

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 10

ELSE-IF



Disusun oleh:

RAFLI NURHIDAYAT

109082500152

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 umur int
    var kk bool

    fmt.Scan(&umur, &kk)

    if umur >= 17 && kk {
        fmt.Println("bisa membuat KTP")
    } else {
        fmt.Println("belum bisa membuat KTP")
    }
}
```

Screenshot program

The screenshot shows a Go code editor interface. On the left, there's a file tree for a project named 'ALPRO_PRAKTIKUM'. Inside 'minggu10', there are several files: 'laprak', 'minggu1', 'minggu2', 'minggu3', 'minggu4', 'minggu5', 'minggu6', 'minggu7(asessment)', 'minggu9', 'minggu10', 'guided', 'unguided', and three empty files ('1.go', '2.go', '3.go'). The file '1.go' is selected and open in the main editor area. The code in '1.go' is:

```
package main
import "fmt"
func main() {
    var umur int
    var kk bool

    fmt.Scan(&umur, &kk)
    if umur >= 17 && kk {
        fmt.Println("bisa membuat KTP")
    } else {
        fmt.Println("belum bisa membuat KTP")
    }
}
```

To the right of the editor is a terminal window showing the output of running the program. It shows three runs of the command 'go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\guided\1.go"'. The first run with an age of 17 and a family card (true) prints 'bisa membuat KTP'. The second run with an age of 20 and no family card (false) prints 'belum bisa membuat KTP'. The third run with an age of 15 and no family card also prints 'belum bisa membuat KTP'. Below the terminal is a status bar with information like 'Ln 3, Col 17 | 40 character Plain t | 100% | Wind UTF-8'.

Deskripsi program

Program ini berfungsi untuk menentukan apakah seorang penduduk bisa membuat KTP atau tidak.

1. Package main menandakan file ini adalah program utama.
2. Import ("fmt") untuk mengimpor package fmt yang menyediakan fungsi seperti Print, Println, Printf, Scan, dll.
3. funct main() { ... } adalah isi utama program, ketika program dijalankan maka fungsi main dipanggil.
4. Var umur int mendeklarasikan variabel tersebut integer.
5. Var kk bool mendeklarasikan variabel tersebut boolean.
6. Fmt.Scan(&umur, &kk) membaca input user dan menyimpan ke dalam variabel sesuai urutan.
7. Lalu ada program if yang mengecek apakah umur lebih dari sama dengan 17 dan mempunyai kartu keluarga (kk) jika keduanya memenuhi maka akan menjalankan program didalam if tersebut, jika tidak memenuhi di program if maka akan menjalankan program default di else.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var x string
    var huruf, hkecil, hkapital bool

    fmt.Scan(&x)

    huruf = (x >= "a" && x <= "z") || (x >= "A" && x <= "Z")
    hkecil = (x == "a" || x == "e" || x == "i" || x == "o" ||
    x == "u")
    hkapital = (x == "A" || x == "E" || x == "I" || x == "O" ||
    x == "U")

    if huruf && (hkecil || hkapital) {
        fmt.Println("vokal")
    } else if huruf && !(hkecil || hkapital) {
        fmt.Println("konsonan")
    } else {
        fmt.Println("bukan huruf")
    }
}
```

Screenshot program

The screenshot shows a code editor interface with a dark theme. On the left is a file tree for a directory named 'ALPRO_PRAKTIKUM'. Inside 'minggu10' are several files: 'laprak', 'minggu1', 'minggu2', 'minggu3', 'minggu4', 'minggu5', 'minggu6', 'minggu7(asessment)', 'minggu9', 'guided', 'unguided', and three numbered files ('1.go', '2.go', '3.go'). The file '2.go' is currently selected and open in the main editor area. The code in '2.go' is as follows:

```
package main
import "fmt"
func main() {
    var x string
    var huruf, hkecil, hkapital bool
    fmt.Scan(&x)

    huruf = (x >= "a" && x <= "z") || (x >= "A" && x <= "Z")
    hkecil = (x == "a" || x == "e" || x == "i" || x == "o" || x == "u")
    hkapital = (x == "A" || x == "E" || x == "I" || x == "O" || x == "U")

    if huruf && (hkecil || hkapital) {
        fmt.Println("vokal")
    } else if huruf && !(hkecil || hkapital) {
        fmt.Println("konsonan")
    } else {
        fmt.Println("bukan huruf")
    }
}
```

Below the editor is a terminal window showing the execution of the program:

```
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\guided\2.go"
A
vokal
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\guided\2.go"
f
konsonan
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\guided\2.go"
1
bukan huruf
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\guided\2.go"
$>
bukan huruf
PS D:\Tel-U\ALPRO_PRAKTIKUM>
```

The status bar at the bottom right indicates the date and time: 11/21/2025 4:52 PM.

Deskripsi program

Program ini berfungsi untuk menentukan suatu alfabet yang diberikan adalah vokal atau konsonan.

1. Package main menandakan file ini adalah program utama.
2. Import ("fmt") untuk mengimpor package fmt yang menyediakan fungsi seperti Print, Println, Printf, Scan, dll.
3. funct main() { ... } adalah isi utama program, ketika program dijalankan maka fungsi main dipanggil.
4. Var x string mendeklarasikan variabel tersebut dari tersebut adalah teks.
5. Var huruf, hkecil, hkapital bool mendeklarasikan variabel tersebut adalah boolean.
6. Fmt.Scan(&x) membaca input user lalu menyimpan kedalam variabel x.
7. Lalu ada variabel huruf, hkecil, hkapital yang masing masing variabel akan mengecek apakah x sudah memenuhi dari isi masing masing variabel tersebut.
8. Ada algoritma if yang pertama akan mengecek apakah huruf dan (hkecil/hkapital) bernilai true, jika iya maka akan menjalankan program yang berada pada if.
9. Lalu jika tidak memenuhi di if pertama maka akan dilanjut ke else if yang akan mengecek apakah huruf bernilai true dan (hkecil/hkapital bernilai false), jika memenuhi maka akan menjalankan program selanjutnya.
10. Jika tidak memenuhi di 2 kali pengecekan tadi maka akan menjalankan program default di else.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var angka int

    var teks string

    fmt.Scan(&angka)

    d1 := angka / 1000

    d2 := (angka / 100) % 10

    d3 := (angka / 10) % 10

    d4 := angka % 10

    if d1 > d2 && d2 > d3 && d3 > d4 {

        teks = "terurut mengecil"

    } else if d1 < d2 && d2 < d3 && d3 < d4 {

        teks = "terurut membesar"

    } else {

        teks = "tidak terurut"

    }

    fmt.Println("Digit pada bilangan", angka, teks)
```

```
}
```

Screenshot program

The screenshot shows a Go development environment with the following details:

- File Explorer:** Shows a project structure under "ALPRO_PRAKTIKUM". The "guided" folder contains "1.go", "2.go", and "3.go", with "3.go" currently selected.
- Code Editor:** Displays the content of "3.go". The code reads an integer from the user, then calculates and prints its digits. It uses division by 1000 to get the hundreds digit, and then checks if it's greater than or equal to 100. If so, it's a three-digit number. Otherwise, it checks if the tens digit is greater than the ones digit. If both conditions are met, it's a three-digit number where the hundreds digit is the largest. Otherwise, it's a two-digit number where the tens digit is the largest. If none of these conditions are met, it's a one-digit number.
- Terminal:** Shows the command "go run" being run on "3.go" with different inputs (2489, 3861, 9651) and the resulting output: "Digit pada bilangan 2489 terurut membesar", "Digit pada bilangan 3861 tidak terurut", and "Digit pada bilangan 9651 terurut mengecil".
- Output Window:** Shows the output of the program execution, which includes the input numbers and the corresponding classification ("terurut", "tidak terurut", "mengecil").

Deskripsi program

Program ini berfungsi untuk menentukan apakah digit dalam suatu bilangan terurut membesar, mengecil, atau tidak terurut.

1. Package main menandakan file ini adalah program utama.
2. Import ("fmt") untuk mengimpor package fmt yang menyediakan fungsi seperti Print, Println, Printf, Scan, dll.
3. funct main() { ... } adalah isi utama program, ketika program dijalankan maka fungsi main dipanggil.
4. Var angka int mendeklarasikan variabel tersebut integer.
5. Var teks string mendeklarasikan variabel tersebut teks.
6. Fmt.Scan(&angka) membaca input user dan menyimpan ke variabel tersebut.
7. Lalu ada d1, d2, d3, d4 yang akan memisahkan masing masing digit.
8. Lalu ada algoritma if pertama yang akan mengecek apakah terurut mengecil dengan membandingkan d1 lebih besar dari d2, d2 lebih besar dari d3, dan d3 lebih besar dari d4 jika memenuhi semua maka akan menjalankan program yang didalam if.
9. Jika di algoritma if pertama tidak terpenuhi maka akan di cek kembali disini apakah terurut membesar dengan membandingkan seperti tadi dan jika memenuhi maka akan menjalankan program yang ada pada if kedua. Lalu

jika tidak memenuhi kedua pengecekan kondisi sebelumnya maka akan menjalankan program default di else.

TUGAS

1. Tugas 1

Source code

```
package main

import ("fmt"
        "math")

func main() {
    var berat, biayaGram float64

    fmt.Scan(&berat)

    kilo := math.Floor(berat / 1000)
    gram := math.Mod(berat, 1000)

    if kilo > 10 {
        biayaGram = 0
    } else if gram >= 500 {
        biayaGram = gram * 5
    } else {
        biayaGram = gram * 15
    }

    biayaKilo := kilo * 10000
    total := biayaKilo + biayaGram

    fmt.Println("Berat Parsel (gram):", berat)
    fmt.Println("Detail Berat:", kilo, "kg +", gram, "gr")
    fmt.Println("Detail Biaya: Rp.", biayaKilo, "+ Rp", biayaGram)
    fmt.Println("Total Biaya: Rp.", total)
}
```

Screenshot program

The screenshot shows a code editor interface with a sidebar on the left containing project files like 'laprak', 'minggu1' through 'minggu9', and 'unguided'. The main area displays a Go program named '1.go' under the 'unguided' folder. The code calculates the cost of a parcel based on weight (berat) and gram (gram). It uses floating-point arithmetic and integer division to determine the cost per kilogram (biayaKilo) and total cost (total). The terminal below shows the execution of the program with two different input values (8500 and 9250), demonstrating the output of Berat Parcel, Detail Berat, Detail Biaya, and Total Biaya.

```
minggu10 > unguided > 1.go > main
package main

import ("fmt"
        "math")

func main() {
    var berat, biayaGram float64

    fmt.Scan(&berat)

    kilo := math.Floor(berat / 1000)
    gram := math.Mod(berat, 1000)

    if kilo > 10 {
        biayaGram = 8
    } else if gram > 500 {
        biayaGram = gram * 5
    } else {
        biayaGram = gram * 15
    }

    biayaKilo := kilo * 10000
    total := biayaKilo + biayaGram

    fmt.Println("Berat Parsel (gram):", berat)
    fmt.Println("Detail Berat: ", kilo, "kg +", gram, "gr")
    fmt.Println("Detail Biaya: Rp.", biayaKilo, "+ Rp.", biayaGram)
    fmt.Println("Total Biaya: Rp.", total)
}

PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\unguided\1.go"
8500
Berat Parsel (gram): 8500
Detail Berat: 8 kg + 500 gr
Detail Biaya: Rp. 80000 + Rp 2500
Total Biaya: Rp. 82500
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\unguided\1.go"
9250
Berat Parsel (gram): 9250
Detail Berat: 9 kg + 250 gr
Detail Biaya: Rp. 90000 + Rp 3750
Total Biaya: Rp. 93750
PS D:\Tel-U\ALPRO_PRAKTIKUM>
```

Deskripsi program

Program ini berfungsi untuk menghitung biaya pengiriman.

1. Package main menandakan file ini adalah program utama.
 2. Import (“fmt”) untuk mengimpor package fmt yang menyediakan fungsi seperti Print, Println, Printf, Scan, dll, Dan math untuk impor package yang menyediakan fungsi matematika.
 3. funct main() { ... } adalah isi utama program, ketika program dijalankan maka fungsi main dipanggil.
 4. Var berat, biayaGram float64 mendeklarasikan variabel tersebut bilangan real.
 5. Fmt.Scan(&berat) membaca input user dan menyimpan ke variabel tersebut.
 6. Lalu ada variabel kilo yang akan membagi nilai berat dengan 100 untuk mendapatkan nilai satuan kilo.
 7. Variabel gram akan menghitung nilai berat di modulus 1000 untuk mendapatkan nilai gram.
 8. Lalu ada algoritma if yang akan mengecek jika nilai kilo lebih dari 10 maka biayaGram akan di set 0 atau gratis dan menghentikan program if else.
 9. Selanjutnya jika tidak memenuhi di kondisi pertama maka akan mengecek jika gram lebih dari sama dengan 500 maka akan menghitung biayaGram dengan mengalikan jumlah gram dikali 5 dan menghentikan program if else.
 10. Jika tidak memenuhi pengecekan di kondisi kondisi sebelumnya maka akan menjalankan program default di else.

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 && nam <= 80 {

        nmk = "AB"

    } else if nam > 65 && nam <= 72.5 {

        nmk = "B"

    } else if nam > 57.5 && nam <= 65 {

        nmk = "BC"

    } else if nam > 50 && nam <= 57.5 {

        nmk = "C"

    } else if nam > 40 && nam <= 50 {

        nmk = "D"

    } else {

        nmk = "E"

    }
}
```

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

Screenshot program

The screenshot shows a Visual Studio Code interface with two main panes. The left pane displays a Go program named `2.go` which calculates a grade based on a score. The right pane shows the terminal output of running the program.

```
File Edit Selection View Go Run ... ← → Q ALPRO_PRAKTIKUM
```

```
minggu10 > unguided > 2.go > main
```

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

```
109082500152
S1IF-13-04
Rafli Nurhidayat
```

```
Ln 3, Col 17 | 40 character Plain t | 100% | Wind UTF-8
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\unguided\2.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\unguided\2.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\unguided\2.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS D:\Tel-U\ALPRO_PRAKTIKUM> ]
```

Deskripsi program

Program ini berfungsi untuk menentukan NMK.

1. Package main menandakan file ini adalah program utama.
 2. Import ("fmt") untuk mengimpor package fmt yang menyediakan fungsi seperti Print, Println, Printf, Scan, dll.
 3. funct main() { ... } adalah isi utama program, ketika program dijalankan maka fungsi main dipanggil.
 4. Var nam float64 mendeklarasikan variabel tersebut bilangan real.
 5. Var nmk string string mendeklarasikan variabel tersebut text.
 6. Fmt.Scan akan membaca input user dan menyimpan ke variabel tersebut.
 7. Lalu ada algoritma pengecekan pertama yang akan mengecek jika nilai nam lebih dari 80 maka nmk = A, lalu lanjut ke pengecekan selanjutnya jika tidak memenuhi di pengecekan sebelumnya, jika nam lebih dari 72.5 dan kurang dari sama dengan 80 maka nmk = AB, lalu lanjut ke pengecekan selanjutnya jika tidak memenuhi di pengecekan sebelumnya, yaitu jika nam lebih dari 65 dan kurang dari sama dengan 72.5 maka nmk = B, lalu lanjut ke pengecekan selanjutnya jika tidak memenuhi di pengecekan sebelumnya, yaitu jika nam lebih dari 57.5 dan kurang dari sama dengan 65 maka nmk = BC, lalu lanjut ke pengecekan selanjutnya jika tidak memenuhi di pengecekan sebelumnya, yaitu jika nam lebih dari 50 dan

kurang dari sama dengan 57.5 maka nmk = C, lalu lanjut ke pengecekan selanjutnya jika tidak memenuhi di pengecekan sebelumnya, yaitu jika nam lebih dari 40 dan kurang dari sama dengan 50 maka nmk = D, dan jika tidak memenuhi seluruh pengecekan kondisi maka nilai nmk default = E.

Jawaban Pertanyaan:

1. Error karna nilai nmk belum terdefinisi. Eksekusi program tidak sesuai spesifikasi soal.
2. Kesalahannya adalah seharusnya variabel yang didalam if adalah nmk bukan nam, ini membuat variabel nmk tidak terdefinisi.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var b, faktor int
    var prima bool

    fmt.Scan(&b)

    fmt.Println("Bilangan: ", b)
    fmt.Print("Faktor: ")

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

    if faktor == 2 {
        fmt.Println("\nPrima:", prima)
    } else {
        fmt.Println("\nPrima:", !prima)
    }
}
```

Screenshot program

The screenshot shows a Go code editor interface. On the left is a file tree for a directory named 'ALPRO_PRAKTIKUM' containing various Go files and sub-directories like 'minggu10', 'guided', and 'unguided'. The current file is '3.go' under the 'unguided' folder, which is selected and highlighted in blue. The main pane displays the Go code:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var b, faktor int
7     var prima bool
8
9     fmt.Scan(&b)
10
11    fmt.Println("Bilangan: ", b)
12    fmt.Print("Faktor: ")
13
14    for i := 1; i <= b; i++ {
15        if b % i == 0 {
16            fmt.Print(i, " ")
17            faktor++
18        }
19
20    }
21
22    if faktor == 2 {
23        fmt.Println("\nPrima:", prima)
24    } else {
25        fmt.Println("\nPrima:", !prima)
26    }
}
```

To the right of the code editor is a terminal window showing the output of running the program. It prints the number 12, its factors (1 2 3 4 6 12), and then asks if 12 is prime. The user inputs '7' and the program outputs that 7 is prime and not equal to 12.

```
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\unguided\3.go"
12
Bilangan: 12
Faktor: 1 2 3 4 6 12
Prima: true
PS D:\Tel-U\ALPRO_PRAKTIKUM> go run "d:\Tel-U\ALPRO_PRAKTIKUM\minggu10\unguided\3.go"
7
Bilangan: 7
Faktor: 1 7
Prima: false
PS D:\Tel-U\ALPRO_PRAKTIKUM>
```

Deskripsi program

Program ini berfungsi untuk mencari dan menampilkan semua faktor bilangan dan menentukan prima.

1. Package main menandakan file ini adalah program utama.
2. Import ("fmt") untuk mengimpor package fmt yang menyediakan fungsi seperti Print, Println, Printf, Scan, dll.
3. funct main() { ... } adalah isi utama program, ketika program dijalankan maka fungsi main dipanggil.
4. Didalam program tersebut ada fungsi perulangan yang berfungsi mengecek satu per satu angka dari 1 sampai bilangan yang diinputkan dengan cara di modulus, jika hasil dari modulus tersebut 0 maka akan print nilai i dan menambahkan 1 ke nilai variabel prima.
5. Lalu ada algoritma if yang akan mengecek apakah faktor tepat sama dengan 2 jika iya maka true, dan jika tidak sama dengan 2 maka false.