

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 10

ELSE - IF



Disusun oleh:

Manggala Patra Raditya

109082500179

S1IF-13-02

Asisten Praktikum

Adithana dharma putra

Alfin Ilham Berlianto

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

1. Guided 1

Source Code

```
package main

import "fmt"

func main() {

    var usia int

    var status bool

    fmt.Scanln(&usia)

    fmt.Scanln(&status)

    if usia >= 17 && status {

        fmt.Println("bisa membuat KTP")

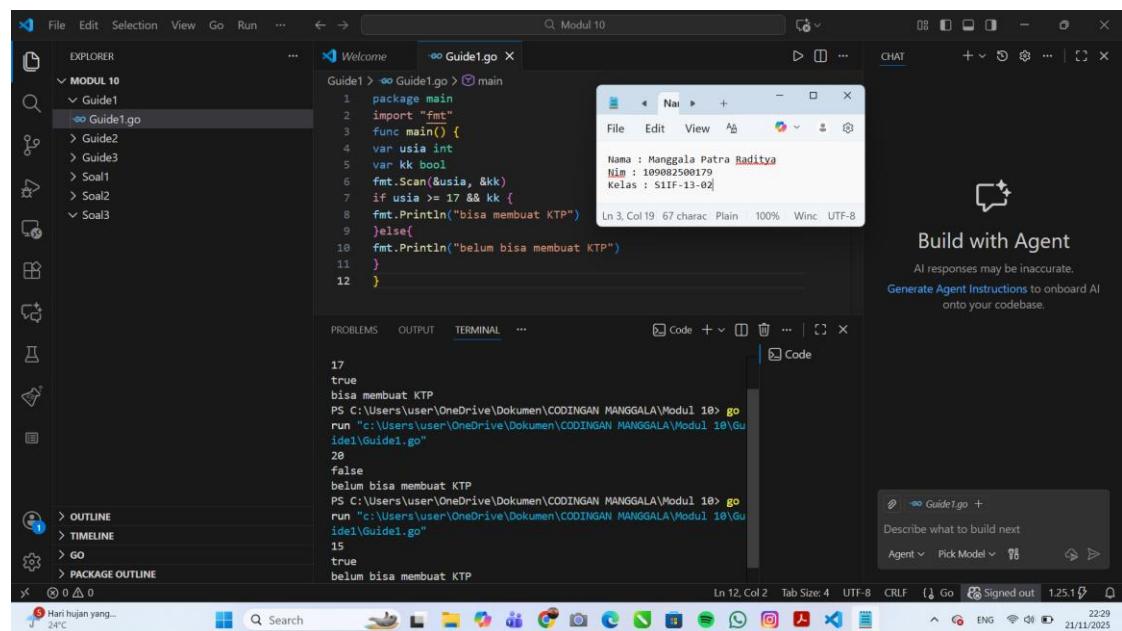
    } else {

        fmt.Println("belum bisa membuat KTP")

    }

}
```

Screenshot program



Deskripsi program

Program ini membaca dua input, yaitu **usia** (angka) dan **status** (true/false). Jika usia pengguna ≥ 17 tahun dan status bernilai **true**, program menampilkan pesan "**bisa membuat KTP**". Jika tidak memenuhi salah satu syarat, program menampilkan "**belum bisa membuat KTP**". Program menggunakan kondisi **if** untuk menentukan hasil.

1. package main – Menandai program utama.
2. import "fmt" – Mengambil paket untuk input/output.
3. func main() – Fungsi utama program.
4. var usia int – Menyimpan input usia.
5. var status bool – Menyimpan input status (true/false).
6. fmt.Scanln(&usia) – Membaca input usia.
7. fmt.Scanln(&status) – Membaca input status.
8. if usia ≥ 17 && status – Mengecek apakah usia ≥ 17 dan status true.
9. fmt.Println("bisa membuat KTP") – Output jika syarat terpenuhi.
10. else – Jalan alternatif jika syarat tidak terpenuhi.
11. fmt.Println("belum bisa membuat KTP") – Output jika tidak memenuhi syarat.
12. } – Menutup blok kondisi.
13. } – Menutup fungsi main.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var alfabet string
    fmt.Scan(&alfabet)
    if (alfabet >= "A" && alfabet <= "Z") || (alfabet >= "a" && alfabet <= "z") {
        if alfabet == "A" || alfabet == "a" ||
           alfabet == "I" || alfabet == "i" ||
           alfabet == "U" || alfabet == "u" ||
           alfabet == "E" || alfabet == "e" ||
           alfabet == "O" || alfabet == "o" {
            fmt.Println("vokal")
        } else {
            fmt.Println("konsonan")
        }
    }
}
```

```

    } else {
        fmt.Println("bukan huruf")
    }
}

```

Screenshot program

The screenshot shows a Go development environment with two tabs open: `Guide1.go` and `Guide2.go`. The `Guide2.go` tab contains the following 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 {
        fmt.Println("bukan huruf")
    }
}

```

The terminal window below shows the execution of `Guide2.go` with the input character 'A'. The output is 'vokal'.

Deskripsi program

Program ini digunakan untuk menentukan apakah sebuah karakter yang diinputkan oleh pengguna merupakan **huruf vokal**, **huruf konsonan**, atau **bukan huruf**.

Pengguna memasukkan satu karakter (misalnya A, b, u, 5, @). Program kemudian memeriksa:

1. **Apakah karakter tersebut huruf A-Z atau a-z**
 - Jika tidak termasuk huruf, program menampilkan "**bukan huruf**".
2. **Jika karakter adalah huruf**, program mengecek apakah termasuk huruf vokal: **A, I, U, E, O** (baik besar maupun kecil).
 - Jika ya → program menampilkan "**vokal**"
 - Jika tidak → program menampilkan "**konsonan**"

Program ini menggunakan operator logika `&&`, `||`, serta percabangan `if-else` untuk menentukan kategori huruf.

3. Guided 3

Source Code

```
package main

import (
    "fmt"
)

func main() {
    var n int

    fmt.Print("Bilangan: ")

    fmt.Scanln(&n)

    d1 := n / 1000

    d2 := (n / 100) % 10

    d3 := (n / 10) % 10

    d4 := n % 10

    if d1 < d2 && d2 < d3 && d3 < d4 {
        fmt.Printf("Digit pada bilangan %d terurut
membesar\n", n)
    } else if d1 > d2 && d2 > d3 && d3 > d4 {
        fmt.Printf("Digit pada bilangan %d terurut
mengecil\n", n)
    } else {
        fmt.Printf("Digit pada bilangan %d tidak
terurut\n", n)
    }
}
```

Screenshot program

The screenshot shows a Go code editor interface. The left sidebar lists files under 'MODUL 10' including 'Guide1.go', 'Guide2.go', 'Guide3.go', 'Soal1', 'Soal2', and 'Soal3'. The main editor window displays a Go program named 'Guide3.go' with the following code:

```
func main() {
    var n int
    fmt.Println("Bilangan: ")
    fmt.Scanln(&n)

    d1 := n / 1000
    d2 := (n / 100) % 10
    d3 := (n / 10) % 10
    d4 := n % 10
}
```

The terminal window below shows the output of running the program with input '2489':

```
Bilangan: 2489
Digit pada bilangan 2489 terurut membesar
```

The status bar at the bottom indicates the file is 'Guide3.go', the line is 'Ln 3, Col 19', and the character count is '67 chars'.

Deskripsi program

Program ini digunakan untuk menentukan apakah **empat digit** pada suatu bilangan **terurut membesar**, **terurut mengecil**, atau **tidak terurut**.

Pertama, program meminta pengguna memasukkan sebuah bilangan bulat (contoh: 3579 atau 8642). Kemudian program memisahkan bilangan tersebut menjadi empat digit:

- **d1** = digit ribuan
- **d2** = digit ratusan
- **d3** = digit puluhan
- **d4** = digit satuan

Setelah setiap digit diperoleh, program melakukan pengecekan:

1. **Jika $d1 < d2 < d3 < d4$**
Artinya setiap digit lebih besar dari digit sebelumnya.
Program menampilkan “**terurut membesar**”.
2. **Jika $d1 > d2 > d3 > d4$**
Artinya setiap digit lebih kecil dari digit sebelumnya.
Program menampilkan “**terurut mengecil**”.
3. **Selain kedua kondisi di atas**
Berarti digit tidak mengikuti urutan tertentu.
Program menampilkan “**tidak terurut**”.

Program ini menggunakan operasi pembagian dan modulus untuk memecah digit, serta struktur percabangan if–else if–else untuk menentukan hasil akhir.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var berat int

    fmt.Print("Berat parsel (gram): ")

    fmt.Scan(&berat)

    kg := berat / 1000

    sisa := berat % 1000

    biayaKg := kg * 10000

    biayaSisa := 0

    if berat <= 10000 {

        if sisa >= 500 {

            biayaSisa = sisa * 5

        } else {

            biayaSisa = sisa * 15

        }

    }

    total := biayaKg + biayaSisa

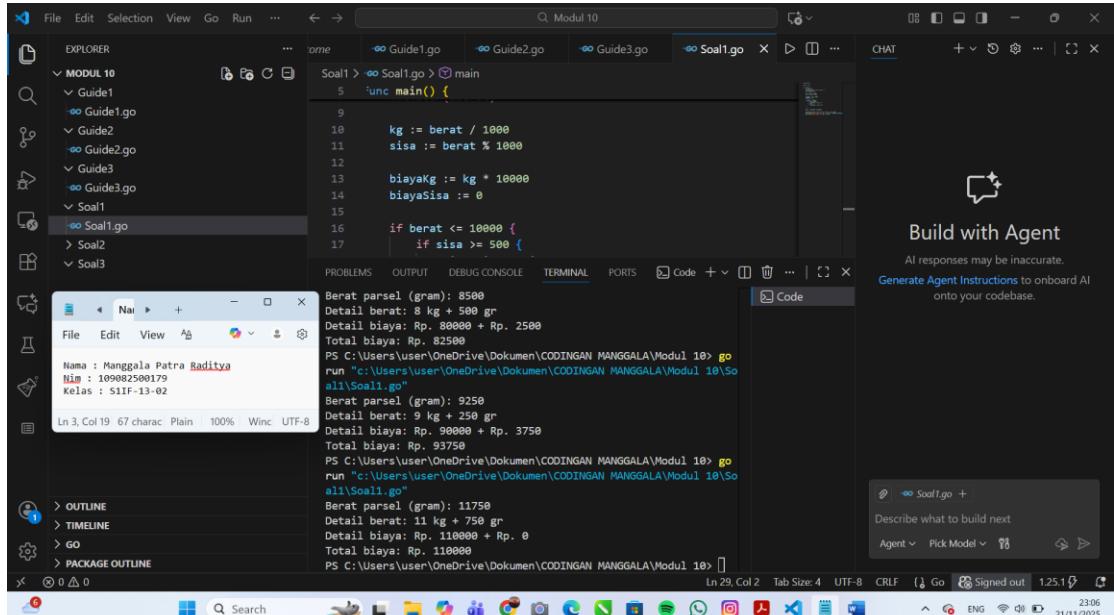
    fmt.Printf("Detail berat: %d kg + %d gr\n", kg, sisa)

    fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayaKg,
biayaSisa)

    fmt.Printf("Total biaya: Rp. %d\n", total)

}
```

Screenshot program



Deskripsi program

Program ini berfungsi untuk menghitung biaya pengiriman paket berdasarkan berat dalam gram. Pertama, program meminta input berupa berat parsel. Berat tersebut dipisahkan menjadi dua bagian, yaitu berat dalam kilogram dan sisa berat dalam gram.

Biaya dasar dihitung dari jumlah kilogram dikali tarif Rp 10.000 per kg. Untuk sisa berat, program memberikan biaya tambahan dengan ketentuan berikut:

1. Jika total berat tidak lebih dari 10 kg:
 - o Sisa \geq 500 gram dikenakan biaya Rp 5 per gram.
 - o Sisa $<$ 500 gram dikenakan biaya Rp 15 per gram.
2. Jika total berat lebih dari 10 kg, biaya sisa gram digratiskan.

Setelah perhitungan selesai, program menampilkan detail berat dalam bentuk “kg + gram”, kemudian menampilkan biaya dari kg dan biaya sisa, serta total biaya akhir.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var nam float64

    var nmk string
```

```
fmt.Println("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)  
}
```

Screenshot program

```
package main
import "fmt"
func main() {
    var nam float64
    var nmk string
}
a12\Soal2.go
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS C:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10> go run "c:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10\Soal2\Soal2.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS C:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10> go run "c:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10\Soal2\Soal2.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS C:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10> go run "c:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10\Soal2\Soal2.go"
Nilai akhir mata kuliah: 85.0
Nilai mata kuliah: A
PS C:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10>
```

Deskripsi program

Program ini digunakan untuk menentukan **nilai huruf** mata kuliah berdasarkan **nilai akhir** yang dimasukkan oleh pengguna. Nilai akhir dibaca sebagai tipe data *float64* karena dapat berupa angka desimal.

Setelah menerima input, program menggunakan rangkaian **percabangan else-if** untuk menentukan kategori nilai. Setiap rentang nilai akan menghasilkan huruf yang berbeda, yaitu:

- **A** untuk nilai ≥ 80
- **AB** untuk nilai ≥ 72.5
- **B** untuk nilai ≥ 65
- **BC** untuk nilai ≥ 57.5
- **C** untuk nilai ≥ 50
- **D** untuk nilai ≥ 40
- **E** untuk nilai di bawah 40

Setiap kondisi diperiksa berurutan dari nilai tertinggi ke terendah. Setelah sebuah kondisi terpenuhi, program akan menyimpan nilai huruf tersebut ke variabel `nmk`. Terakhir, program menampilkan hasil nilai huruf yang telah ditentukan.

Jawaban Pertanyaan:

A. Pada program tersebut, input 80.1 akan membuat semua kondisi if yang bernilai benar melakukan perubahan nilai variabel. Namun karena **semua kondisi ditulis dengan if terpisah**, bukan else if, maka urutan eksekusinya seperti ini:

- $80.1 > 80 \rightarrow$ nilai diubah menjadi "A"
- $80.1 > 72.5 \rightarrow$ ditimpa jadi "AB"
- $80.1 > 65 \rightarrow$ ditimpa lagi jadi "B"
- $80.1 > 57.5 \rightarrow$ ditimpa lagi jadi "BC"

- $80.1 > 50 \rightarrow$ ditimpa jadi "C"
- $80.1 > 40 \rightarrow$ ditimpa lagi jadi "D"

Bagian else if nam ≤ 40 tidak dijalankan.

Akhirnya nilai huruf menjadi "D"

Tetapi variabel yang dicetak adalah **nmk**, yang tidak pernah diubah.

Sehingga keluaran program:

"Nilai mata kuliah: " (kosong / string kosong)

Jadi, program **tidak sesuai spesifikasi**, karena:

- Tidak mengeluarkan nilai huruf yang benar
- Salah variabel yang dicetak
- Logika percabangan salah

B. Kesalahan pada program dan alasannya

1. Menggunakan banyak if terpisah, bukan else if

Akibatnya nilai terus ditimpa dari A \rightarrow AB \rightarrow B \rightarrow BC \rightarrow C \rightarrow D.

2. Menulis nilai huruf ke variabel yang salah

Program menulis hasil di variabel nam (float), padahal variabel huruf disediakan yaitu nmk.

3. Variabel nam bertipe float64, tetapi diisi string (salah total)

Contohnya:

nam = "A"

Ini **tidak bisa dijalankan** karena tipe datanya float64, bukan string.

4. Kondisi interval tidak benar

Misalnya:

- 80.1 lebih besar dari semua batas (72.5, 65, dst), sehingga berakhir di "D".

5. Tidak ada kondisi nilai tertinggi (misalnya $\geq 85, \geq 70$, dll)

6. Output salah

Variabel yang dicetak:

fmt.Println("Nilai mata kuliah: ", nmk)

Padahal nmk **tidak pernah diubah**.

C. Program yang sudah diperbaiki + Hasil uji

| Input Nilai | Proses Penilaian | Output Huruf |
|-------------|---|--------------|
| 93.5 | $\geq 85 \rightarrow$ masuk kategori A | A |
| 70.6 | ≥ 70 dan $< 85 \rightarrow$ masuk kategori B | B |
| 49.5 | ≥ 40 dan $< 55 \rightarrow$ masuk kategori D | D |

Program yang benar:

```
package main

import "fmt"

func main() {

    var nam float64

    var nmk string

    fmt.Print("Nilai akhir mata kuliah: ")

    fmt.Scan(&nam)

    if nam >= 85 {

        nmk = "A"

    } else if nam >= 70 {

        nmk = "B"

    } else if nam >= 55 {

        nmk = "C"

    } else if nam >= 40 {

        nmk = "D"
}
```

```
    } else {

        nmk = "E"

    }

    fmt.Println("Nilai mata kuliah:", nmk)

}
```

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var b int
    fmt.Scan(&b)
    jumlah := 0
    fmt.Print("Faktor: ")
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Print(i, " ")
            jumlah++
        }
    }
    prima := jumlah == 2
    fmt.Println("\nPrima:", prima)
}
```

Screenshot program

The screenshot shows a Go code editor interface with the following details:

- File Explorer:** Shows a project structure under "MODUL 10" with files like "Guide1.go", "Guide2.go", "Guide3.go", "Soal1.go", "Soal2.go", and "Soal3.go".
- Code Editor:** Displays the content of "Soal3.go".

```
func main() {
    jumlah := 0
    fmt.Println("Faktor: ")
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Println(i, " ")
            jumlah++
        }
    }
    if jumlah == 2 {
        Prima = true
    } else {
        Prima = false
    }
}
```
- Terminal:** Shows command-line output for running the program.

```
Faktor: 1 2 3 4 6 12
Prima: false
```

```
Faktor: 1 7
Prima: true
```
- Status Bar:** Shows file path "C:\Users\user\OneDrive\Dokumen\CODINGAN MANGGALA\Modul 10\Soal3\Soal3.go", line 21, column 2, and other system information.

Deskripsi program

Program menerima satu input berupa bilangan bulat **b**. Program kemudian mencari semua faktor dari bilangan tersebut dengan memeriksa setiap angka dari 1 sampai **b**. Jika angka tersebut dapat membagi **b** tanpa sisa, angka itu dicetak sebagai faktor, dan penghitung jumlah faktor ditambah satu. Setelah semua faktor ditemukan, program menentukan apakah **b** adalah bilangan prima dengan mengecek apakah jumlah faktornya tepat dua. Terakhir, program menampilkan hasil pengecekan prima sebagai nilai true atau false.