

**LAPORAN PRAKTIKUM ALGORITMA**

**DAN PEMROGRAMAN 1**

**MODUL 9**

**IF - THEN**



**Disusun oleh:**

**FERDINAND AXEL VALERIAN**

**109082500154**

**S1IF-13-07**

**Asisten Praktikum**

Adithana dharma putra

Apri pandu wicaksono

**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)
}
```

#### Screenshot program

```
g1.go > package main
1 import "fmt"
2 func main() {
3     var bilangan int
4     fmt.Scan(&bilangan)
5     if bilangan < 0 {
6         bilangan = -bilangan
7     }
8     fmt.Println(bilangan)
9
10 }
```

109082500154  
S1F-13-07  
Ferdinand Axel Valerian|

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\vscode\College\Modul 9> go run g1.go
10
10
PS D:\vscode\College\Modul 9> go run g1.go
-3
3
PS D:\vscode\College\Modul 9> go run g1.go
5
5
PS D:\vscode\College\Modul 9> go run g1.go
0
0
PS D:\vscode\College\Modul 9> go run g1.go
-9999
9999
PS D:\vscode\College\Modul 9> []
```

#### Deskripsi program

Program ini berfungsi untuk menghasilkan nilai mutlak dari bilangan yang dimasukkan. Ketika pengguna memberikan input sebuah angka, program akan mengonversinya menjadi nilai positif jika angka tersebut bernilai negatif. Sebaliknya, jika angka input

sudah bernilai positif atau nol, program akan mempertahankan nilai aslinya. Hasil akhir kemudian ditampilkan kepada pengguna.

## 2. Guided 2

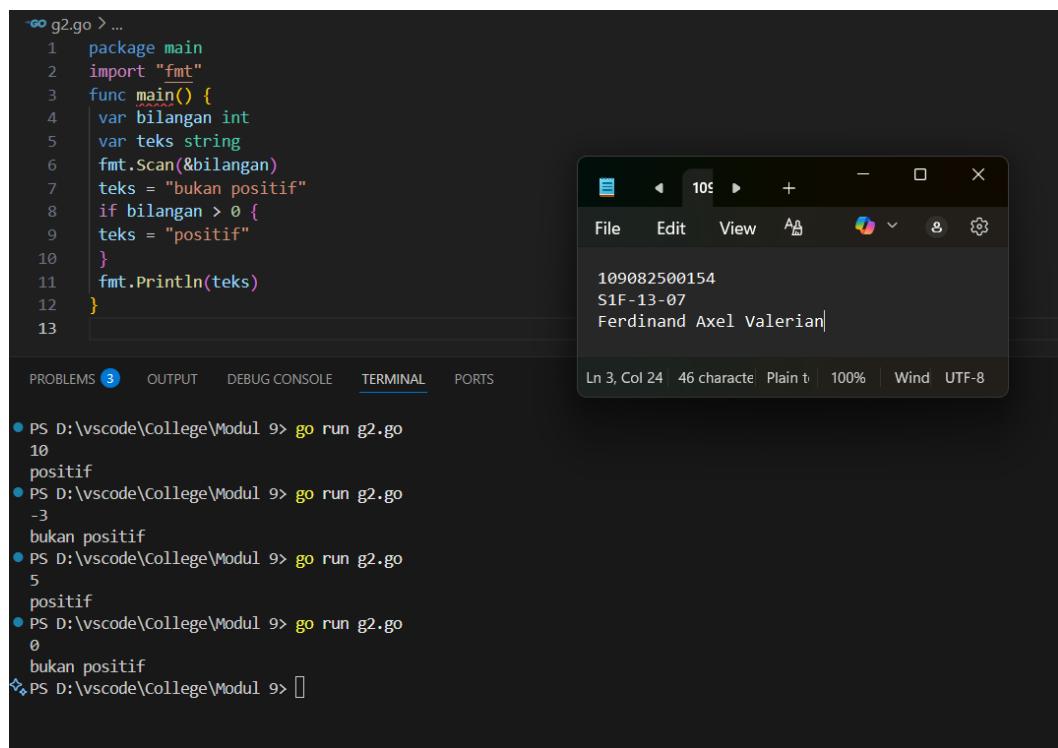
### Source Code

```
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 terminal window with the following content:

```
-∞ g2.go > ...
1 package main
2 import "fmt"
3 func main() {
4     var bilangan int
5     var teks string
6     fmt.Scan(&bilangan)
7     teks = "bukan positif"
8     if bilangan > 0 {
9         teks = "positif"
10    }
11    fmt.Println(teks)
12 }
```

The terminal window has a dark theme and includes a status bar at the bottom with tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. The status bar also shows file statistics: Ln 3, Col 24 | 46 characters | Plain text | 100% | Wind | UTF-8.

Below the terminal window, there is a command-line history:

- PS D:\vscode\College\Modul 9> go run g2.go
- 10
- positif
- PS D:\vscode\College\Modul 9> go run g2.go
- 3
- bukan positif
- PS D:\vscode\College\Modul 9> go run g2.go
- 5
- positif
- PS D:\vscode\College\Modul 9> go run g2.go
- 0
- bukan positif
- PS D:\vscode\College\Modul 9> []

## Deskripsi program

Program ini berfungsi untuk mengidentifikasi jenis bilangan yang dimasukkan pengguna. Jika bilangan yang diberikan lebih besar dari nol, program akan menampilkan teks "positif". Namun jika bilangan tersebut nol atau negatif, program akan menampilkan teks "bukan positif". Dengan demikian, pengguna dapat dengan mudah mengetahui status bilangan yang dimasukkannya.

### 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 the VS Code interface with a Go file named g2.go open in the editor. The code defines a main function that reads an integer from standard input, checks if it's positive or not, and prints the result as a string. The terminal below shows the execution of the program with different inputs, demonstrating the expected behavior.

```
g2.go > ...
1 package main
2 import "fmt"
3 func main() {
4     var bilangan int
5     var teks string
6     fmt.Scanf(&bilangan)
7     teks = "bukan positif"
8     if bilangan > 0 {
9         teks = "positif"
10    }
11 }
12 fmt.Println(teks)
13 }
```

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
109082500154
S1F-13-07
Ferdinand Axel Valerian
```

Ln 3, Col 24 | 46 character Plain text 100% Wind UTF-8

- PS D:\vscode\College\Modul 9> go run g3.go  
10  
false
- PS D:\vscode\College\Modul 9> go run g3.go  
-3  
false
- PS D:\vscode\College\Modul 9> go run g3.go  
-4  
true
- PS D:\vscode\College\Modul 9> go run g3.go  
0  
false
- PS D:\vscode\College\Modul 9> go run g3.go  
-2  
true

PS D:\vscode\college\Modul 9> []

## Deskripsi program

Program ini memeriksa apakah bilangan yang dimasukkan merupakan bilangan genap negatif. Jika pengguna memasukkan angka seperti -4 atau -2, program akan menampilkan "true". Untuk angka-angka lainnya seperti bilangan positif, nol, atau bilangan ganjil negatif, program akan menampilkan "false".

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

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

    motor := (orang + 1) / 2
    fmt.Println(motor)
}
```

#### Screenshot program

The screenshot shows a terminal window with the following content:

```
-eo t1.go > package main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var orang int
7      fmt.Scan(&orang)
8
9      motor := (orang + 1) / 2
10     fmt.Println(motor)
11 }
```

Below the code, there is a terminal history:

- PS D:\vscode\College\Modul 9> go run t1.go
- 10
- 5
- PS D:\vscode\College\Modul 9> go run t1.go
- 1
- 1
- PS D:\vscode\College\Modul 9> go run t1.go
- 25
- 13
- PS D:\vscode\College\Modul 9> go run t1.go
- 9
- 5
- PS D:\vscode\College\Modul 9>

To the right of the terminal, a small window displays the output of the last command:

File	Edit	View	A <sub>d</sub>	undo	redo	Plain t	100%	Wind	UTF-8
109082500154									
S1F-13-07									
Ferdinand Axel Valerian									

## Deskripsi program

Program ini menghitung jumlah motor yang dibutuhkan berdasarkan jumlah orang. Karena satu motor bisa membawa 2 orang (satu pengemudi dan satu penumpang), program menggunakan rumus  $(\text{orang} + 1) / 2$  untuk memastikan pembulatan ke atas. Misalnya untuk 10 orang butuh 5 motor, untuk 1 orang butuh 1 motor, dan untuk 9 orang butuh 5 motor.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

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

    if bilangan < 0 && bilangan % 2 == 0 {
        fmt.Println("genap negatif")
    } else {
        fmt.Println("bukan")
    }
}
```

### Screenshot program

The screenshot shows a terminal window with the following content:

```
-∞ t2.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan int
7     fmt.Scan(&bilangan)
8
9     if bilangan < 0 && bilangan % 2 == 0 {
10        fmt.Println("genap negatif")
11    } else {
12        fmt.Println("bukan")
13    }
14 }
```

Below the terminal, there is a list of command-line interactions:

- PS D:\vscode\College\Modul 9> go run t2.go  
10  
bukan
- PS D:\vscode\College\Modul 9> go run t2.go  
-4  
genap negatif
- PS D:\vscode\College\Modul 9> go run t2.go  
0  
bukan
- PS D:\vscode\College\Modul 9> go run t2.go  
-2  
genap negatif
- PS D:\vscode\College\Modul 9> []

## Deskripsi program

Program ini dirancang untuk mengidentifikasi apakah suatu bilangan merupakan bilangan genap negatif. Program akan memeriksa dua kondisi: apakah bilangan tersebut kurang dari nol (negatif) dan apakah bilangan tersebut habis dibagi dua (genap). Jika kedua kondisi tersebut terpenuhi, program akan menampilkan "genap negatif". Untuk semua kondisi lainnya, termasuk bilangan positif, nol, atau bilangan ganjil negatif, program akan menampilkan "bukan".

### 3. Tugas 3

#### Source code

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Scan(&x, &y)

    faktorXY := y % x == 0
    faktorYX := x % y == 0

    fmt.Println(faktorXY)
    fmt.Println(faktorYX)
}
```

#### Screenshot program

```
PS D:\vscode\College\Modul 9> go run t3.go
10 5
false
true
PS D:\vscode\College\Modul 9> go run t3.go
3 21
true
false
PS D:\vscode\College\Modul 9> go run t3.go
4 4
true
true
PS D:\vscode\College\Modul 9>
```

### **Deskripsi program**

Program ini digunakan untuk menentukan hubungan faktor antara dua bilangan.

Program menerima dua bilangan bulat positif sebagai input, kemudian memeriksa apakah bilangan pertama merupakan faktor dari bilangan kedua, dan sebaliknya.

Hasil output berupa dua nilai boolean: baris pertama menunjukkan apakah x adalah faktor dari y (dicek dengan  $y \% x == 0$ ), sedangkan baris kedua menunjukkan apakah y adalah faktor dari x (dicek dengan  $x \% y == 0$ ).