

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 10

IF - ELSE



Disusun oleh:

ERIC SETIAWAN

109082500197

S1IF-13-04

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN – GUIDED

1. Guided 1

Source Code

```
package main

import "fmt"

func main() {

    var usia int

    var kk bool

    fmt.Scan(&usia, &kk)

    if usia >= 17 && kk {

        fmt.Print("bisa membuat KTP")

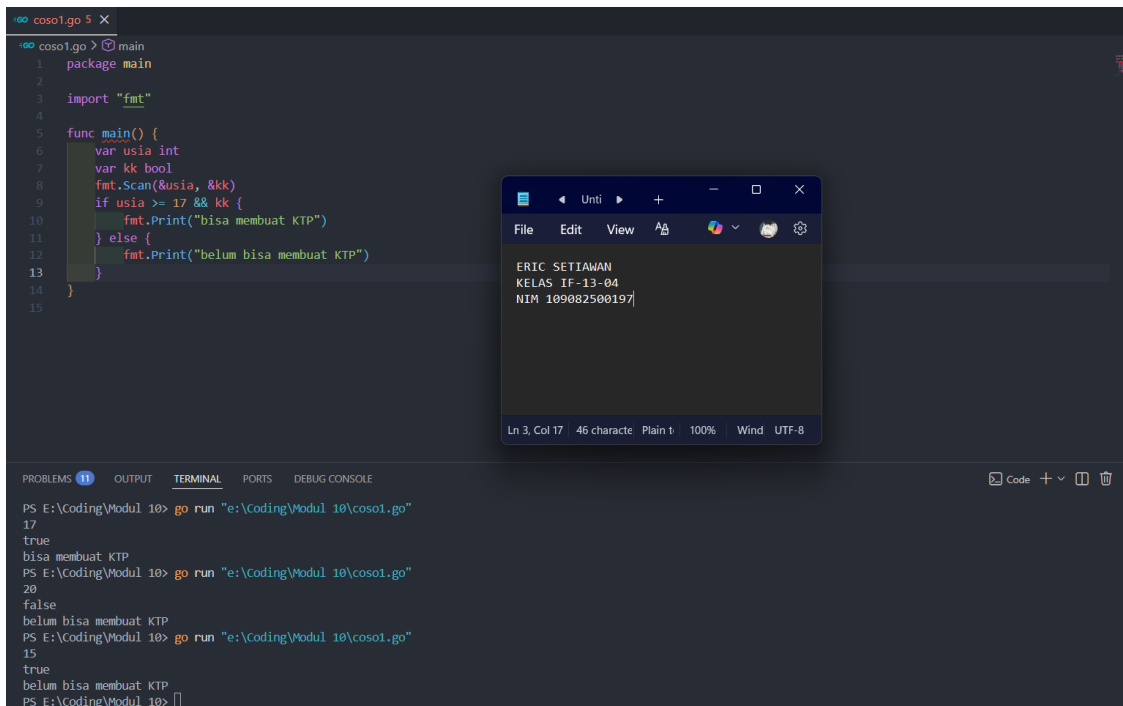
    } else {

        fmt.Print("belum bisa membuat KTP")

    }

}
```

Screenshoot program



```
cosol.go 5 X
cosol.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var usia int
7     var kk bool
8     fmt.Scan(&usia, &kk)
9     if usia >= 17 && kk {
10         fmt.Print("bisa membuat KTP")
11     } else {
12         fmt.Print("belum bisa membuat KTP")
13     }
14 }
15

PROBLEMS 11 OUTPUT TERMINAL PORTS DEBUG CONSOLE
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\cosol.go"
17
true
bisa membuat KTP
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\cosol.go"
20
false
belum bisa membuat KTP
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\cosol.go"
15
true
belum bisa membuat KTP
PS E:\Coding\Modul 10> []
```

ERIC SETIAWAN
KELAS IF-13-04
NIM 109082500197

Ln 3, Col 17 46 character Plain t 100% Wind UTF-8

Deskripsi program

Program ini menentukan apakah seseorang memenuhi syarat untuk membuat Kartu Tanda Penduduk (KTP) berdasarkan usia dan memiliki (KK). Lalu membaca dua input

usia (bilangan bulat) dan KK (boolean, true jika memiliki KK). Jika usia minimal 17 tahun dan memiliki KK (kk = true), program mencetak "bisa membuat KTP"; jika tidak, mencetak "belum bisa membuat KTP". Program mendeklarasikan dua variabel: **usia** sebagai bilangan bulat dan **kk** sebagai boolean. Menggunakan **fmt.Scan** untuk membaca input, menyimpannya ke **usia** dan **kk**. lalu **if usia >= 17 && kk**: jika usia 17 dan **kk** bernilai true, maka cetak "bisa membuat KTP". Jika kondisi tidak terpenuhi, cetak "belum bisa membuat KTP".

2. Guided 2

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")
    }
}
```

Screenshoot program

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x rune
7     var huruf, vKecil, vBesar bool
8     fmt.Scanf("%c", &x)
9     huruf = (x >= 'a' && x <= 'z') || (x >= 'A' && x <= 'Z')
10    vKecil = x == 'a' || x == 'i' || x == 'u' || x == 'e' || x == 'o'
11    vBesar = x == 'A' || x == 'I' || x == 'U' || x == 'E' || x == 'O'
12    if huruf && (vKecil || vBesar) {
13        fmt.Println("vokal")
14    } else if huruf && !(vKecil || vBesar) {
15        fmt.Println("konsonan")
16    } else {
17        fmt.Println("bukan huruf")
18    }
19 }
20
```

```
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\coso2.go"
A
vokal
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\coso2.go"
f
konsonan
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\coso2.go"
1
bukan huruf
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\coso2.go"
$
bukan huruf
PS E:\Coding\Modul 10>
```

Deskripsi program

Program mendeklarasikan variabel **x** sebagai rune untuk menyimpan karakter input. Menggunakan **fmt.Scanf("%c", &x)** untuk membaca satu karakter dari pengguna. Mengecek apakah **x** adalah huruf dengan kondisi **(x >= 'a' && x <= 'z') || (x >= 'A' && x <= 'Z')**, disimpan dalam **huruf**. Mengecek apakah **x** adalah vokal kecil (**vKecil**) atau besar (**vBesar**) dengan membandingkan terhadap 'a', 'i', 'u', 'e', 'o' dan 'A', 'I', 'U', 'E', 'O'. Jika **huruf** benar dan (**vKecil** atau **vBesar**) benar, cetak "vokal". Jika **huruf** benar tapi bukan vokal, cetak "konsonan". Jika bukan huruf, cetak "bukan huruf".

3. Guided 3

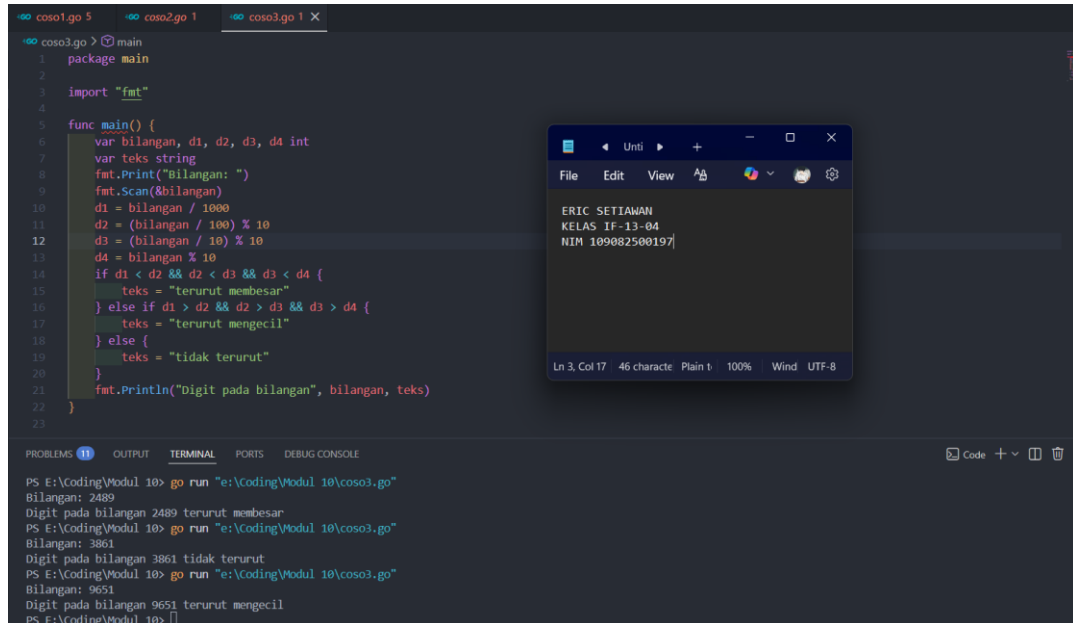
Source Code

```
package main

import "fmt"

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

Screenshoot program



The screenshot shows a Go program in a code editor and its execution output in a terminal window. The program is a simple sorting algorithm that takes a number and prints its digits in ascending or descending order based on a comparison of the digits.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan, d1, d2, d3, d4 int
7     var teks string
8     fmt.Print("Bilangan: ")
9     fmt.Scan(&bilangan)
10    d1 = bilangan / 1000
11    d2 = (bilangan / 100) % 10
12    d3 = (bilangan / 10) % 10
13    d4 = bilangan % 10
14    if d1 < d2 && d2 < d3 && d3 < d4 {
15        teks = "terurut membesar"
16    } else if d1 > d2 && d2 > d3 && d3 > d4 {
17        teks = "terurut mengecil"
18    } else {
19        teks = "tidak terurut"
20    }
21    fmt.Println("Digit pada bilangan", bilangan, teks)
22 }
23
```

The terminal window shows the output of the program for three different inputs:

```
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\coso3.go"
Bilangan: 2489
Digit pada bilangan 2489 terurut membesar
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\coso3.go"
Bilangan: 3861
Digit pada bilangan 3861 tidak terurut
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\coso3.go"
Bilangan: 9651
Digit pada bilangan 9651 terurut mengecil
PS E:\Coding\Modul 10>
```

Deskripsi program

Program mendeklarasikan variabel **bilangan** untuk input, **d1** hingga **d4** untuk digit, dan **teks** untuk hasil. Menggunakan **fmt.Scan** untuk membaca bilangan. **d1** (ribuan), **d2** (ratusan), **d3** (puluhan), **d4** (satuan) menggunakan operasi pembagian dan modulo. Mengecek kondisi: Jika **d1 < d2 < d3 < d4**, set **teks** ke "terurut membesar". Jika **d1 > d2 > d3 > d4**, set **teks** ke "terurut mengecil". Jika tidak, set **teks** ke "tidak terurut". Mencetak hasil dengan format "Digit pada bilangan bilangan dan teks".

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 {

        biayaSisa = 0

    } else {

        if sisa < 500 {

            biayaSisa = sisa * 5

        } else {

            biayaSisa = sisa * 15

        }

    }

    total := biayaKg + biayaSisa

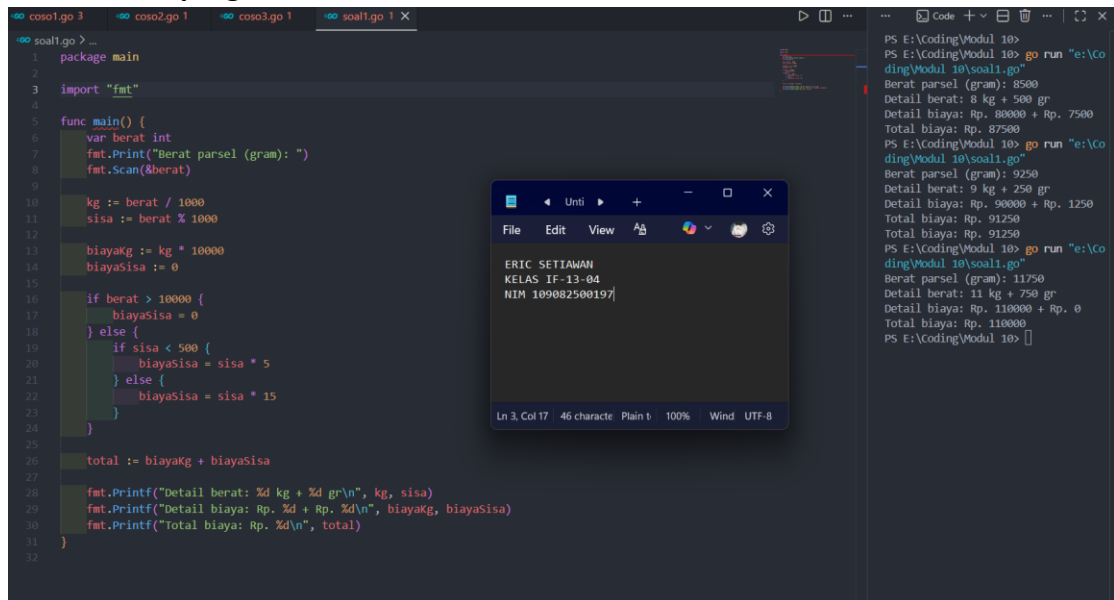
    fmt.Printf("Detail berat: %d kg + %d gr\n", kg, sisa)

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

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

}
```

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var berat int
7     fmt.Print("Berat parcel (gram): ")
8     fmt.Scan(&berat)
9
10    kg := berat / 1000
11    sisa := berat % 1000
12
13    biayaKg := kg * 10000
14    biayaSisa := 0
15
16    if berat > 10000 {
17        biayaSisa = 0
18    } else {
19        if sisa < 500 {
20            biayaSisa = sisa * 5
21        } else {
22            biayaSisa = sisa * 15
23        }
24    }
25
26    total := biayaKg + biayaSisa
27
28    fmt.Printf("Detail berat: %d kg + %d gr\n", kg, sisa)
29    fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayaKg, biayaSisa)
30    fmt.Printf("Total biaya: Rp. %d\n", total)
31 }
32
```

PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\soal1.go"

Berat parcel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 7500
Total biaya: Rp. 87500

PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\soal1.go"

Berat parcel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 1250
Total biaya: Rp. 91250

PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\soal1.go"

Berat parcel (gram): 11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 0
Total biaya: Rp. 110000

PS E:\Coding\Modul 10>

Deskripsi program

Program ini menghitung biaya pengiriman parcel berdasarkan berat dalam gram, dengan tarif Rp. 10.000 per kg dan tarif tambahan untuk gram sisa (Rp. 5 per gram jika < 500 gram, Rp. 15 per gram jika >= 500 gram), namun biaya sisa nol jika berat > 10.000 gram. Lalu membaca input berat, menghitung komponen biaya, dan mencetak detail berat, biaya per bagian, serta total biaya.

2. Tugas 2

Source code

```
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)
}

```

Screenshoot program

```

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

```

```

PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\soal2.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\soal2.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS E:\Coding\Modul 10> go run "e:\Coding\Modul 10\soal2.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS E:\Coding\Modul 10>

```

Deskripsi program

Program mendeklarasikan variabel **nam** sebagai float64 untuk nilai akhir dan **nmk** sebagai string untuk nilai huruf. Menggunakan **fmt.Scan** untuk membaca nilai.

Jika **nam > 80**, **nmk = "A"**.

Jika **nam** > 72.5, **nmk** = "AB".
Jika **nam** > 65, **nmk** = "B".
Jika **nam** > 57.5, **nmk** = "BC".
Jika **nam** > 50, **nmk** = "C".
Jika **nam** > 40, **nmk** = "D".
Jika tidak, **nmk** = "E".
Mencetak "Nilai mata kuliah: " **nmk**.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var b int
    fmt.Print("Bilangan: ")
    fmt.Scan(&b)

    fmt.Print("Faktor : ")

    jumlahFaktor := 0

    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Print(" ", i)
            jumlahFaktor++
        }
    }

    fmt.Println()

    isPrima := false
    if jumlahFaktor == 2 {
        isPrima = true
    }

    fmt.Printf("Prima   : %v\n", isPrima)
}
```

Screenshoot program

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var b int
7
8     fmt.Print("Bilangan: ")
9     fmt.Scan(&b)
10
11     fmt.Print("Faktor : ")
12
13     jumlahFaktor := 0
14
15     for i := 1; i <= b; i++ {
16         if b%i == 0 {
17             fmt.Print(" ", i)
18             jumlahFaktor++
19         }
20     }
21
22     fmt.Println()
23
24     isPrima := false
25     if jumlahFaktor == 2 {
26         isPrima = true
27     }
28
29     fmt.Printf("Prima : %v\n", isPrima)
30 }
31
```

ERIC SETIAWAN
KELAS IF-13-04
NIM 109082500197

```
PS E:\CodingModul 10> go run "e:\CodingModul 10\soal3.go"
Bilangan: 12
Faktor : 1 2 3 4 6 12
Prima : false
PS E:\CodingModul 10> go run "e:\CodingModul 10\soal3.go"
Bilangan: 7
Prima : true
PS E:\CodingModul 10> go run "e:\CodingModul 10\soal3.go"
Bilangan: 7
Faktor : 1 7
Prima : true
PS E:\CodingModul 10>
```

Deskripsi program

Program mendeklarasikan variabel **b** untuk bilangan input. Menggunakan **fmt.Scan** untuk membaca bilangan. Menginisialisasi **jumlahFaktor** ke 0. Menggunakan loop **for i := 1; i <= b; i++** untuk memeriksa setiap angka dari 1 hingga **b**: Jika **b % i == 0**, cetak **i** sebagai faktor dan tambahkan 1 ke **jumlahFaktor**. Setelah loop, set **isPrima** ke true jika **jumlahFaktor == 2**, jika tidak false. Mencetak "Prima : " diikuti dengan nilai boolean **isPrima**.