

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 10

ELSE-IF



Disusun oleh:

Leonardo Farriz Garcya

109082530036

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

LATIHAN KELAS – 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 == true{
        fmt.Print("bisa membuat ktp")
    }else{
        fmt.Print("tidak bisa membuat ktp")
    }
}
```

Screenshot program

The screenshot shows a code editor interface with a sidebar containing file navigation and a terminal window below the code editor.

Code Editor Content:

```
package main
import "fmt"
func main(){
    var usia int
    var kk bool
    fmt.Scan(&usia, &kk)
    if usia >= 17 && kk == true{
        fmt.Println("bisa membuat ktp")
    }else{
        fmt.Println("tidak bisa membuat ktp")
    }
}
```

Terminal Output:

```
PS C:\modul 10> go run "c:\modul 10\guided 10.1.go"
17
True
bisa membuat ktp
PS C:\modul 10> go run "c:\modul 10\guided 10.1.go"
20
false
tidak bisa membuat ktp
PS C:\modul 10> go run "c:\modul 10\guided 10.1.go"
19
true
tidak bisa membuat ktp
PS C:\modul 10>
```

Deskripsi program

1.package main

- Bahwa program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
"fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini

4. var usia int dan var kk bool

- Menyatakan var usia bertipe integer
- Menyatakan var kk bertipe boolean

5. fmt.Scan(&usia, &kk)

- Membaca input dari pengguna dan menyimpannya ke masing' variabel nya(usia dan kk)

6. if usia >= 17 && kk == true

- Baris ini memulai kondisi. Program mengecek dua hal sekaligus:
 - 1.apakah usia lebih besar atau sama dengan 17,

2. dan apakah kk bernilai **true** (artinya memiliki KK).

- Keduanya harus benar karena menggunakan operator **&&**.

7. `fmt.Println("bisa membuat ktp")`

- Jika dua syarat pada **if** terpenuhi, program menjalankan baris ini. Lalu ini akan mencetak teks "**bisa membuat ktp**" ke layar.

8. `else`

- kondisi ini akan terpenuhi jika saat kondisi if tidak terpenuhi.

9. `fmt.Println("tidak bisa membuat ktp")`

-Jika salah satu atau kedua syarat pada if tidak terpenuhi, baris ini dijalankan. Maka program akan menampilkan teks "**tidak bisa membuat ktp**".

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var huruf string
    fmt.Scan(&huruf)

    c := huruf[0]

    if (c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z') {
        if c == 'a' || c == 'i' || c == 'u' || c == 'e'
        || c == 'o' ||
            c == 'A' || c == 'I' || c == 'U' || c == 'E'
        || c == 'O' {
    }
```

```
        fmt.Println("vokal")

    } else {

        fmt.Println("konsonan")

    }

} else {

    fmt.Println("bukan huruf")

}

}
```

Screenshot program

Deskripsi program

1.package main

- Bawa program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
"fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini

4. var huruf string

- Menyatakan variabel huruf bertipe string

5. fmt.Scan(&huruf)

- Membaca input dari pengguna dan menyimpannya ke variabel huruf

6. c := huruf[0]

- huruf[0] mengambil karakter pertama dari string huruf. Disimpan ke variabel c (tipe byte yang mewakili 1 karakter ASCII).

7. if (c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z')

- Untuk mengecek apakah inputan huruf alfabet

- Jika salah satu saja benar, maka benar karna menggunakan operator “atau”

8. if c == 'a' || c == 'i' || c == 'u' || c == 'e' || c == 'o' || c == 'A' || c == 'I' || c == 'U' || c == 'E' || c == 'O'

- Baris ini mengecek apakah inputan huruf tersebut salah satu dari **vokal (AIUEO)** dalam bentuk a, i, u, e, o, A, I, U, E, O.

- Jika kondisi ini benar , maka program menampilkan “vokal”.

9. else { fmt.Println("konsonan")}

- Jika huruf alfabet tetapi bukan salah satu vokal, maka otomatis dianggap **konsonan**.

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

- Jika karakter tidak berada pada rentang alfabet (A–Z atau a–z), program menampilkan “bukan huruf”.

Screenshoot program

The screenshot shows a Go development environment with two windows. The left window is the code editor for file 'guided 10.2.go' containing the following Go code:

```
package main
import "fmt"
func main() {
    var huruf string
    fmt.Scan(&huruf)
    c := huruf[0]
    if (c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z') {
        if c == 'a' || c == 'i' || c == 'u' || c == 'e' || c == 'o' ||
           c == 'A' || c == 'I' || c == 'U' || c == 'E' || c == 'O' {
            fmt.Println("vokal")
        } else {
            fmt.Println("konsonan")
        }
    } else {
        fmt.Println("bukan huruf")
    }
}
```

The right window is a terminal window titled 'IF 13-04' showing the execution of the program and its output:

```
PS C:\modul 10> go run "c:\modul 10\guided 10.2.go"
A
vokal
PS C:\modul 10> go run "c:\modul 10\guided 10.2.go"
f
konsonan
PS C:\modul 10> go run "c:\modul 10\guided 10.2.go"
1
bukan huruf
PS C:\modul 10> go run "c:\modul 10\guided 10.2.go"
$ 
bukan huruf
```

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Scan(&n)

    a := n / 1000
    b := n / 100 % 10
    c := n / 10 % 10
    d := n % 10

    if a < b && b < c && c < d {
        fmt.Println("digit membesar")
    } else if a > b && b > c && c > d {
        fmt.Println("digit mengecil")
    } else {
        fmt.Println("tidak terurut")
    }
}
```

Screenshot program

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

- File Explorer:** Shows several files in the 'modul 10' directory, including 'guided 10.1.go', 'guided 10.2.go', 'guided 10.3.go', 'latihan 10.1.go', 'latihan 10.2.go', 'latihan 10.3.go', 'AT 9.1.go', 'AT 9.2.go', 'AT 9.3.go', 'AT 9.4.go', 'AT 9.5.go', 'guided 10.1.go', 'guided 10.2.go', 'guided 10.3.go', 'latihan 10.1.go', 'latihan 10.2.go', 'latihan 10.3.go', and 'tempCodeRunnerFile.go'. 'guided 10.3.go' is currently selected.
- Code Editor:** Displays the content of 'guided 10.3.go'. The code defines a package main with a function main() that reads an integer n and prints its size based on the number of digits (a, b, c) and the digit itself (d). The output of the code is shown in the terminal below.
- Terminal:** Shows the command PS C:\modul 10> go run "c:\modul 10\guided 10.3.go" followed by the output:

```
2489
digit membesar
PS C:\modul 10> go run "c:\modul 10\guided 10.3.go"
3861
tidak terurut
PS C:\modul 10> go run "c:\modul 10\guided 10.3.go"
9651
digit mengecil
PS C:\modul 10>
```

Deskripsi program

1.package main

- Bahwa program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
"fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini

4.var n int

- Menyatakan variabel n bertipe integer.

5. fmt.Scan(&n)

- Membaca input dari pengguna dan menyimpannya ke variabel huruf.

6. a := n / 1000

- Untuk digit ribuan

7. b := n / 100 % 10

- Untuk digit ratusan

8. c := n / 10 % 10

- Untuk digit puluhan

9. `d := n % 10`

- Untuk digit satuan

10. `if a < b && b < c && c < d`

- Program memastikan bahwa setiap digit dari kiri ke kanan semakin besar.

- Jika hasil inputan semua nya benar, maka akan true. Jika salah satu nya salah, maka hasilnya false.

11. `fmt.Println("digit membesar")`

- code ini akan menampilkan di layar jika if di atas True.

12. `else if a > b && b > c && c > d`

- Baris ini mengecek apakah digit-digit angka semakin mengecil dari kiri ke kanan.

- Jika hasil inputan semua nya benar, maka akan true. Jika salah satu nya salah, maka hasilnya false.

13. `fmt.Println("digit mengecil")`

- code ini akan menampilkan di layar jika else if di atas True.

14. `else { fmt.Println("tidak terurut") }`

- Bagian else akan dijalankan jika kedua kondisi sebelumnya salah. Dan menampilkan di layar "tidak terurut".

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var gram int
    fmt.Scan(&gram)
```

```
kg := gram / 1000

sisa := gram % 1000


biayaKg := kg * 10000

biayaSisa := 0


if kg > 10 {

    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)

}
```

Screenshot program

The screenshot shows a code editor interface with several tabs open. The active tab is 'latihan 10.1.go'. The code in the editor is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var gram int
7     fmt.Scan(&gram)
8
9     kg := gram / 1000
10    sisa := gram % 1000
11
12    biayaKg := kg * 10000
13    biayaSisa := 0
14
15    if kg > 10 {
16        biayaSisa = 0
17    } else {
18        if sisa >= 500 {
19            biayaSisa = sisa * 5
20        } else {
21            biayaSisa = sisa * 15
22        }
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.Println("Total biaya: Rp. %d", total)
31 }
```

Below the code editor, there is a terminal window showing the output of running the program:

```
PS C:\modul 10> go run "c:\modul 10\latihan 10.1.go"
8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
```

The terminal also displays the student's information:

```
IF 13-04
109082530036
LEONARDO FARRIZ GARCYA
```

Deskripsi program

1.package main

- Bahwa program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
"fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini

4. var gram int

- Menyatakan variabel gram bertipe integer

5. fmt.Scan(&gram)

- Membaca input dari pengguna dan menyimpannya ke variabel huruf.

6. kg := gram / 1000

- Menghitung jumlah kilogram dari total gram yang dimasukkan.

7. sisa := gram % 1000

- Menghitung sisa gram setelah kilogram diambil

8. biayaKg := kg * 10000

- Menghitung biaya pengiriman berdasarkan berat kilogram (Rp. 10.000 per kg).

9. biayaSisa := 0

- Menginisialisasi variabel biaya sisa gram dengan nilai awal 0.

10. if kg > 10 { biayaSisa = 0

- Jika berat lebih dari 10 kg, sisa gram **gratis**, jadi biaya sisa tetap 0.

11. else { if sisa >= 500 { biayaSisa = sisa * 5

- Jika berat tidak lebih dari 10 kg, maka biaya sisa akan dihitung berdasarkan aturan sisa gram.

- Jika sisa gram 500 gram atau lebih, biaya tambahan Rp. 5 per gram.

12. else { biayaSisa = sisa * 15 }

- Jika sisa gram **kurang dari 500 gram**, biaya tambahan Rp. 15 per gram.

13. total := biayaKg + biayaSisa

- Menghitung total biaya keseluruhan.

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

- Menampilkan rincian berat: jumlah kg dan sisa gram.

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

- Menampilkan rincian biaya: biaya kg dan biaya sisa gram.

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

- Menampilkan total biaya yang harus dibayar.

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 {
    nam = "A"
}

if nam > 72.5 {
    nam = "AB"
}

if nam > 65 {
    nam = "B"
}

if nam > 57.5 {
    nam = "BC"
}

if nam > 50 {
    nam = "C"
}

if nam > 40 {
    nam = "D"
} else if nam <= 40 {
    nam = "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?

Jawab: Tidak sesuai.

Program tidak memberi nilai huruf yang benar dan memiliki error logika.

b. Apa saja kesalahan dari program tersebut? Mengapa demikian? Jelaskan alur program seharusnya!

Jawab:

1. Variabel nam dipakai untuk dua jenis tipe data (angka dan string)

Ini salah karena:

nam awalnya float64

Lalu diset menjadi "A", "B", dsb (string)

Ini tidak mungkin dilakukan di Go.

2. Semua kondisi menggunakan if tanpa else if

Saat satu kondisi benar, kondisi berikutnya tetap diperiksa

Padahal seharusnya *hanya satu nilai huruf* yang dipilih

3. Range nilai tidak eksklusif dan tidak berurutan

Seharusnya:

$\geq 85 \rightarrow A$

$\geq 70 \rightarrow B$

$\geq 55 \rightarrow C$

dst

Dan harus memakai **else if**, bukan if semua.

c. Perbaiki program tersebut! Ujilah dengan masukan: 93.5; 70.6; dan 49.5.

Seharusnya keluaran yang diperoleh adalah 'A', 'B', dan 'D'.

Jawab:

Program yang Sudah Diperbaiki

```
package main

import "fmt"

func main() {
    var nilai float64
    var huruf string
```

```
fmt.Println("Nilai akhir mata kuliah: ")  
fmt.Scan(&nilai)  
  
if nilai >= 85 {  
    huruf = "A"  
}  
else if nilai >= 70 {  
    huruf = "B"  
}  
else if nilai >= 55 {  
    huruf = "C"  
}  
else if nilai >= 40 {  
    huruf = "D"  
}  
else {  
    huruf = "E"  
}  
  
fmt.Println("Nilai mata kuliah:", huruf)  
}
```

Screenshot program

The screenshot shows a code editor interface with several tabs open. The tabs include: guided 10.1.go, guided 10.2.go, guided 10.3.go, latihan 10.1.go, latihan 10.2.go, latihan 10.3.go, AT 9.1.go, and AT 9.2.go. The file latihan 10.2.go is currently selected. The code in this file is a Go program that reads a float64 value from the user, prints it as a grade letter (A-E), and then prints the grade letter again. A terminal window is also visible, showing the execution of the program and its output.

```
package main

import "fmt"

func main() {
    var nilai float64
    fmt.Println("Masukkan nilai: ")
    fmt.Scan(&nilai)

    if nilai >= 85 {
        huruf = "A"
    } else if nilai >= 70 {
        huruf = "B"
    } else if nilai >= 55 {
        huruf = "C"
    } else if nilai >= 40 {
        huruf = "D"
    } else {
        huruf = "E"
    }

    fmt.Println("Nilai mata kuliah:", huruf)
}
```

```
PS C:\modul 10> go run "c:\modul 10\latihan 10.2.go"
Nilai mata kuliah: A
PS C:\modul 10> go run "c:\modul 10\latihan 10.2.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS C:\modul 10> go run "c:\modul 10\latihan 10.2.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS C:\modul 10>
```

3. Tugas 3

Source code

```
package main

import "fmt"

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

    fmt.Print("Faktor: ")
    prima := true

    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Printf("%d ", i)
            if i != 1 && i != b {
                prima = false
            }
        }
    }
}
```

```

        fmt.Printf("\nPrima: %v\n", prima)
    }

```

Screenshot program

```

package main
import "fmt"
func main() {
    var b int
    fmt.Scan(&b)
    fmt.Println("Faktor: ")
    for i := 1; i < b; i++ {
        if b%i == 0 {
            fmt.Printf("%d ", i)
            if i != 1 && i != b {
                prima = false
            }
        }
    }
    fmt.Printf("\nPrima: %v\n", prima)
}

```

The screenshot shows the Visual Studio Code interface with several tabs open. The active tab contains the provided Go code. To the right, a terminal window displays the execution of the program. The terminal output is as follows:

```

IF 13-0
109082530036
LEONARDO FARRIZ GARCYA

```

Deskripsi program

1.package main

- Bawa program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import “fmt”

- Format (fmt), digunakan untuk input/output standar.
- Seperti “fmt.Scanln” : untuk membaca input dari user.
- “fmt.Println” : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini

4. var b int

- Menyatakan variabel b bertipe integer

5. fmt.Scan(&b)

- Membaca input dari pengguna dan memasukkannya ke variabel b

6. `fmt.Println("Faktor: ")`

- Menampilkan teks "Faktor: " sebagai label sebelum daftar faktor muncul.

7. `prima := true`

- Menganggap awalnya bahwa bilangan tersebut adalah prima.

- Jika ditemukan faktor selain 1 dan b, nilai ini diubah menjadi false.

8. `for i := 1; i <= b; i++`

- Perulangan dari 1 sampai b untuk mengecek semua kemungkinan faktor bilangan b.

9. `if b%i == 0`

- Untuk Mengecek apakah i adalah faktor dari b. Jika ya, berarti b habis dibagi i.

10. `fmt.Printf("%d ", i)`

- Untuk mencetak faktor tersebut ke layar dalam format angka diikuti spasi.

11. `if i != 1 && i != b { prima = false }`

- Jika ditemukan faktor selain 1 atau b, maka bilangan langsung dipastikan bukan prima.

12. `fmt.Printf("\nPrima: %v\n", prima)`

- \n untuk pindah baris setelah daftar faktor.

- Menampilkan apakah bilangan tersebut prima atau tidak.

- %v digunakan untuk mencetak nilai boolean (true atau false).