {
        biayaSisa = sisaGram * 5
    } else {
        biayaSisa = sisaGram * 15
    }
    totalBiaya = biayaKg + biayaSisa
    fmt.Printf("Detail berat: %d kg + %d gr\n", totalKg, sisaGram)
    fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayaKg,
    biayaSisa)
    fmt.Printf("Total biaya: Rp. %d\n", totalBiaya)
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>  
OMework_WEEK10(LAB)\soal1.go"  
Berat parcel (gram): 8500  
Detail berat: 8 kg + 500 gr  
Detail biaya: Rp. 80000 + Rp. 2500  
Total biaya: Rp. 82500
```

Deskripsi Program:

Program ini bertujuan untuk menghitung biaya pengiriman parcel berdasarkan beratnya, dengan perhitungan yang memperhatikan detail berat dalam kilogram dan gram. Program menerima input berat parcel dalam satuan gram, kemudian memisahkan berat tersebut menjadi dua bagian: total berat dalam kilogram dan sisa berat dalam gram. Biaya pengiriman dihitung berdasarkan aturan tertentu: Rp. 10.000 untuk setiap kilogram, dan tambahan biaya untuk sisa berat dihitung berdasarkan jumlah gram. Jika sisa berat 500 gram atau lebih, biaya tambahan adalah Rp. 5 per gram, sedangkan jika kurang dari 500 gram, biaya tambahan adalah Rp. 15 per gram. Namun, jika total berat melebihi 10 kilogram, biaya untuk sisa gram diabaikan. Program ini mencetak detail berat (dalam format kilogram dan gram), rincian biaya dari setiap komponen, serta total biaya keseluruhan. Dengan demikian, program ini memudahkan pengguna untuk menghitung biaya pengiriman dengan akurat dan transparan.

- 2) Diberikan sebuah nilai akhir mata kuliah (NAM) [0..100] dan standar penilaian nilai mata kuliah (NMK) sebagai berikut:

NAM	NMK
NAM > 80	A
72.5 < NAM <= 80	AB
65 < NAM <= 72.5	B
57.5 < NAM <= 65	BC
50 < NAM <= 57.5	C
40 < NAM <= 50	D
NAM <= 40	E

Program berikut menerima input sebuah bilangan riil yang menyatakan NAM. Program menghitung NMK dan menampilkannya.

```
package main
import "fmt"
func main() {
    var nam float64
    var nmk string
    fmt.Print("Nilai akhir mata kuliah: ")
    fmt.Scan(&nam)
    if nam > 80 {
        nmk = "A"
    }
    if nam > 72.5 {
        nmk = "AB"
    }
    if nam > 65 {
        nmk = "B"
    }
    if nam > 57.5 {
        nmk = "BC"
    }
    if nam > 50 {
        nmk = "C"
    }
    if nam > 40 {
        nmk = "D"
    } else if nam <= 40 {
        nmk = "E"
    }
    fmt.Println("Nilai mata kuliah: ", nmk)
}
```


Jawablah pertanyaan-pertanyaan berikut:

- a. Jika nam diberikan adalah 80.1, apa keluaran dari program tersebut? Apakah eksekusi program tersebut sesuai spesifikasi soal?
- b. Apa saja kesalahan dari program tersebut? Mengapa demikian? Jelaskan alur program seharusnya!
- c. Perbaiki program tersebut! Ujilah dengan masukan: 93.5; 70.6; dan 49.5. Seharusnya keluaran yang diperoleh adalah 'A', 'B', dan 'D'.

A. Keluaran dari program untuk nam = 80.1 adalah D, karena:

- Kondisi if nam > 80 { nam = "A" } benar, tetapi nilai nmk tidak diperbarui karena kesalahan pada implementasi variabel.
- Semua kondisi lainnya tidak mengatur nilai variabel nmk dengan benar, sehingga menghasilkan nilai terakhir pada else if nam <= 40 menjadi D.

Eksekusi program tidak sesuai spesifikasi soal, karena program tidak menghasilkan nilai yang benar sesuai tabel standar penilaian, yaitu A untuk nam > 80.

B. Kesalahan dari Program tersebut:

- Variabel nam digunakan sebagai angka (float64) tetapi kemudian diubah menjadi string, yang seharusnya diatur pada variabel nmk.
- Semua kondisi menggunakan pernyataan if terpisah, sehingga nilai variabel nmk terus diperbarui tanpa memperhatikan urutan logika. Harusnya menggunakan struktur if-else if.
- Kondisi tidak diatur dari rentang nilai terbesar ke terkecil, sehingga hasilnya tidak sesuai tabel.
- Tidak ada validasi untuk memastikan input berada dalam rentang [0..100].

C.

```
package main

import "fmt"

func main() {
    var nam float64
    var nmk string
    fmt.Scan(&nam)
    if nam < 0 || nam > 100 {
        fmt.Println("Nilai tidak valid! Harus dalam rentang 0-100.")
        return
    }
    if nam > 80 {
        nmk = "A"
    } else if nam > 72.5 {
        nmk = "AB"
    } else if nam > 65 {
        nmk = "B"
    } else if nam > 57.5 {
        nmk = "BC"
    } else if nam > 50 {
        nmk = "C"
    } else if nam > 40 {
        nmk = "D"
    } else {
        nmk = "E"
    }
    fmt.Println(nmk)
}
```

Output:

```
OMEWORK_WEEK10(LAB)\soal2.go"
93.5
A
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro\
OMEWORK_WEEK10(LAB)\soal2.go"
70.6
B
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro\
OMEWORK_WEEK10(LAB)\soal2.go"
49.5
D
```

- 3) Sebuah bilangan bulat b memiliki faktor bilangan $f > 0$ jika f habis membagi b . Contoh: 2 merupakan faktor dari bilangan 6 karena 6 habis dibagi 2.

Buatlah program yang menerima input sebuah bilangan bulat b dan $b > 1$. Program harus dapat mencari dan menampilkan semua faktor dari bilangan tersebut!

Source Code (1):

```
package main

import "fmt"

func main() {
    var b int
    fmt.Scan(&b)
    if b <= 1 {
    } else {
        fmt.Printf("Bilangan: %d\n", b)
        fmt.Print("Faktor: ")
        for f := 1; f <= b; f++ {
            if b%f == 0 {
                fmt.Print(f, " ")
            }
        }
        fmt.Println()
    }
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
OMEWORK_WEEK10(LAB)\soal3.go"
12
Bilangan: 12
Faktor: 1 2 3 4 6 12
```

Bilangan bulat $b > 0$ merupakan bilangan prima p jika dan hanya jika memiliki persis dua faktor bilangan saja, yaitu 1 dan dirinya sendiri.

Lanjutkan program sebelumnya. Setelah menerima masukan sebuah bilangan bulat $b > 0$. Program tersebut mencari dan menampilkan semua faktor bilangan tersebut.

Kemudian, program menentukan apakah b merupakan bilangan prima.

Source Code (2) :

```
package main

import "fmt"

func main() {
    var b int
    var jumlahFaktor int

    fmt.Scan(&b)
    if b <= 0 {
    } else {
        fmt.Printf("Bilangan: %d\n", b)
        fmt.Print("Faktor: ")
        for f := 1; f <= b; f++ {
            if b%f == 0 {
                fmt.Print(f, " ")
                jumlahFaktor++
            }
        }
        fmt.Println()
        if jumlahFaktor == 2 {
            fmt.Println("Prima: True")
        } else {
            fmt.Println("Prima: False")
        }
    }
}
```

Output :

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1
OMEWORK_WEEK10(LAB)\soal3(pt2).go"
12
Bilangan: 12
Faktor: 1 2 3 4 6 12
Prima: False
```

Deskripsi Program :

Program ini bertujuan untuk mencari semua faktor dari sebuah bilangan bulat positif dan menentukan apakah bilangan tersebut adalah bilangan prima. Program ditulis dalam bahasa Go dan menerima input berupa bilangan bulat b dengan syarat $b > 0$. Jika input bilangan tidak valid (misalnya, $b \leq 0$), program akan memberikan pesan peringatan dan menghentikan proses lebih lanjut.

Setelah input valid diterima, program akan mencetak bilangan tersebut, kemudian mencari semua faktornya. Faktor dihitung dengan cara memeriksa setiap bilangan dari 1 hingga b apakah bilangan tersebut habis membagi b tanpa sisa. Semua faktor yang ditemukan dicetak dalam satu baris dengan spasi sebagai pemisah. Selain itu, program menghitung jumlah faktor bilangan tersebut untuk menentukan apakah bilangan tersebut merupakan bilangan prima. Bilangan dikatakan prima jika memiliki tepat dua faktor, yaitu 1 dan dirinya sendiri. Jika jumlah faktor adalah dua, program mencetak "Prima: Benar"; jika tidak, mencetak "Prima: Salah".

DAFTAR PUSTAKA

- Agung, N. (2019, February 17). *Seleksi Kondisi*. Retrieved from Dasar Pemrograman Golang: <https://dasarpemrogramangolang.novalagung.com/A-seleksi-kondisi.html>
- Kurniawan, R. (2022, July 4). *If Expression - Belajar Golang Dari Dasar*. Retrieved from Ruang Developer: https://blog.ruangdeveloper.com/golang-if-expression/#google_vignette