

**LAPORAN PRAKTIKUM ALGORITMA**

**DAN PEMROGRAMAN 1**

**MODUL 9**

**IF-THEN**



**Disusun oleh:**

**Anindya Rahadita Yumnaa**

**109082500138**

**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

The screenshot shows a Visual Studio Code interface with several tabs open in the background. In the foreground, a terminal window displays the command `go run "c:/Users/MSI/Documents/VSCode/MINGGU 9/LAPRAK/guided 1/guided 1.go"` being run three times, with the output showing the number 9999 each time. A separate window titled "LightClock" is open, showing the text "109082500138", "SIIF-13-07", and "ANINDYA RAHADITA YUMNAA".

```
File Edit Selection View Go Run ... ⏪ ⏩ Search
C: > Users > MSI > Documents > VSCode > MINGGU 9 > LAPRAK > guided 1 > go guided 1.go ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan int
7     fmt.Scan(&bilangan)
8     if bilangan < 0 {
9         bilangan = -bilangan
10    }
11    fmt.Println(bilangan)
12 }

LightClock
File Edit View
109082500138
SIIF-13-07
ANINDYA RAHADITA YUMNAA

Ln 3, Col 24 48 character Plain t 100% Wind UTF-8

PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\MSI> go run "c:/Users/MSI/Documents/VSCode/MINGGU 9/LAPRAK/guided 1/guided 1.go"
-9999
9999
PS C:\Users\MSI> go run "c:/Users/MSI/Documents/VSCode/MINGGU 9/LAPRAK/guided 1/guided 1.go"
10
10
PS C:\Users\MSI> go run "c:/Users/MSI/Documents/VSCode/MINGGU 9/LAPRAK/guided 1/guided 1.go"
0
0
PS C:\Users\MSI> []
```

## Deskripsi program

### 1. package main

Menandakan bahwa file ini adalah program yang dapat dieksekusi secara langsung (bukan *library*).

### 2. import "fmt"

Mengimpor paket standar Go **fmt** (format). Paket ini diperlukan untuk melakukan operasi I/O (Input/Output) dasar, seperti membaca dari input (`fmt.Scan`) dan menulis ke layar (`fmt.Println`).

### 3. func main() {

Mendefinisikan **func main()** yang merupakan titik masuk (*entry point*) program. Eksekusi kode dimulai dari sini.

### 4. var bilangan int

Mendeklarasikan sebuah variabel bernama **bilangan** dengan tipe data **int** (integer/bilangan bulat). Variabel ini akan digunakan untuk menampung nilai yang dimasukkan pengguna.

### 5. fmt.Scan(&bilangan)

Membaca input dari pengguna (dari konsol) dan menyimpannya ke dalam variabel **bilangan**. Tanda ampersand (&) diperlukan karena kita perlu memberikan **alamat memori** dari variabel tersebut agar Scan dapat mengubah nilainya.

### 6. if bilangan < 0 {

Memulai blok kondisi **if**. Program akan mengeksekusi kode di dalamnya jika nilai variabel **bilangan** lebih kecil dari **0** (yaitu, jika *input* adalah bilangan negatif).

### 7. bilangan = -bilangan

Jika kondisi di atas terpenuhi (bilangan negatif), nilai dari **bilangan** diubah menjadi **negatif dari dirinya sendiri**. Secara efektif, ini mengubah bilangan negatif menjadi nilai absolutnya

### 8. }

Mengakhiri blok kode untuk kondisi **if**.

### 9. fmt.Println(bilangan)

Mencetak nilai akhir dari variabel **bilangan** ke layar (konsol), diikuti dengan baris baru. Jika input negatif, yang dicetak adalah nilai absolutnya; jika input positif/nol, yang dicetak adalah nilai aslinya.

**10. }**

Mengakhiri definisi dari fungsi **main()**.

## 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)
}
```

**Screenshot program**

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan int
7     var teks string
8     fmt.Scan(&bilangan)
9     teks = "bukan positif"
10    if bilangan > 0 {
11        teks = "positif"
12    }
13    fmt.Println(teks)
14 }
```

LightClock \* File Edit View 109082500138 S1IF-13-07 ANINDYA RAHADITA YUMNAA

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\guided 2\guided 2.go"
-3
bukan positif
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\guided 2\guided 2.go"
5
positif
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\guided 2\guided 2.go"
-17
bukan positif
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\guided 2\guided 2.go"
11
positif
PS C:\Users\MSI>
```

## Deskripsi program

### 1. package main

Menandakan bahwa file ini adalah program yang dapat dieksekusi secara langsung (bukan *library*).

### 2. import "fmt"

Mengimpor paket standar Go **fmt** (format). Paket ini diperlukan untuk melakukan operasi **I/O** (Input/Output) dasar, seperti membaca dari input (`fmt.Scan`) dan menulis ke layar (`fmt.Println`).

### 3. func main() {

`func main()` adalah **titik masuk** program, kode di dalam `func main()` dijalankan pertama kali setelah kode atau program diberi nama. Bentuknya harus **“func main()”** tanpa parameter dan tanpa nilai balik.

### 4. var bilangan int

Mendeklarasikan variabel bernama **bilangan** dengan tipe data **int** (bilangan bulat). Variabel ini akan menyimpan input angka dari pengguna.

### 5. var teks string

Mendeklarasikan variabel bernama **teks** dengan tipe data **string** (teks). Variabel ini akan menyimpan pesan hasil ("positif" atau "bukan positif").

### 6. fmt.Scan(&bilangan)

Membaca input dari pengguna (konsol) dan menyimpannya ke variabel **bilangan**. Tanda & (alamat memori) diperlukan agar fungsi Scan dapat mengubah nilai bilangan.

#### **7. `teks = "bukan positif"`**

Memberikan nilai awal "**bukan positif**" kepada variabel **teks**. Nilai ini akan dipertahankan jika bilangan adalah nol atau negatif.

#### **8. `if bilangan > 0 {`**

Memulai blok kondisi **if**. Program akan mengeksekusi kode di dalamnya hanya jika nilai variabel **bilangan lebih besar dari 0** (yaitu, bilangan positif).

#### **9. `teks = "positif"`**

Jika kondisi di atas terpenuhi (yaitu, bilangan adalah positif), nilai variabel **teks** akan diubah dari "bukan positif" menjadi "**positif**".

#### **10. `}`**

Mengakhiri blok kode untuk kondisi **if**.

#### **11. `fmt.Println(teks)`**

Mencetak nilai akhir dari variabel **teks** ke layar (konsol). Hasilnya adalah "positif" atau "bukan positif", tergantung input.

#### **12.**

Mengakhiri definisi dari fungsi **main()** (asumsi ini adalah bagian dari **func main()**).

### **3. Guided 3**

#### **Source Code**

```
package main

import "fmt"

func main() {
    var bilangan int
    var hasil bool
```

```

        fmt.Scan(&bilangan)

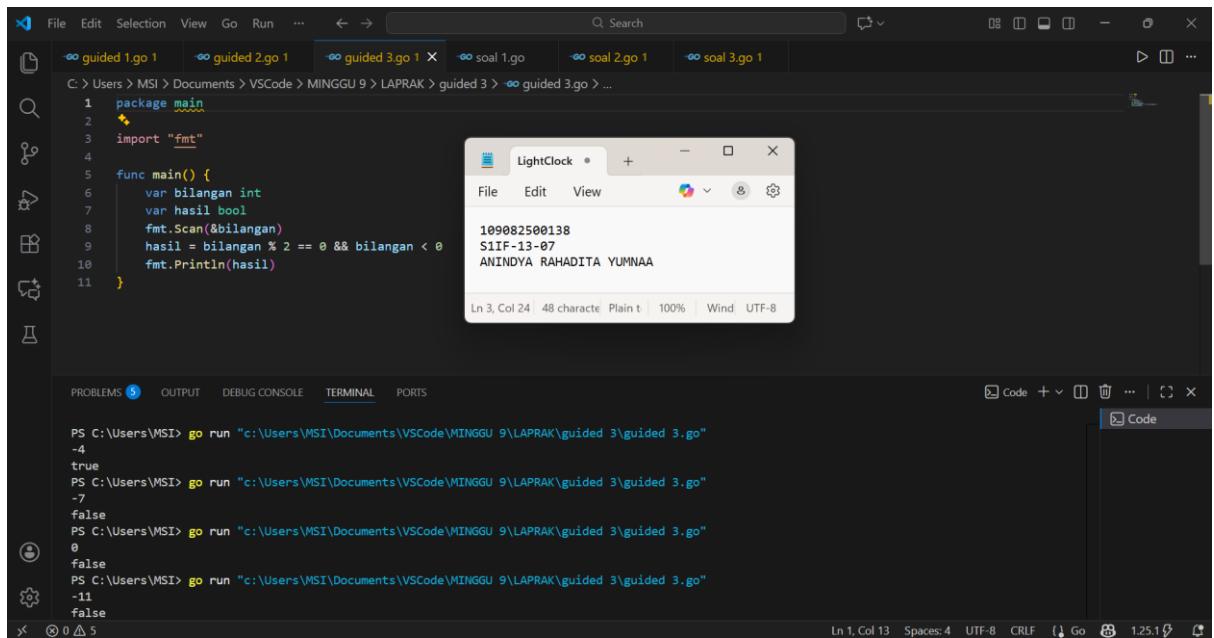
        hasil = bilangan % 2 == 0 && bilangan < 0

        fmt.Println(hasil)

    }

```

## Screenshot program



## Deskripsi program

### 1. package main

Menandakan bahwa file ini adalah program yang dapat dieksekusi secara langsung (bukan *library*).

### 2. import "fmt"

Mengimpor paket standar Go **fmt** (format). Paket ini diperlukan untuk melakukan operasi **I/O** (Input/Output) dasar, seperti membaca dari input (`fmt.Scan`) dan menulis ke layar (`fmt.Println`).

### 3. func main() {

`func main()` adalah **titik masuk** program, kode di dalam `func main()` dijalankan pertama kali setelah kode atau program diberi nama. Bentuknya harus **“func main()”** tanpa parameter dan tanpa nilai balik.

### 4. var bilangan int

Mendeklarasikan variabel bernama **bilangan** dengan tipe data **int** (bilangan bulat). Variabel ini akan menyimpan input angka dari pengguna.

### 5. var hasil bool

Mendeklarasikan variabel bernama **hasil** dengan tipe data **bool** (boolean, hanya bisa bernilai true atau false). Variabel ini akan menyimpan hasil evaluasi dari ekspresi logika di baris 9.

### 6. fmt.Scan(&bilangan)

Membaca input dari pengguna (konsol) dan menyimpannya ke variabel **bilangan**. Tanda **&** (alamat memori) diperlukan agar fungsi Scan dapat mengubah nilai bilangan.

### 7. hasil = bilangan % 2 == 0 && bilangan < 0

Mengevaluasi ekspresi logika gabungan dan menetapkan hasilnya ke variabel **hasil**. Ekspresi ini memiliki dua bagian yang dihubungkan oleh operator **AND (&&)**:

1. **bilangan % 2 == 0**: Memeriksa apakah bilangan **genap**. Operator modulo (%) mengembalikan sisa bagi; jika sisa bagi dengan 2 adalah 0, maka bilangan tersebut genap.
2. **bilangan < 0**: Memeriksa apakah bilangan **negatif**. Nilai hasil akan menjadi **true** jika **kedua** kondisi di atas terpenuhi (bilangan genap **DAN** negatif), selain itu akan menjadi **false**.

### 8. fmt.Println(hasil)

Mencetak nilai akhir dari variabel **hasil** (true atau false) ke layar (konsol).

### 9. }

Mengakhiri fungsi tempat kode ini berada (asumsi func main()).

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"
```

```

func main() {
    var orang int
    fmt.Print("Masukkan jumlah orang: ")
    fmt.Scan(&orang)

    motor := orang / 2
    if orang%2 != 0 {
        motor++
    }
    fmt.Println("Jumlah motor yang dibutuhkan:", motor)
}

```

## Screenshoot program

The screenshot shows the Visual Studio Code interface with several tabs open at the top. The active tab contains the following Go code:

```

package main
import "fmt"
func main() {
    var orang int
    fmt.Print("Masukkan jumlah orang: ")
    fmt.Scan(&orang)

    motor := orang / 2
    if orang%2 != 0 {
        motor++
    }
    fmt.Println("Jumlah motor yang dibutuhkan:", motor)
}

```

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

```

PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\soal 1\soal 1.go"
Masukkan jumlah orang: 25
Jumlah motor yang dibutuhkan: 13
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\soal 1\soal 1.go"
Masukkan jumlah orang: 15
Jumlah motor yang dibutuhkan: 8
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\soal 1\soal 1.go"
Masukkan jumlah orang: 8
Jumlah motor yang dibutuhkan: 4
PS C:\Users\MSI>

```

A separate window titled "LightClock" is also visible, showing the system date and time.

## Deskripsi program

### 1. package main

Menandakan bahwa file ini adalah program yang dapat dieksekusi secara langsung (bukan *library*).

### 2. import "fmt"

Mengimpor paket standar Go **fmt** (format). Paket ini diperlukan untuk melakukan operasi **I/O** (Input/Output) dasar, seperti membaca dari input (**fmt.Scan**) dan menulis ke layar (**fmt.Println**).

### **3. func main() {**

**func main()** adalah **titik masuk** program, kode di dalam **func main()** dijalankan pertama kali setelah kode atau program diberi nama. Bentuknya harus "**func main()**" tanpa parameter dan tanpa nilai balik.

### **4. var orang int**

Mendeklarasikan variabel bernama **orang** dengan tipe data **int** (bilangan bulat). Variabel ini akan menyimpan input jumlah orang.

### **5. fmt.Print("Masukkan jumlah orang: ")**

Mencetak pesan "**Masukkan jumlah orang:**" ke konsol. Fungsi **fmt.Print** tidak menambahkan baris baru setelah pesan.

### **6. fmt.Scan(&orang)**

Membaca bilangan bulat dari input pengguna dan menyimpannya ke variabel **orang**. Tanda **&** menunjukkan alamat memori variabel.

### **7. motor := orang / 2**

Mendeklarasikan dan menginisialisasi variabel **motor**. Karena ini adalah pembagian **integer**, hasil pembagian **orang / 2** akan dibulatkan ke bawah (floor). Contoh:  $10 / 2 = 5$ ,  $9 / 2 = 4$ .

### **8. if orang%2 != 0 {**

Memulai blok kondisi **if**. Ekspresi **orang%2 != 0** memeriksa apakah sisa bagi dari **orang** dengan 2 **tidak sama dengan 0**. Ini berarti kondisinya **true** jika **orang** adalah **bilangan ganjil**.

### **9. motor++**

Jika kondisi di atas terpenuhi (jumlah **orang** ganjil), ini berarti ada satu orang yang tersisa dari pembagian sebelumnya. Operator **++** akan **menambahkan 1** ke nilai variabel **motor** untuk menampung satu orang yang tersisa tersebut.

### **10. }**

Mengakhiri blok kode untuk kondisi **if**.

### **11. fmt.Println("Jumlah motor yang dibutuhkan:", motor)**

Mencetak hasil akhir ke konsol, menampilkan string diikuti dengan nilai variabel **motor**.

**12. }**

Mengakhiri fungsi tempat kode ini berada (asumsi **func main()**).

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)

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

### Screenshoot program

The screenshot shows a Visual Studio Code interface with several tabs open at the top. The active tab contains Go code for determining if a number is even or odd. The terminal below shows the execution of the program, where it prompts for input, reads -7, and outputs "bukan genap negatif". It then runs again with inputs -12 and -8, both of which are correctly identified as even numbers.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     fmt.Print("Masukkan bilangan: ")
8     fmt.Scan(&n)
9
10    if n < 0 && n%2 == 0 {
11        fmt.Println("genap negatif")
12    } else {
13        fmt.Println("bukan genap negatif")
14    }
15 }
```

```
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\soal 2\soal 2.go"
Masukkan bilangan: -7
bukan genap negatif
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\soal 2\soal 2.go"
Masukkan bilangan: -12
genap negatif
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 9\LAPRAK\soal 2\soal 2.go"
Masukkan bilangan: -8
genap negatif
PS C:\Users\MSI>
```

## Deskripsi program

### 1. package main

Menandakan bahwa file ini adalah program yang dapat dieksekusi secara langsung (bukan *library*).

### 2. import "fmt"

Mengimpor paket standar Go **fmt** (format). Paket ini diperlukan untuk melakukan operasi **I/O** (Input/Output) dasar, seperti membaca dari input (`fmt.Scan`) dan menulis ke layar (`fmt.Println`).

### 3. func main() {

`func main()` adalah **titik masuk** program, kode di dalam `func main()` dijalankan pertama kali setelah kode atau program diberi nama. Bentuknya harus "**func main()**" tanpa parameter dan tanpa nilai balik.

### 4. var n int

Mendeklarasikan variabel bernama **n** dengan tipe data **int** (bilangan bulat). Variabel ini akan menyimpan input angka dari pengguna.

### 5. `fmt.Print("Masukkan bilangan: ")`

Mencetak pesan "**Masukkan bilangan:**" ke konsol, meminta pengguna untuk memasukkan angka.

### 6. `fmt.Scan(&n)`

Membaca bilangan bulat dari *input* pengguna dan menyimpannya ke variabel **n**. Tanda **&** (alamat memori) diperlukan untuk menyimpan nilai yang dibaca.

### **7. if n < 0 && n%2 == 0 {**

Memulai blok kondisi. Program akan mengeksekusi kode di dalamnya hanya jika **kedua** syarat ini terpenuhi (dihubungkan oleh operator **AND &&**):

1. **n < 0**: Bilangan **n** harus **negatif**.
2. **n%2 == 0**: Bilangan **n** harus **genap** (sisa bagi dengan 2 adalah 0).

### **8. fmt.Println("genap negatif")**

Jika kondisi di Baris 10 terpenuhi (bilangan genap **dan** negatif, misalnya -4), program akan mencetak "**genap negatif**".

### **9. } else {**

Memulai blok kode alternatif yang akan dieksekusi jika kondisi di Baris 10 **tidak** terpenuhi (yaitu, jika bilangan tersebut positif, nol, atau ganjil).

### **10. fmt.Println("bukan genap negatif")**

Jika kondisi di Baris 10 gagal, program akan mencetak "**bukan genap negatif**".

### **11. }**

Mengakhiri blok kode untuk kondisi **if-else**.

### **12. }**

Mengakhiri fungsi tempat kode ini berada (asumsi **func main()**).

## **3. Tugas 3**

### **Source code**

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Print("Masukkan bilangan X: ")
    fmt.Scan(&x)
    fmt.Print("Masukkan bilangan Y: ")
    fmt.Scan(&y)

    fmt.Println(y%x == 0)
```

```
    fmt.Println(x%y == 0)  
}
```

## Screenshot program

The screenshot shows a Visual Studio Code interface. On the left, there's a sidebar with icons for files, search, and other code-related functions. The main area displays a Go file named 'main.go' with the following code:

```
package main  
import "fmt"  
  
func main() {  
    var x, y int  
    fmt.Print("Masukkan bilangan X: ")  
    fmt.Scan(&x)  
    fmt.Print("Masukkan bilangan Y: ")  
    fmt.Scan(&y)  
  
    fmt.Println(y%x == 0)  
    fmt.Println(x%y == 0)  
}
```

To the right of the editor, a terminal window titled 'LightClock' is open, showing the output of the program:

```
109082500138  
S1IF-13-07  
ANINDYA RAHADITA YUMNAA
```

Below the terminal, the status bar indicates 'Ln 3, Col 24 48 caractere Plain t 100% Wind UTF-8'. At the bottom of the screen, the terminal shows command-line history and file paths.

## Deskripsi program

### 1. package main

Menandakan bahwa file ini adalah program yang dapat dieksekusi secara langsung (bukan *library*).

### 2. import "fmt"

Mengimpor paket standar Go **fmt** (format). Paket ini diperlukan untuk melakukan operasi **I/O** (Input/Output) dasar, seperti membaca dari input (`fmt.Scan`) dan menulis ke layar (`fmt.Println`).

### 3. func main() {

`func main()` adalah **titik masuk** program, kode di dalam `func main()` dijalankan pertama kali setelah kode atau program diberi nama. Bentuknya harus "**func main()**" tanpa parameter dan tanpa nilai balik.

### 4. var x, y int

Mendeklarasikan dua variabel sekaligus, **x** dan **y**, dengan tipe data **int** (bilangan bulat). Kedua variabel ini akan menyimpan input angka dari pengguna.

### 5. `fmt.Print("Masukkan bilangan X: ")`

Mencetak pesan "**Masukkan bilangan X:**" ke konsol.

## **6. fmt.Scan(&x)**

Membaca bilangan bulat dari *input* dan menyimpannya ke variabel **x**.

## **7. fmt.Println("Masukkan bilangan Y: ")**

Mencetak pesan "**Masukkan bilangan Y:**" ke konsol.

## **8. fmt.Scan(&y)**

Membaca bilangan bulat dari *input* dan menyimpannya ke variabel **y**.

## **9. fmt.Println(y%x == 0)**

**Pengecekan 1 (Y habis dibagi X):** Mencetak hasil evaluasi boolean dari ekspresi  $y \% x == 0$ . Ini bernilai **true** jika sisa pembagian **Y** oleh **X** adalah **nol**, yang berarti **Y habis dibagi X** (atau **X** adalah faktor dari **Y**).

## **10. fmt.Println(x%y == 0)**

**Pengecekan 2 (X habis dibagi Y):** Mencetak hasil evaluasi boolean dari ekspresi  $x \% y == 0$ . Ini bernilai **true** jika sisa pembagian **X** oleh **Y** adalah **nol**, yang berarti **X habis dibagi Y** (atau **Y** adalah faktor dari **X**).

## **11. }**

Mengakhiri fungsi tempat kode ini berada (asumsi **func main()**).