

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 9

IF-THEN



Disusun oleh:

Ichsan Maulana Muhammad

109082500093

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

LATIHAN KELAS – GUIDED

1. Guided 1

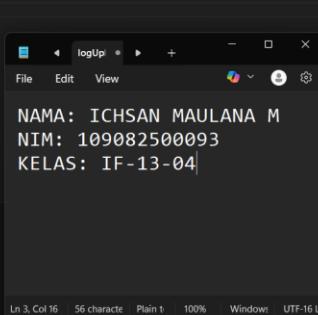
Source Code

```
package main

import "fmt"

func main() {
    var bilangan int
    fmt.Scan(&bilangan)
    if bilangan < 0 {
        bilangan = -bilangan
    }
    fmt.Println(bilangan)
}
```

Screenshot program



```
NAMA: ICHSAN MAULANA M
NIM: 109082500093
KELAS: IF-13-04
```

The screenshot shows a Windows terminal window titled "logUp". Inside the terminal, the user has run a Go program named "guided 1.go". The program prompts the user to enter a number, which is then printed back to the screen. The user has entered "9999" and the program has printed "9999" back to the user.

```
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\guided 1.go"
9999
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\guided 1.go"
9999
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\guided 1.go"
9999
PS C:\Users\ASUS\yy>
```

Deskripsi program

1. Program diawali dengan package main, import "fmt", serta fungsi main() sebagai fungsi utama untuk menjalankan program.
2. Mendeklarasikan variabel bilangan dengan tipe data integer yang akan digunakan untuk menampung input dari pengguna.
3. Program meminta pengguna menginputkan sebuah bilangan bulat, kemudian membaca input tersebut menggunakan fmt.Scan(&bilangan).
4. Melakukan pengecekan nilai bilangan menggunakan struktur seleksi if:
 - Jika bilangan < 0, maka program mengubah nilainya menjadi positif dengan cara bilangan = -bilangan.
5. Jika bilangan ≥ 0 , maka nilai bilangan dibiarkan tanpa perubahan karena sudah bernilai absolut.
6. Program menampilkan nilai absolut dari bilangan yang dimasukkan pengguna menggunakan fmt.Println(bilangan).
7. Output berupa nilai bilangan dalam bentuk absolut, sehingga bilangan negatif berubah menjadi positif, sedangkan bilangan positif maupun nol tetap.

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

The screenshot shows a Go code editor interface. On the left is the Explorer sidebar with project structures for MODUL 4, MODUL 5 DAN 6, MODUL 7, and MODUL 9. The 'guided II.go' file is selected in the Explorer. The main editor area contains the following Go 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)
}
```

To the right of the editor is a terminal window titled 'logUp' showing the execution of the program:

```
NAMA: ICHSAN MAULANA M
NIM: 109082500093
KELAS: IF-13-04

PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\guided II.go"
10
positif
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\guided II.go"
-3
bukan positif
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\guided II.go"
5
positif
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\guided II.go"
0
bukan positif
PS C:\Users\ASUS\yy>
```

The status bar at the bottom indicates the current line (Ln 15), column (Col 1), tab size (Tab Size 4), and encoding (UTF-8).

Deskripsi program

1. Program diawali dengan package main, import "fmt", dan fungsi main() sebagai fungsi utama untuk menjalankan program.
2. Di dalam fungsi main(), dideklarasikan dua variabel:
 - bilangan bertipe int untuk menampung input angka dari pengguna.
 - teks bertipe string yang digunakan untuk menyimpan hasil keluaran.
3. Program meminta pengguna untuk memasukkan sebuah bilangan bulat, dan input tersebut dibaca menggunakan fmt.Scan(&bilangan).
4. Variabel teks pertama-tama diisi dengan nilai awal "bukan positif", karena secara default program mengasumsikan bilangan bukan positif.
5. Program memeriksa apakah nilai bilangan lebih besar dari 0 menggunakan percabangan if:
 - Jika bilangan > 0, maka teks diubah menjadi "positif".
6. Jika bilangan sama dengan 0 atau kurang dari 0, kondisi if akan bernilai false, sehingga nilai teks tetap "bukan positif".
7. Program menampilkan hasil akhir menggunakan fmt.Println(teks), yaitu teks yang menyatakan apakah bilangan positif atau bukan positif.
8. Output berupa teks "positif" atau "bukan positif", tergantung dari bilangan yang diinputkan 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 the following components:

- EXPLORER:** Shows a file tree with packages like MODUL 5 DAN 6, MODUL 7, and MODUL 9, and files such as guided one.go, guided three.go, tempCodeRunnerFile.go, tugas four.go, tugas one.go, tugas three.go, tugas two.go, assessment1.go, assessment2.go, assessment3.go, and guided III.go.
- CODE EDITOR:** Displays the source code for `guided III.go`. The code defines a `main()` function that reads an integer from the user, checks if it's even and less than zero, and prints the result.
- TERMINAL:** Shows command-line history and output. It includes commands like `go run` and `PS C:\Users\ASUS\yy>`, along with the results of the program execution.
- OUTPUT:** A floating window titled "logUp" showing the program's output:

```
NAMA: ICHSAN MAULANA M
NIM: 109082500093
KELAS: IF-13-04|
```

Deskripsi program

1. Program diawali dengan package main, import "fmt", dan fungsi main() sebagai fungsi utama untuk menjalankan program.
2. Di dalam fungsi main(), dideklarasikan dua variabel:
 - bilangan bertipe int untuk menampung input angka dari pengguna.
 - hasil bertipe bool yang digunakan untuk menyimpan keluaran berupa nilai benar atau salah.
3. Program meminta pengguna untuk memasukkan sebuah bilangan bulat, dan nilai tersebut dibaca menggunakan fmt.Scan(&bilangan).
4. Variabel hasil pada awalnya tidak diberi nilai khusus, karena nilai akhirnya akan langsung ditentukan oleh ekspresi logika pada langkah berikutnya.
5. Program memeriksa apakah bilangan tersebut genap negatif dengan menggunakan kondisi:
 - bilangan % 2 == 0 untuk mengecek apakah bilangan genap.
 - bilangan < 0 untuk mengecek apakah bilangan negatif.
 - Kedua kondisi digabung menggunakan operator logika &&.
6. Jika kedua syarat terpenuhi (genap dan negatif), variabel hasil bernilai true.
 - Jika salah satu atau kedua syarat tidak terpenuhi, maka hasil bernilai false.
7. Program menampilkan nilai keluaran menggunakan fmt.Println(hasil), yaitu nilai boolean yang menunjukkan apakah bilangan tersebut merupakan bilangan genap negatif atau bukan.
8. Output berupa nilai true atau false, sesuai dengan hasil evaluasi terhadap bilangan yang dimasukkan pengguna.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

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

    if orang%2 == 0 {
        orang = orang / 2
    } else if orang%2 != 0 {
        orang = orang/2 + 1
    }
    fmt.Println(orang)
}
```

Screenshot program

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

- File Explorer:** Shows files in the current workspace, including `yayayaya.go`, `guided I.go`, `guided II.go`, `guided III.go`, `tugas I.go`, `tugas II.go`, `tugas III.go`, and `abcdefwygdeuhw.go`.
- Code Editor:** The `tugas I.go` file is open, containing the following Go code:

```
package main
import "fmt"
func main() {
    var orang int
    fmt.Scan(&orang)
    if orang%2 == 0 {
        orang = orang / 2
    } else if orang%2 != 0 {
        orang = orang/2 + 1
    }
    fmt.Println(orang)
}
```
- Terminal:** The terminal window shows command-line interactions:

```
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\tugas I.go"
10
10
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\tugas I.go"
1
1
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\tugas I.go"
25
25
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 9\tugas I.go"
9
5
PS C:\Users\ASUS\yy>
```
- Output Window:** Displays the output of the program runs, showing the value of `orang` for different input values.

Deskripsi program

1. Program diawali dengan package main, import "fmt", dan fungsi main() sebagai fungsi utama yang akan dieksekusi.
 2. Di dalam fungsi main(), dideklarasikan sebuah variabel bertipe int, yaitu:
 - orang → menyimpan jumlah orang yang akan ikut touring.
 3. Program meminta pengguna untuk memasukkan sebuah bilangan bulat, yaitu jumlah peserta touring.
Input ini dibaca menggunakan fmt.Scan(&orang).
 4. Setiap motor hanya dapat ditumpangi oleh 2 orang, sehingga jumlah motor dapat dihitung dengan cara membagi jumlah orang dengan 2.
 5. Program menggunakan percabangan untuk memastikan pembagian dilakukan dengan benar:
 - Jika jumlah orang genap (orang % 2 == 0), maka jumlah motor adalah orang / 2.
 - Jika jumlah orang ganjil (orang % 2 != 0), maka orang terakhir tetap membutuhkan satu motor tambahan, sehingga jumlah motor dihitung dengan orang / 2 + 1.
 6. Hasil jumlah motor disimpan ke variabel dan kemudian ditampilkan menggunakan fmt.Println().
 7. Keluaran program berupa sebuah bilangan bulat yang menyatakan jumlah motor yang diperlukan untuk seluruh peserta touring.

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 Go development environment with the following details:

- File Explorer:** Shows various Go files in the workspace, including `main.go`, `yayayaya.go`, and several `guided` and `tugas` files.
- Code Editor:** Displays the source code for `main.go`. The code checks if a scanned integer is even and negative, printing "Genap Negatif" if true, or "bukan" otherwise.
- Terminal:** Shows the command `go run "c:\Users\ASUS\yy\MODUL 9\tempCodeRunnerFile.go"` being run three times, resulting in the output:
 - Line 10: bukan
 - Line 12: Genap Negatif
 - Line 14: bukan
- Output Panel:** Shows the user information:

NAMA: ICHSAN MAULANA M
NIM: 109082500093
KELAS: IF-13-04

Deskripsi program

1. Program diawali dengan package main, pernyataan import "fmt", serta fungsi main() sebagai fungsi utama yang akan dijalankan ketika program dieksekusi.
2. Di dalam fungsi main(), dideklarasikan sebuah variabel bertipe int, yaitu:
 - bilangan → digunakan untuk menyimpan bilangan bulat yang dimasukkan oleh pengguna.
3. Program meminta pengguna untuk memasukkan sebuah bilangan bulat. Nilai ini dibaca menggunakan perintah fmt.Scan(&bilangan).
4. Program kemudian melakukan pengecekan terhadap dua kondisi, yaitu apakah bilangan tersebut:
 - bernilai negative dan merupakan bilangan genap (bilangan % 2 == 0)
5. Program menggunakan percabangan (if–else) untuk menentukan output:
 - Jika bilangan < 0 dan bilangan % 2 == 0, maka bilangan tersebut adalah "genap negatif".
 - Jika kedua syarat tersebut tidak terpenuhi, maka program menampilkan "bukan".
6. Output program dicetak menggunakan fmt.Println() sesuai hasil pengecekan kondisi.
7. Keluaran program berupa teks, yaitu:
 - "genap negatif" jika bilangan yang dimasukkan termasuk bilangan genap dan bernilai negatif.
 - "bukan" jika bilangan tidak memenuhi kedua syarat tersebut.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var x, y int
    var hasil1, hasil2 bool
    fmt.Scan(&x, &y)
    if x <= y && y%x == 0 {
        hasil1 = true
    } else {
        hasil1 = false
    }
    if y <= x && x%y == 0 {
```

```
    hasil2 = true

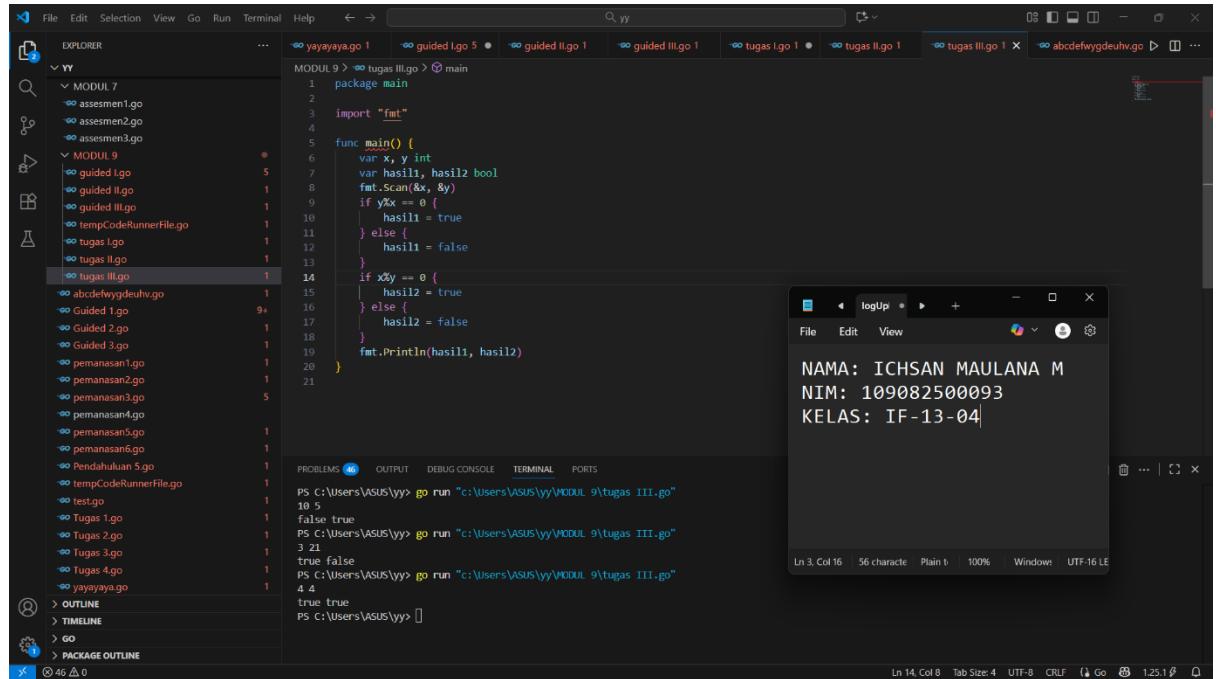
} else {

    hasil2 = false

}

fmt.Println(hasil1, hasil2)
```

Screenshot program



Deskripsi program

1. Program diawali dengan package main, pernyataan import "fmt", serta fungsi main() sebagai fungsi utama yang akan dieksekusi ketika program dijalankan.
 2. Di dalam fungsi main(), dideklarasikan dua variabel bertipe int, yaitu:
 - x → bilangan pertama yang diinput oleh pengguna
 - y → bilangan kedua yang diinput oleh pengguna

Selain itu, terdapat dua variabel bertipe bool, yaitu

- hasil1 → menyimpan hasil pengecekan apakah x merupakan faktor dari y
 - hasil2 → menyimpan hasil pengecekan apakah y merupakan faktor dari x

3. Program meminta pengguna untuk memasukkan dua bilangan bulat. Kedua nilai tersebut dibaca menggunakan perintah:
 - `fmt.Scan(&x, &y)`
 4. Program kemudian melakukan dua pengecekan kondisi, yaitu apakah:
 - x adalah faktor dari y , yang berarti y habis dibagi x ($y \% x == 0$)

- y adalah faktor dari x , yang berarti x habis dibagi y ($x \% y == 0$)
5. Program menggunakan percabangan (if–else) untuk menentukan nilai boolean masing-masing kondisi:
 - Jika $y \% x == 0$, maka variabel hasil1 bernilai true, jika tidak maka bernilai false.
 - Jika $x \% y == 0$, maka variabel hasil2 bernilai true, jika tidak maka bernilai false.
 6. Hasil dari kedua pengecekan ditampilkan menggunakan `fmt.Println()`, masing-masing pada baris terpisah.
 7. Keluaran program berupa dua nilai boolean, yaitu:
 - Nilai pertama menunjukkan apakah x merupakan faktor dari y .
 - Nilai kedua menunjukkan apakah y merupakan faktor dari x .
Nilai output dapat berupa true atau false sesuai hasil pengecekan pada program.