

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 09

IF-THEN



Disusun oleh:

FAREL JULIYANDRA RESTU HERMAWAN

109082530038

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 bilangan int

    fmt.Scan(&bilangan)

    if bilangan < 0 {

        bilangan = -bilangan

    }

    fmt.Println(bilangan)

}
```

Screenshoot program

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

- Code Editor:** Displays the source code of the program.
- Terminal:** Shows the command `go run` being used to run the program multiple times, with the output showing the input value and its negation.
- Output Window:** Shows the results of the program execution, including the input values 3, 5, 5, 0, -2000, and 9999, and their corresponding outputs -3, -5, -5, 0, 2000, and -9999.

Deskripsi program

Program Go di atas dimulai dengan mendeklarasikan variabel bilangan untuk menampung input dari pengguna. Setelah itu, program membaca nilai yang dimasukkan melalui `fmt.Scan(&bilangan)`. Selanjutnya, program melakukan pengecekan menggunakan `if bilangan`

< 0, yaitu untuk mengetahui apakah bilangan tersebut bernilai negatif. Jika benar negatif, maka bilangan diubah menjadi positif dengan membuatnya bernilai kebalikannya melalui bilangan = -bilangan. Jika tidak negatif, nilainya tetap. Terakhir, program mencetak hasil akhir bilangan menggunakan fmt.Println(bilangan). Dengan alur tersebut, program memastikan bahwa output yang ditampilkan selalu dalam bentuk bilangan positif.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var bilangan int

    var teks string

    fmt.Scan(&bilangan)

    teks = "bukan positif"

    if bilangan > 0 {

        teks = "positif"

    }

    fmt.Println(teks)

}
```

Screenshoot program

The screenshot shows a Go development environment with three panes. The left pane displays the file structure under 'MODULES' with files 'contoh soal 1.go', 'contoh soal 2.go', and 'contoh soal 3.go'. The middle pane shows the content of 'contoh soal 3.go': package main; import "fmt"; func main() { var bilangan int; var teks string; fmt.Scan(&bilangan); teks = "bukan positif"; if bilangan > 0 { teks = "positif"; } fmt.Println(teks) }. The right pane shows a terminal window with the command 'go run F:\VALPRO\modul\kontoh soal 3.go' and its output: '30 positif'. Below the terminal is a code editor window with the same code as the middle pane, and its status bar shows 'Line 8, Col 17 24 character (empty) 0m0s - UTF-8'.

Deskripsi program

Program Go di atas dimulai dengan mendeklarasikan variabel bilangan sebagai input angka dan teks sebagai penampung hasil berupa string. Setelah pengguna memasukkan sebuah bilangan melalui `fmt.Scan(&bilangan)`, program langsung memberikan nilai awal untuk variabel teks, yaitu "bukan positif". Nilai awal ini akan menjadi keluaran default jika bilangan tidak lebih dari nol. Program kemudian melakukan pengecekan menggunakan kondisi `if bilangan > 0`; jika syarat ini terpenuhi, artinya bilangan tersebut positif, maka nilai variabel teks diubah menjadi "positif". Terakhir, program menampilkan isi variabel teks ke layar, sehingga output yang muncul sesuai dengan hasil evaluasi bilangan yang dimasukkan pengguna.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var bilangan int

    var hasil bool

    fmt.Scan(&bilangan)

    hasil = bilangan % 2 == 0 && bilangan < 0

    fmt.Println(hasil)

}
```

Screenshot program

The screenshot shows a Go development environment with three tabs in the code editor: `contoh soal 1.go`, `contoh soal 2.go`, and `contoh soal 3.go`. The current file is `contoh soal 3.go`, which contains the following code:

```
package main
import "fmt"
func main() {
    var bilangan int
    var hasil bool
    fmt.Scan(&bilangan)
    hasil = bilangan % 2 == 0 && bilangan < 0
    fmt.Println(hasil)
}
```

To the right of the code editor is a terminal window showing the execution of the program and its output:

```
PS F:\UINPRO\modul_9\go\mas\11\UINPRO\modul\pertemuan\soal_3.go
$3
False
PS F:\UINPRO\modul_9\go\mas\11\UINPRO\modul\pertemuan\soal_3.go
$3
False
PS F:\UINPRO\modul_9\go\mas\11\UINPRO\modul\pertemuan\soal_3.go
$4
true
PS F:\UINPRO\modul_9\go\mas\11\UINPRO\modul\pertemuan\soal_3.go
$5
false
PS F:\UINPRO\modul_9\go\mas\11\UINPRO\modul\pertemuan\soal_3.go
$2
true
PS F:\UINPRO\modul_9\go\mas\11\UINPRO\modul\pertemuan\soal_3.go
```

Below the terminal is a small window titled "NAMA:" containing student information:

NAMA :	FAREL JULIYANDRA RESTU HERMANA
NIM :	189002530018
KELAS :	IF-13-04

Deskripsi program

Program Go di atas dimulai dengan mendeklarasikan variabel bilangan untuk menyimpan input angka dan hasil untuk menampung nilai boolean. Setelah pengguna memasukkan sebuah bilangan melalui `fmt.Scan(&bilangan)`, program langsung mengevaluasi dua kondisi sekaligus, yaitu apakah bilangan tersebut genap (`bilangan % 2 == 0`) dan apakah bilangan tersebut negatif (`bilangan < 0`). Kedua kondisi ini digabungkan dengan operator logika `&&`, sehingga hasil akan bernilai true hanya jika bilangan memenuhi kedua syarat tersebut. Jika salah satu kondisi tidak terpenuhi, maka hasil otomatis bernilai false. Pada bagian akhir, program mencetak nilai boolean tersebut ke layar, sehingga pengguna dapat mengetahui apakah bilangan yang dimasukkan merupakan bilangan genap negatif atau tidak.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var n, m, s int
    fmt.Print("Masukan :")
```

```

fmt.Scan(&n)

m = n / 2

s = n % 2

if s != 0 {

    m++

}

fmt.Println("Keluaran : ",m)
}

```

Screenshot program

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

- File Explorer:** Shows a file tree with modules named "controlawal1", "controlawal2", and "controlawal3".
- Code Editor:** Displays the source code of the "main" package's "main.go" file.
- Terminal:** Shows the command-line interface where the program is run, displaying its execution and output.
- Output Window:** Shows the results of the program's execution, including input values and calculated outputs.
- Run Window:** A separate window titled "NAME : FAREL JULIYANDRA RESTU HERMANN" showing personal information.

Deskripsi program

Program Go di atas digunakan untuk menghitung jumlah motor yang diperlukan berdasarkan jumlah orang yang akan berangkat. Pertama, program mendeklarasikan variabel orang untuk menampung input jumlah orang dari pengguna. Setelah nilai dibaca melalui `fmt.Scan`, program menghitung jumlah motor awal dengan membagi jumlah orang per dua, karena satu motor dapat menampung dua orang. Selanjutnya, program melakukan pengecekan menggunakan kondisi `if orang%2 != 0` untuk mengetahui apakah jumlah orang bernilai ganjil; jika ganjil, berarti masih ada satu orang yang belum mendapat motor sehingga variabel motor ditambah satu. Terakhir, program menampilkan jumlah motor yang harus disiapkan sesuai perhitungan tersebut.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var x int
    var teks string
    fmt.Print("masukan :")
    fmt.Scan(&x)

    teks = "bukan"
    if x < 0 {
        x = x / 2
        teks = "genap negatif"
    }
    fmt.Print(teks)
}
```

Screenshoot program

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

- Left Sidebar (File Explorer):** Shows files in the current directory:
 - kontoh.go
 - kontoh2.go
 - kontoh3.go
 - kontoh4.go
 - kontoh5.go
- Code Editor:** The file "kontoh5.go" is open, containing the following Go code:

```
package main

import "fmt"

func main() {
    var x int
    var teks string
    fmt.Println("masukan :")
    fmt.Scan(&x)

    teks = "baik"
    if x < 0 {
        x = x / 2
        teks = "guru negatif"
    }
    fmt.Println(teks)
}
```
- Terminal:** A terminal window on the right shows the output of running the program:

```
PS F:\VALPRO\modul 0> go run "F:\VALPRO\modul\kontoh5.go"
masukan :10
baik
PS F:\VALPRO\modul 0> go run "F:\VALPRO\modul\kontoh5.go"
masukan :14
guru negatif
PS F:\VALPRO\modul 0> go run "F:\VALPRO\modul\kontoh5.go"
masukan :12
guru negatif
PS F:\VALPRO\modul 0>
```
- Bottom Status Bar:** Shows file path (kontoh5.go), character count (74), line number (106), word count (1), and encoding (UTF-8).

Deskripsi program

Program Go di atas berfungsi untuk mengecek apakah sebuah bilangan termasuk kategori "genap negatif". Program dimulai dengan mendeklarasikan variabel n untuk menampung input dari pengguna. Setelah nilai bilangan dimasukkan melalui fmt.Scan(&n), program masuk ke proses pengecekan menggunakan kondisi if $n < 0 \ \&\& \ n \% 2 == 0$. Kondisi ini memeriksa dua hal sekaligus: apakah bilangan tersebut bernilai negatif ($n < 0$) dan apakah bilangan tersebut genap ($n \% 2 == 0$). Jika kedua syarat terpenuhi, program mencetak "genap negatif". Namun, jika salah satu atau kedua kondisi tidak terpenuhi, bagian else akan dijalankan dan program mencetak "bukan". Dengan cara ini, program memastikan output sesuai dengan karakteristik bilangan yang dimasukkan pengguna.

3. Tugas 3

Source code

```
package main

import "fmt"

func main (){
    var x, y, a,b int
    fmt.Print("Masukan :")
    fmt.Scan(&x,&y)

    a = y % x
    b = x % y
```

```

if a == 0{
    fmt.Println("true")
}else {
    fmt.Println("false")
}

if b == 0 {
    fmt.Println("true")
}else {
    fmt.Println("false")
}
}

```

Screenshoot program

The screenshot shows a Go development environment with several tabs open. The main tab contains the following Go code:

```

package main
import "fmt"
func main() {
    var x, y, a, b int
    fmt.Println("Masukan x:")
    fmt.Scan(&x, &y)

    a = y % x
    b = x % y

    if a == 0{
        fmt.Println("true")
    }else {
        fmt.Println("false")
    }

    if b == 0 {
        fmt.Println("true")
    }else {
        fmt.Println("false")
    }
}

```

To the right of the code, there are two terminal windows showing the execution of the program. The first terminal shows the user inputting values for x and y, and the program outputting 'false' for both cases. The second terminal shows the user inputting different values, and the program outputting 'true' for both cases.

Deskripsi program

Program Go di atas digunakan untuk mengecek hubungan faktor antara dua bilangan, yaitu x dan y, yang dimasukkan oleh pengguna. Setelah kedua nilai dibaca melalui `fmt.Scan(&x, &y)`, program melakukan pengecekan pertama dengan kondisi `if y%x == 0` untuk menentukan apakah x merupakan faktor dari y; jika sisa pembagian y oleh x adalah nol, program mencetak true, jika tidak maka mencetak false. Selanjutnya, program melakukan pengecekan kebalikan menggunakan kondisi `if x%y == 0` untuk mengetahui apakah y merupakan faktor dari x. Sama seperti sebelumnya, jika hasil pembagian tidak bersisa maka program menampilkan true, dan jika bersisa maka menampilkan false. Dengan dua pengecekan ini, program memberikan informasi apakah kedua angka tersebut saling menjadi faktor satu sama lain atau tidak.