

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 9**

**IF-THEN**



**Disusun oleh:**

**HAFIZD SAMA'I SYAMSI**

**109082500183**

**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)

}
```

#### Screenshoot program

```

Welcome | guided1.go 2 X | guided2.go 1 | guided3.go 1
guided1.go > main
1 package main
2 import "fmt"
3 func main() {
4     var bilangan int
5     fmt.Scan(&bilangan)
6     if bilangan < 0 {
7         bilangan = -bilangan
8     }
9     fmt.Println(bilangan)
10 }

PROBLEMS 4 | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS

10
10
PS D:\go\praktek6> go run guided1.go
-3
3
PS D:\go\praktek6> go run guided1.go
5
5
PS D:\go\praktek6> go run guided1.go
0
0
PS D:\go\praktek6> go run guided1.go
-9999
9999
PS D:\go\praktek6>

```

### Deskripsi program

Program GO ini di mulai dengan membuat sebuah variabel bernama bilangan dengan tipe data int. Setelah itu program menggunakan `fmt.Scan()` untuk menerima input angka dari pengguna dan menyimpannya dengan variabel.

Selanjutnya program menjalankan dengan if. Jika nilai bilangan kurang dari 0 (artinya bilangan negatif), maka program akan mengalikan nilai tersebut dengan -1, sehingga berubah menjadi positif.

Jika bilangan yang dimasukan sudah positif, bagian if dilewati begitu saja. Diakhir program mencetak nilai bilangan yang sudah diproses tadi menggunakan `fmt.Println()`.

Jadi apapun angkanya – negatif atau positif – program akan menampilkan nilai akhirnya, namun jika awalnya negatif, program memastikan hasilnya menjadi positif terlebih dahulu.

## 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

```
welcome | guided1.go 2 | guided2.go 1 x | guided3.go 1
guided2.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 }
13

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS
go run: unknown command
Run 'go help' for usage.
PS D:\go\praktek6> go run guided2.go
10
positif
PS D:\go\praktek6> go run guided2.go
-3
bukan positif
PS D:\go\praktek6> go run guided2.go
5
positif
PS D:\go\praktek6> go run guided2.go
0
bukan positif
PS D:\go\praktek6> 
```

### Deskripsi program

Program ini digunakan untuk menentukan apakah sebuah bilangan termasuk positif atau bukan positif. Di dalam program, ada dua variabel yang digunakan.

1. Bilangan – bertipe int, untuk menyimpan angka yang dimasukan oleh pengguna
2. Teks – bertipe string, untuk menampung hasil berupa tulisan

Pertama program meminta pengguna memasukan angka, lalu angka itu dibaca dengan `fmt.Scan()` dan disimpan ke variabel `bilangan`.

Setelah angka diterima, program langsung memberi nilai awal pada variabel teks yaitu “bukan positif. Ini artinya program menganggap bahwa bilangan tersebut bukan positif kecuali terbukti sebaliknya.

Lalu pengecekan dengan if. Jika nilai bilangan lebih besar dari 0, maka isi variabel teks diganti menjadi positif. Akhirnya program mencetak isi variabel teks ke layar, sehingga pengguna bisa mengetahui apakah bilangan yang dimasukkan adalah bilangan positif atau bukan.

### 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)
}
```

#### Screenshoot program

Go guided3.go > main

```
1 package main
2 import "fmt"
3 func main() {
4     var bilangan int
5     var hasil bool
6     fmt.Scan(&bilangan)
7     hasil = bilangan % 2 == 0 && bilangan < 0
8     fmt.Println(hasil)
9 }
```

PROBLEMS 4

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

PS D:\go\praktek6> go run guided3.go

Masukan bilangan bulat 10

Hasil true

PS D:\go\praktek6> go run guided3.go

10

false

PS D:\go\praktek6> go run guided3.go

-3

false

PS D:\go\praktek6> go run guided3.go

-4

true

PS D:\go\praktek6> go run guided3.go

0

false

PS D:\go\praktek6> go run guided3.go

-2

true

PS D:\go\praktek6> █

### Deskripsi program

Program ini digunakan untuk memeriksa apakah sebuah bilangan memenuhi dua syarat sekaligus, yaitu genap dan negatif. Didalm program, program terdapat dua variabel.

1. Bilangan – bertipe int, digunakan untuk menyimpan angka yang dimasukan pengguna
2. Hasil – bertipe bool, berfungsi menyimpan nilai true atau false berdasarkan pengecekan kondisi.

Pertama program membaca input angka dari pengguna menggunakan `fmt.Scan()` dan menyimpan ke variabel bilangan. Setelah itu program melakukan pengecekan dengan ekspresi boolean: `bilangan % 2 == 0 && bilangan < 0` penjelasannya:

- `Bilangan % 2 == 0` → mengecek apakah bilangan tersebut genap
- `Bilangan < 0` → mengecek apakah bilangan tersebut negatif
- Operator `&&` berarti kedua kondisi harus benar agar hasilnya true

Jika bilangan yang dimasukan adalah genap dan negatif, variabel hasil akan bernilai true. Jika salah satu atau keduanya syarat tidak terpenuhi nilai hasil menjadi false.

Terakhir program menampilkan nilai boolean tersebut menggunakan `fmt.Println()`, sehingga pengguna bisa mengetahui apakah bilangan yang dimasukan adalah bilangan genap negatif atau bukan.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {
    var orang int

    fmt.Print("Masukan jumlah orang")

    fmt.Scan(&orang)

    var motor int

    if orang%2 == 0 {
        motor = orang / 2
    } else {
```



```

        motor = (orang / 2) + 1
    }

    fmt.Println(motor)
}

```

## Screenshoot program

The screenshot shows a Go program in VS Code. The editor displays the following code:

```

3 func main() {
4     var orang int
5     fmt.Print("Masukan jumlah orang")
6     fmt.Scan(&orang)
7     var motor int
8     if orang%2 == 0 {
9         motor = orang / 2
10    } else {
11        motor = (orang / 2) + 1
12    }
13    fmt.Println(motor)
14
15
16 }

```

The terminal shows the following output:

```

# command-line-arguments
.\soal1.go:6:17: syntax error: unexpected {, expected expression
.\soal1.go:8:2: syntax error: unexpected }, expected {
.\soal1.go:12:1: syntax error: unexpected }, expected expression
PS D:\go\praktek6> go run soal1.go
10
10
PS D:\go\praktek6> go run soal1.go
Masukan jumlah orang 10
5
PS D:\go\praktek6> go run soal1.go
Masukan jumlah orang 1
1
PS D:\go\praktek6> go run soal1.go
Masukan jumlah orang 25
13
PS D:\go\praktek6> go run soal1.go
Masukan jumlah orang 9
5

```

On the right side of the terminal, there is a small window titled 'NA' containing the following text:

```

NAMA : HAFIZD SAMA'I SYAMSI
KELAS : S1IF-13-07
NIM : 109082500183

```

## Deskripsi program

Program ini dimulai dengan membuat sebuah variabel orang untuk menampung jumlah orang yang akan dihitung.

Pengguna diminta memasukkan angkanya melalui `fmt.Scan(&orang)`.

Setelah jumlah orang didapat, program menyiapkan variabel motor untuk menyimpan jumlah motor yang dibutuhkan.

Lalu program melakukan pengecekan:

➤ Jika `orang % 2 == 0`

Artinya jumlah orang genap.

Maka setiap dua orang bisa naik satu motor, sehingga jumlah motor dihitung dengan `motor = orang / 2`.

➤ Jika kondisi genap tidak terpenuhi (berarti ganjil)

Pembagian orang / 2 menyisakan satu orang (karena pembagian bilangan bulat).

Untuk orang yang tersisa itu, program menambah satu motor.

Rumus yang dipakai:  $\text{motor} = (\text{orang} / 2) + 1$ .

Terakhir, program mencetak nilai motor yang berisi total motor yang harus disediakan.

## 2. Tugas 2

### Source code

```
import "fmt"

func main() {
    var bilangan int
    var teks string
    fmt.Scan(&bilangan)
    teks = "bukan"
    if bilangan < 0 {
        teks = "genap positif"
    }
    fmt.Println(teks)
}
```

### Screenshoot program

The screenshot shows a Go IDE with several tabs: 'Welcome', 'guided1.go 4', 'guided2.go 1', 'guided3.go 1', 'soal1.go 1', and 'soal2.go 1 X'. The active tab is 'soal2.go', which contains the following code:

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

The bottom panel shows the 'TERMINAL' tab with the following output:

```
genap negatif
PS D:\go\praktek6> go run soal2.go
10
genap negatif
PS D:\go\praktek6> go run soal2.go
10
genap positif
PS D:\go\praktek6> go run soal2.go
10
bukan
PS D:\go\praktek6> go run soal2.go
-4
genap positif
PS D:\go\praktek6> go run soal2.go
0
bukan
PS D:\go\praktek6> go run soal2.go
-2
genap positif
PS D:\go\praktek6>
```

On the right side, there is a small window titled 'NA' with a menu bar (File, Edit, View, Aa) and the following text:

```
NAMA : HAFIZD SAMA'I SYAMSI
KELAS : S1IF-13-07
NIM : 109082500183
```

### Deskripsi program

Program ini menyiapkan dua variabel:

- bilangan untuk menyimpan angka yang dimasukkan pengguna
- teks untuk menampung tulisan yang akan ditampilkan

Setelah pengguna memasukkan sebuah angka melalui `fmt.Scan(&bilangan)`, program langsung memberi nilai awal pada teks, yaitu "bukan".

Selanjutnya ada pengecekan:

- Jika `Bilangan < 0`, berarti angkanya bernilai negatif. Jika kondisi ini benar, nilai teks diubah menjadi "genap positif".

Jika kondisi tersebut tidak terpenuhi, nilai teks tetap "bukan".

Di akhir, program mencetak isi variabel teks ke layar.

### 3. Tugas 3

#### Source code

```
package main
import "fmt"
func main() {
```

```

var x, y int
fmt.Print("Masukan x dan y")
fmt.Scan(&x, &y)

var xFaktorY bool
var yFaktorX bool

if y % x == 0 {
    xFaktorY = true
} else {
    xFaktorY = false
}
if x % y == 0 {
    yFaktorX = true
} else {
    yFaktorX = false
}

fmt.Println(xFaktorY)
fmt.Println(yFaktorX)
}

```

## Screenshoot program

The screenshot shows the Go program code in the editor and its execution results in the terminal. The code checks if one number is a factor of another. The terminal shows three test cases: (10, 5) where x is a factor of y, (3, 21) where y is a factor of x, and (4, 4) where both are factors.

```

soal3.go > main
3 func main() {
4     fmt.Scan(&x, &y)
5
6     var xFaktorY bool
7     var yFaktorX bool
8
9     if y % x == 0 {
10        xFaktorY = true
11    } else {
12        xFaktorY = false
13    }
14    if x % y == 0 {
15        yFaktorX = true
16    } else {
17        yFaktorX = false
18    }
19    fmt.Println(xFaktorY)
20    fmt.Println(yFaktorX)
21 }

```

TERMINAL OUTPUT:

```

Masukan x dan y 10 5
false
true
PS D:\go\praktek6> go run soal3.go
Masukan x dan y 3 21
false
true
PS D:\go\praktek6> go run soal3.go
Masukan x dan y 10 5
false
true
PS D:\go\praktek6> go run soal3.go
Masukan x dan y 3 21
true
false
PS D:\go\praktek6> go run soal3.go
Masukan x dan y 4 4
true
true
PS D:\go\praktek6>

```

File Edit View

NAMA : HAFIZD SAMA'I SYAMSI  
 KELAS : S1IF-13-07  
 NIM : 109082500183

Ln 3, Col 19 - 64 characters Plain Text 100% Wind - UTF

### Deskripsi program

Program ini membaca dua angka dari pengguna, yaitu x dan y. Setelah nilainya diterima, program membuat dua variabel boolean:

- xFaktorY → untuk menandai apakah x merupakan faktor dari y
- yFaktorX → untuk menandai apakah y merupakan faktor dari x

Kemudian program melakukan dua pengecekan:

1. Pengecekan apakah x faktor dari y  
Dilakukan dengan kondisi  $y \% x == 0$ .
  - Jika sisa bagi y dibagi x adalah 0, berarti x membagi y dengan tepat, dan xFaktorY diisi true.
  - Jika tidak, xFaktorY bernilai false.
2. Pengecekan apakah y faktor dari x  
Dilakukan dengan kondisi  $x \% y == 0$ .
  - Jika x habis dibagi y, maka y adalah faktor dari x, sehingga yFaktorX menjadi true.
  - Jika tidak, nilainya false.

Setelah kedua pengecekan selesai, program menampilkan hasilnya dengan mencetak nilai:

- xFaktorY
- yFaktorX

Keduanya akan bernilai true atau false sesuai hasil perhitungan