

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 10
“ELSE-IF”



DISUSUN OLEH:
Muhammad Shabrian Fadly
103112400087
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

If Expression

If expression akan sering digunakan dalam membuat program. If expression digunakan untuk logika pengkondisian (percabangan). If expression akan mengeksekusi kode di dalamnya apabila kondisi yang ditentukan bernilai benar (true). If expression biasa digunakan bersamaan dengan operator perbandingan untuk mendapatkan nilai boolean.

Else Expression

Jika if expression mengeksekusi blok kode apabila kondisinya benar, maka else expression adalah kebalikan dari if expression. Else expression akan mengeksekusi blok kode di dalamnya apabila kondisinya bernilai salah. Else expression tidak bisa berdiri sendiri. Ia harus dipasangkan dengan if expression. Else if expression digunakan untuk menambahkan kondisi pada percabangan yang kita buat. Else if expression ditambahkan diantara if expression dan else expression.

CONTOH SOAL

1. Contoh soal 1

Input:

```
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:

```
PS D:\Coding manja> go run "d:\Coding manja\modul10\contoh1\tempCodeRunnerFile.go"
17
true
bisa membuat KTP
PS D:\Coding manja> go run "d:\Coding manja\modul10\contoh1\tempCodeRunnerFile.go"
20
false
belum bisa membuat KTP
```

Deskripsi Program:

Program ini bertujuan untuk mengecek dua kondisi (usia dan status Kartu Keluarga) untuk memutuskan apakah seseorang memenuhi syarat untuk membuat KTP. Jika keduanya terpenuhi, maka program mengizinkan pembuatan KTP, jika tidak, program memberitahukan bahwa belum bisa membuat KTP.

2. Contoh soal 2

Input:

```
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 D:\Coding manja> go run "d:\Coding manja\modul10\contoh2\tempCodeRunnerFile.go"
a
vokal
PS D:\Coding manja> go run "d:\Coding manja\modul10\contoh2\tempCodeRunnerFile.go"
v
konsonan
PS D:\Coding manja> go run "d:\Coding manja\modul10\contoh2\tempCodeRunnerFile.go"
1
bukan huruf
```

Deskripsi Program:

Program ini mengidentifikasi karakter yang dimasukkan oleh pengguna dan memeriksa apakah karakter tersebut termasuk huruf vokal (baik kecil atau besar), huruf konsonan, atau bukan huruf sama sekali. Dengan menggunakan pengkondisian yang tepat, program ini memberikan output yang sesuai berdasarkan karakter input yang diberikan.

3. Contoh soal 3

Input:

```
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 D:\Coding manja> go run "d:\Coding manja\modul10\contoh3\contoh3.go"
bilangan 2489
digit pada bilangan 2489 terurut membesar
PS D:\Coding manja> go run "d:\Coding manja\modul10\contoh3\contoh3.go"
bilangan 3861
digit pada bilangan 3861 tidak terurut
PS D:\Coding manja> go run "d:\Coding manja\modul10\contoh3\contoh3.go"
bilangan 9651
digit pada bilangan 9651 terurut mengecil
```

Deskripsi Program:

Program ini menerima input berupa bilangan empat digit dan memeriksa apakah digit-digit tersebut terurut membesar, mengecil, atau tidak terurut. Program ini menggunakan operasi modulus dan pembagian untuk memisahkan digit-digit bilangan, kemudian mengevaluasi urutan digit menggunakan pengkondisian. Program ini memberikan output yang sesuai berdasarkan urutan digit yang ditemukan.

Latihan Soal

1. Latihan soal 1

Statement Perulangan: 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.

Input:

```
package main

import "fmt"

func main() {
    var beratParsel int
    fmt.Print("Berat parcel (gram): ")
    fmt.Scan(&beratParsel)

    kg := beratParsel / 1000
    gramSisa := beratParsel % 1000

    biayaKg := kg * 10000

    var biayaSisa int
    if kg > 10 {
        biayaSisa = 0
    } else {
        if gramSisa >= 500 {
            biayaSisa = gramSisa * 5
        } else {
            biayaSisa = gramSisa * 15
        }
    }

    totalBiaya := biayaKg + biayaSisa
    fmt.Printf("Detail berat: %d kg + %d gr\n", kg, gramSisa)
    fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayaKg, biayaSisa)
    fmt.Printf("Total biaya: Rp. %d\n", totalBiaya)
}
```

Output:

```
PS D:\Coding manja> go run "d:\Coding manja\modul10\latsol1\tempCodeRunnerFile.go"
Berat parcel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS D:\Coding manja> go run "d:\Coding manja\modul10\latsol1\tempCodeRunnerFile.go"
Berat parcel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
```

Deskripsi Program: Program ini menghitung biaya pengiriman parcel berdasarkan beratnya dalam gram dengan mempertimbangkan tarif per kilogram dan tarif untuk sisa gram. Jika berat parcel lebih dari 10 kg, tidak ada biaya tambahan untuk sisa gram. Jika berat parcel kurang dari 10 kg, biaya tambahan dihitung berdasarkan jumlah gram sisa, dengan tarif berbeda untuk sisa gram lebih besar atau sama dengan 500 gram dan kurang dari 500 gram. Program ini menghasilkan rincian biaya dan total biaya berdasarkan input pengguna.

2. Latihan Soal2

Statement Perulangan:

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.

```
1 package main
2 import "fmt"
3 func main() {
4     var nam float64
5     var nmk string
6     fmt.Print("Nilai akhir mata kuliah: ")
7     fmt.Scan(&nam)
8     if nam > 80 {
9         nmk = "A"
10    }
11    if nam > 72.5 {
12        nmk = "AB"
13    }
14    if nam > 65 {
15        nmk = "B"
16    }
17    if nam > 57.5 {
18        nmk = "BC"
19    }
20    if nam > 50 {
21        nmk = "C"
22    }
23    if nam > 40 {
24        nmk = "D"
25    } else if nam <= 40 {
26        nmk = "E"
27    }
28    fmt.Println("Nilai mata kuliah: ", nmk)
29 }
```


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. Jika nam diberikan adalah 80.1, apa keluaran dari program tersebut? Apakah eksekusi program tersebut sesuai spesifikasi soal?

- **Keluaran Program:** Tidak ada keluaran valid, karena nilai variabel nmk tidak diubah sepanjang program. Akhirnya, `fmt.Println` akan mencetak nilai kosong untuk nmk.
- **Apakah Sesuai Spesifikasi?** Tidak, eksekusi program tidak sesuai spesifikasi soal. Kesalahan utamanya adalah:
 1. Program mencoba mengisi nilai string ke variabel nam, padahal nam bertipe `float64`.
 2. Variabel nmk (seharusnya menyimpan nilai huruf) tidak pernah diubah di dalam program.
 3. Struktur logika menggunakan hanya `if`, bukan `else if`, sehingga semua kondisi akan dievaluasi secara independen tanpa berhenti di kondisi yang cocok pertama kali.

B. Apa saja kesalahan dari program tersebut?

1. **Kesalahan Penggunaan Variabel:**
 - Variabel nam yang bertipe `float64` digunakan untuk menyimpan nilai string (misalnya "A"), yang tidak diperbolehkan di Go.
 - Variabel nmk, yang seharusnya digunakan untuk menyimpan hasil nilai huruf, tidak pernah diubah nilainya dalam program.
2. **Kesalahan Logika:**
 - Program menggunakan banyak `if` terpisah tanpa `else if`. Akibatnya, program akan terus mengevaluasi semua kondisi meskipun salah satu sudah terpenuhi.
 - Kondisi `if nam > 40` dan `else if nam <= 40` berada di bagian bawah, sehingga nilai nam di atas 40 akan menghasilkan D, meskipun kondisi sebelumnya terpenuhi.
3. **Kesalahan Penulisan dan Output:**
 - Tanda kutip ganda pada `fmt.Println` menggunakan gaya yang salah (seharusnya menggunakan `"`, bukan `"` atau `"`).
 - Program tidak menampilkan nilai yang benar karena logika tidak berhenti pada kondisi yang tepat.

C. Perbaiki Program

```
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"
    } 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("Nilai mata kuliah:", nmk)
}
```

Output:

```
PS D:\Coding manja> go run "d:\Coding manja\modul10\latsol2\latsol2.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS D:\Coding manja> go run "d:\Coding manja\modul10\latsol2\latsol2.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
```

3. Latihan Soal 3

Statement Perulangan: 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!

Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

Input:

```
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 D:\Coding manja> go run "d:\Coding manja\modul10\latsol3\latsol3.go"
7
Bilangan: 7
Faktor: 1 7
```

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.

Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read)

Input:

```
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 D:\Coding manja> go run "d:\Coding manja\modul10\latso13\latso13-2.go\latso13-2.go"
7
Bilangan: 7
Faktor: 1 7
Prima: True
```

Deskripsi Program:

Program ini menerima input berupa sebuah bilangan bulat positif dan menampilkan semua faktor dari bilangan tersebut. Setelah itu, program menentukan apakah bilangan tersebut adalah bilangan prima dengan memeriksa jumlah faktor yang dimiliki. Jika jumlah faktor sama dengan 2, maka bilangan tersebut adalah bilangan prima. Jika tidak, bilangan tersebut bukan bilangan prima. Program ini juga tidak melakukan apapun jika input bilangan adalah 0 atau bilangan negatif.

DAFTAR PUSTAKA

<https://blog.ruangdeveloper.com/golang-if-expression/>

<https://jadifullstack.com/kondisional-golang-if-else-switch/>