

# **LAPORAN PRAKTIKUM**

## **Algoritma Pemrograman**

MODUL 9

IF-THEN



**Disusun Oleh:**

MUHAMAD RAFI ALFIANSYAH

109082500191

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 x int
    fmt.Scan(&x)
    if x<0{
        x=-x
    }
    fmt.Println(x)
}
```

#### Screenshot Program

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the workspace structure with files G1.go, G2.go, G3.go, T1.go, T2.go, and T3.go.
- Code Editor:** Displays the content of G1.go.
- Terminal:** Shows the command-line interface where the program is run multiple times with different inputs:

```
PS C:\Modul 9> go run "c:\Modul 9\G1.go"
-3
PS C:\Modul 9> go run "c:\Modul 9\G1.go"
5
PS C:\Modul 9> go run "c:\Modul 9\G1.go"
0
PS C:\Modul 9> go run "c:\Modul 9\G1.go"
-9999
9999
PS C:\Modul 9>
```

#### Deskripsi Program

Program ini diawali dengan deklarasi package main yang menandakan bahwa kode ini merupakan program utama yang dapat langsung dijalankan oleh compiler Go. Setelah itu terdapat perintah import "fmt" yang berfungsi untuk memanggil pustaka standar Go bernama fmt. Pustaka ini digunakan agar program bisa menjalankan proses input dan output, seperti membaca data dari pengguna dengan fmt.Scan dan menampilkan hasil ke layar dengan fmt.Println.

Selanjutnya, fungsi utama func main() didefinisikan sebagai titik awal eksekusi program. Di dalam fungsi ini terdapat satu variabel bernama x yang bertipe data int. Variabel tersebut digunakan untuk menampung nilai bilangan bulat yang dimasukkan oleh pengguna. Nilai input dibaca menggunakan fmt.Scan(&x), di mana tanda & menunjukkan bahwa data yang dimasukkan akan disimpan langsung ke alamat memori dari variabel x.

Setelah pengguna memasukkan nilai, program akan melakukan proses pemeriksaan menggunakan struktur kendali if  $x < 0$ . Kondisi ini digunakan untuk mengecek apakah nilai yang dimasukkan merupakan bilangan negatif. Jika benar, maka program akan mengeksekusi perintah  $x = -x$ , yang berarti nilai x akan dikalikan dengan -1 untuk mengubahnya menjadi positif. Dengan cara ini, bilangan negatif seperti -3 akan diubah menjadi 3, sedangkan bilangan yang sudah positif tidak akan mengalami perubahan karena tidak memenuhi kondisi tersebut.

Langkah terakhir dalam program ini adalah menampilkan hasil akhir ke layar dengan menggunakan perintah fmt.Println(x). Nilai yang ditampilkan merupakan nilai absolut atau nilai mutlak dari bilangan yang dimasukkan oleh pengguna. Dengan demikian, berapapun nilai inputnya baik positif, negatif, maupun nol, program akan selalu menghasilkan bilangan positif yang merepresentasikan nilai mutlak dari input tersebut.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {
    var x int
    fmt.Scan(&x)
    h := "positif"
    if x<0{
        h = "bukan " + h
    }
    fmt.Println(h)

}
```

### Screenshot Program

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** Shows the workspace 'Modul 9' containing files G1.go, G2.go, G3.go, T1.go, T2.go, and T3.go.
- Code Editor:** The file G2.go is open, displaying the provided Go code.
- Terminal:** The terminal window shows the command `go run "c:\Modul 9\G2.go"` being run, followed by the output "positif".

### Deskripsi Program

Program ini diawali dengan package main yang menandakan bahwa program ini merupakan program utama yang dapat langsung dijalankan tanpa perlu dipanggil oleh program lain. Selanjutnya, terdapat perintah import "fmt" yang diperlukan agar program dapat menggunakan fungsi-fungsi input dan output standar seperti fmt.Scan untuk membaca input dari pengguna dan fmt.Println untuk menampilkan output ke layar. Di dalam fungsi main(), program memulai dengan mendeklarasikan sebuah variabel

bernama x yang bertipe data integer, yang akan digunakan untuk menyimpan nilai bilangan bulat yang dimasukkan oleh pengguna. Kemudian, program menggunakan fmt.Scan(&x) untuk meminta input dari pengguna, di mana nilai tersebut akan disimpan ke dalam variabel x.

Pada bagian selanjutnya, program menginisialisasi sebuah variabel string bernama h dengan nilai awal "positif", yang akan digunakan sebagai dasar untuk menentukan status bilangan. Program kemudian melakukan pemeriksaan kondisi dengan menggunakan pernyataan if  $x < 0$ , yang berarti jika nilai x kurang dari nol, maka variabel h akan diubah menjadi "bukan " + h, sehingga menghasilkan string "bukan positif". Jika kondisi tersebut tidak terpenuhi, yaitu jika x lebih besar atau sama dengan nol, maka h tetap bernilai "positif".

Akhirnya, program menampilkan hasil akhir dengan menggunakan fmt.Println(h), yang akan mencetak string yang menunjukkan apakah bilangan tersebut positif atau bukan positif berdasarkan nilai x yang dimasukkan. Dengan demikian, program ini berfungsi untuk menerima input bilangan bulat tunggal dari pengguna, mengevaluasi apakah bilangan tersebut positif atau tidak, dan menampilkan hasil evaluasi tersebut secara langsung, sesuai dengan contoh masukan dan keluaran yang diberikan, di mana bilangan positif didefinisikan sebagai bilangan yang lebih besar dari nol, sedangkan nol dan bilangan negatif dikategorikan sebagai bukan positif.

### 3. Guided 3

#### Source Code

```
package main
import "fmt"
func main() {
    var a int
    fmt.Scan(&a)
    x := false
    if a%2==0&&a<0 {
        x = true
    }
    fmt.Println(x)
}
```

## Screenshot Program

The screenshot shows a Go workspace in a code editor. The Explorer sidebar lists files like G1.go, G2.go, G3.go, T1.go, T2.go, T3.go, and G3.go. The main.go file is open in the editor, containing the following Go code:

```
package main
import "fmt"
func main() {
    var a int
    fmt.Scan(&a)
    x := false
    if a%2==0&&a<0 {
        x = true
    }
    fmt.Println(x)
}
```

The terminal window shows the output of running the program with input 'NIM:189902590019A KELAS:S1IF-13-AK Nama Mahasiswa: Rafa Alifiansyah':

```
NIM:189902590019A
KELAS:S1IF-13-AK
Nama Mahasiswa: Rafa Alifiansyah
```

The bottom status bar indicates the code is saved.

## Deskripsi Program

Program ini berfungsi untuk menentukan apakah suatu bilangan bulat yang dimasukkan oleh pengguna merupakan bilangan genap negatif atau bukan. Program dimulai dengan deklarasi package main yang menunjukkan bahwa kode ini merupakan program utama yang dapat dijalankan langsung oleh compiler Go. Setelah itu terdapat perintah import "fmt" yang digunakan untuk memanggil pustaka standar Go bernama fmt, yang menyediakan fungsi-fungsi untuk melakukan proses input dan output data, seperti membaca nilai dari pengguna dan menampilkan hasil ke layar.

Di dalam fungsi utama func main(), program mendeklarasikan satu variabel bertipe integer bernama a yang digunakan untuk menampung nilai input dari pengguna. Nilai ini dibaca menggunakan perintah fmt.Scan(&a), di mana tanda & berfungsi untuk menyimpan data yang dimasukkan langsung ke alamat memori variabel tersebut. Selain itu, juga terdapat variabel x bertipe boolean yang diinisialisasi dengan nilai awal false. Variabel ini nantinya akan digunakan untuk menyimpan hasil dari proses logika yang menentukan apakah bilangan yang dimasukkan merupakan bilangan genap negatif atau tidak.

Setelah data berhasil dimasukkan, program melakukan pemeriksaan kondisi menggunakan struktur kendali if a%2 == 0 && a < 0. Kondisi ini terdiri dari dua syarat yang dihubungkan dengan operator logika &&, yang berarti kedua syarat harus bernilai benar agar pernyataan di dalam if dijalankan. Syarat pertama, a%2 == 0, digunakan untuk mengecek apakah bilangan tersebut genap, yaitu jika sisa hasil bagi a dengan 2

sama dengan nol. Syarat kedua,  $a < 0$ , memastikan bahwa bilangan tersebut bernilai negatif. Jika kedua syarat tersebut terpenuhi, maka program akan mengeksekusi perintah  $x = \text{true}$ , yang berarti nilai variabel  $x$  diubah menjadi true. Jika salah satu atau kedua syarat tidak terpenuhi, maka variabel  $x$  tetap bernilai false.

Setelah proses pemeriksaan selesai, program menampilkan hasil akhir ke layar menggunakan perintah `fmt.Println(x)`. Nilai `true` menandakan bahwa bilangan yang dimasukkan merupakan bilangan genap sekaligus negatif, sedangkan nilai `false` menunjukkan bahwa bilangan tersebut tidak memenuhi kedua kriteria tersebut

# TUGAS

## 1. Tugas 1

### Source Code

```
package main

import "fmt"

func main() {
    var x, h int
    fmt.Scan(&x)
    h = x/2
    if x%2 == 1 {
        h = x/2 + x%2
    }
    fmt.Println(h)
}
```

### Screenshot Program

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** Shows files in the workspace, including G1.go, G2.go, G3.go, T1.go, T2.go, T3.go, and Modul 9.
- Editor View:** Displays the code for T1.go.
- Terminal View:** Shows the command "go run" being entered and the resulting output:

```
PS C:\Modul 9> go run "c:\Modul 9\T1.go"
10
5
PS C:\Modul 9> go run "c:\Modul 9\T1.go"
1
1
PS C:\Modul 9> go run "c:\Modul 9\T1.go"
25
13
PS C:\Modul 9> go run "c:\Modul 9\T1.go"
9
5
PS C:\Modul 9>
```

### Deskripsi Program

Program ini digunakan untuk menghitung hasil pembagian suatu bilangan bulat dengan dua, dengan memperhatikan kondisi khusus apabila bilangan tersebut merupakan bilangan ganjil. Program diawali dengan deklarasi package main yang menandakan bahwa kode ini merupakan program utama yang dapat langsung dijalankan. Kemudian terdapat perintah import "fmt" yang berfungsi untuk memanggil

pustaka standar Go, yaitu fmt, yang digunakan dalam proses input dan output seperti membaca nilai dari pengguna dan menampilkan hasil perhitungan ke layar.

Di dalam fungsi utama func main(), program mendeklarasikan dua variabel bertipe integer, yaitu x dan h. Variabel x digunakan untuk menyimpan nilai masukan dari pengguna, sedangkan h digunakan untuk menyimpan hasil perhitungan. Nilai untuk variabel x dibaca menggunakan perintah fmt.Scan(&x), di mana tanda & menunjukkan bahwa nilai input akan langsung disimpan ke alamat memori dari variabel tersebut.

Setelah pengguna memasukkan nilai, program melakukan operasi awal  $h = x / 2$ . Operasi ini akan membagi bilangan x dengan dua menggunakan pembagian bilangan bulat, sehingga hasilnya akan dibulatkan ke bawah apabila x merupakan bilangan ganjil. Namun, untuk memastikan hasil pembagian tetap sesuai dengan logika yang diinginkan, program kemudian memeriksa kondisi dengan pernyataan if  $x \% 2 == 1$ . Kondisi ini digunakan untuk mengecek apakah bilangan x merupakan bilangan ganjil, karena bilangan ganjil akan menghasilkan sisa bagi satu jika dibagi dua.

Jika kondisi tersebut benar, maka dilakukan operasi  $h = x/2 + x\%2$ , di mana hasil pembagian ditambah dengan sisa baginya, sehingga hasil akhir menjadi pembulatan ke atas dari pembagian dua bilangan bulat ganjil. Dengan demikian, ketika nilai x adalah bilangan ganjil, hasil h akan bernilai sedikit lebih besar daripada pembagian biasa, sementara jika x genap, nilai h tetap hasil dari  $x/2$ .

Terakhir, program menampilkan nilai h ke layar menggunakan perintah fmt.Print(h).

## 2. Tugas 2

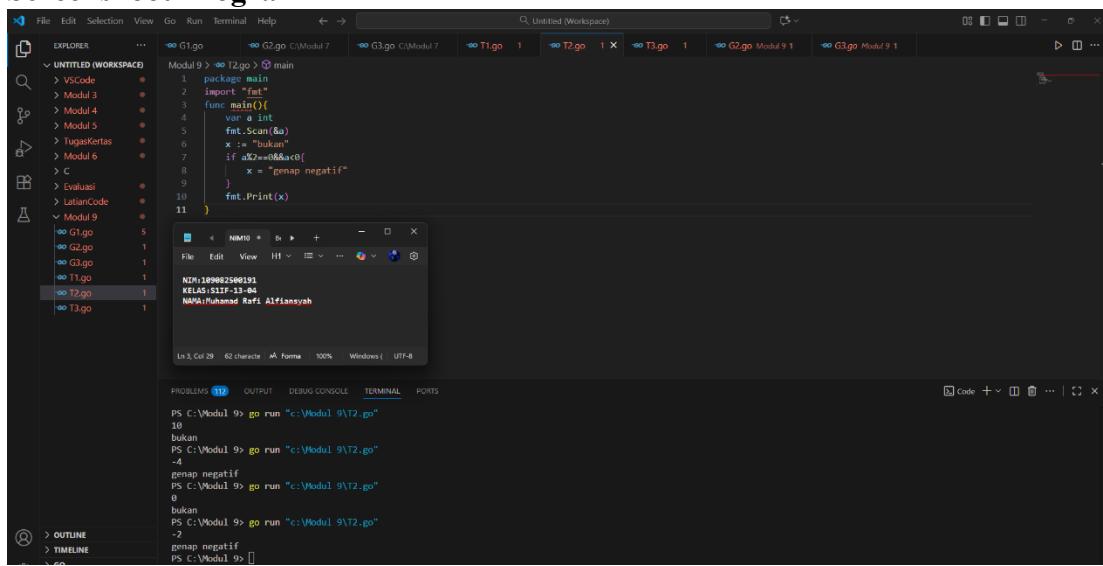
### Source Code

```
package main

import "fmt"

func main() {
    var a int
    fmt.Scan(&a)
    x := "bukan"
    if a%2==0&&a<0{
        x = "genap negatif"
    }
    fmt.Println(x)
}
```

### Screenshot Program



### Deskripsi Program

Program ini dibuat untuk menentukan apakah sebuah bilangan bulat yang dimasukkan oleh pengguna merupakan bilangan genap negatif atau bukan. Program diawali dengan deklarasi package main, yang menandakan bahwa kode tersebut merupakan program utama yang dapat dijalankan langsung oleh compiler Go. Setelah itu, terdapat perintah import "fmt" yang berfungsi untuk memanggil pustaka standar Go bernama fmt. Pustaka ini menyediakan fungsi untuk melakukan input dan output, seperti membaca data dari pengguna dengan fmt.Scan serta menampilkan hasil ke layar dengan fmt.Println.

Di dalam fungsi utama func main(), program mendeklarasikan satu variabel bertipe integer bernama a yang digunakan untuk menyimpan nilai input dari pengguna. Nilai ini dimasukkan melalui perintah fmt.Scan(&a), di mana tanda & menandakan bahwa data yang dimasukkan akan disimpan langsung ke alamat memori variabel a. Selain itu, terdapat juga variabel x yang dideklarasikan sebagai string dan diberi nilai awal "bukan". Nilai ini nantinya akan berubah jika kondisi tertentu terpenuhi.

Setelah pengguna memasukkan nilai, program melakukan pemeriksaan menggunakan pernyataan if  $a \% 2 == 0 \ \&\& \ a < 0$ . Kondisi ini memeriksa dua hal sekaligus dengan operator logika &&, yang berarti kedua kondisi harus benar agar perintah di dalam blok if dijalankan. Syarat pertama,  $a \% 2 == 0$ , digunakan untuk mengecek apakah bilangan tersebut genap, yakni ketika sisa hasil bagi a dengan 2 sama dengan nol. Syarat kedua,  $a < 0$ , digunakan untuk memastikan bahwa bilangan tersebut bernilai negatif. Jika kedua kondisi terpenuhi, maka program akan mengeksekusi perintah `x = "genap negatif"`, yang berarti variabel x diubah menjadi teks yang menunjukkan bahwa bilangan tersebut merupakan bilangan genap negatif.

Apabila salah satu dari kondisi tersebut tidak terpenuhi, maka nilai x tidak berubah dan tetap "bukan", yang menandakan bahwa bilangan yang dimasukkan bukan merupakan bilangan genap negatif. Setelah proses pemeriksaan selesai, program menampilkan hasil akhir ke layar menggunakan perintah `fmt.Print(x)`.

Dengan demikian, ketika pengguna memasukkan bilangan seperti -2 atau -4, program akan menampilkan "genap negatif", sedangkan jika bilangan yang dimasukkan adalah 10, -3, atau 0, maka hasil yang muncul adalah "bukan".

### 3. Tugas 3

#### Source code

```
package main

import "fmt"

func main() {
    var x,y int
    fmt.Scan(&x,&y)
    h := false
    g := false
    if y%x == 0 {
        h = true
    }
    fmt.Println(h)

    if x%y == 0 {
        g = true
    }
    fmt.Println(g)
}
```

#### Screenshot program

The screenshot shows a Visual Studio Code interface with a dark theme. On the left, the Explorer sidebar shows a file tree with several Go files (G1.go, G2.go, G3.go, T1.go, T2.go, T3.go) and a folder named 'Modul 9'. In the center, a code editor window displays the source code for T3.go. The code uses the 'fmt' package to scan for two integers from the user and then determine if they are factors of each other. The right side of the screen shows a terminal window with the output of running the program multiple times with different inputs. The bottom status bar indicates the file path as 'C:\Modul 9\T3.go'.

```
PS C:\Modul 9> go run "c:\Modul 9\T3.go"
10 5
false
true
PS C:\Modul 9> go run "c:\Modul 9\T3.go"
3 21
true
false
PS C:\Modul 9> go run "c:\Modul 9\T3.go"
4 4
true
true
PS C:\Modul 9>
```

#### Deskripsi Program

Program ini berfungsi untuk menentukan apakah suatu bilangan merupakan faktor dari bilangan lainnya. Program diawali dengan deklarasi package main, yang menandakan bahwa kode ini merupakan program utama yang dapat dijalankan secara

langsung. Setelah itu, terdapat perintah import "fmt" yang digunakan untuk memanggil pustaka standar Go bernama fmt. Pustaka ini berfungsi dalam proses input dan output, seperti membaca data dari pengguna serta menampilkan hasil ke layar.

Di dalam fungsi utama func main(), program mendeklarasikan dua variabel bertipe integer, yaitu x dan y, yang masing-masing digunakan untuk menyimpan dua bilangan masukan dari pengguna. Nilai kedua variabel ini dibaca melalui perintah fmt.Scan(&x, &y), di mana tanda & menunjukkan bahwa nilai yang dimasukkan akan langsung disimpan ke alamat memori dari variabel yang bersangkutan. Selain itu, program juga mendeklarasikan dua variabel boolean, yaitu h dan g, yang keduanya diinisialisasi dengan nilai false. Variabel h digunakan untuk menyimpan hasil pemeriksaan apakah x merupakan faktor dari y, sedangkan variabel g digunakan untuk memeriksa apakah y merupakan faktor dari x.

Selanjutnya, program melakukan pemeriksaan pertama dengan struktur kendali if  $y \% x == 0$ . Kondisi ini berarti program akan mengecek apakah sisa hasil bagi y dibagi dengan x sama dengan nol. Jika benar, maka x merupakan faktor dari y, dan variabel h diubah menjadi true. Setelah kondisi ini diperiksa, nilai dari h ditampilkan ke layar menggunakan fmt.Println(h), sehingga pengguna dapat mengetahui apakah x adalah faktor dari y.

Kemudian, program melanjutkan dengan pemeriksaan kedua menggunakan if  $x \% y == 0$ , yang bekerja dengan cara sebaliknya. Kondisi ini mengecek apakah y merupakan faktor dari x. Jika hasil bagi x dibagi y tidak menyisakan sisa, maka variabel g diubah menjadi true. Setelah itu, nilai dari g ditampilkan menggunakan fmt.Println(g).

Dengan demikian, program ini akan menampilkan dua nilai boolean. Nilai pertama (h) menunjukkan apakah x merupakan faktor dari y, sedangkan nilai kedua (g) menunjukkan apakah y merupakan faktor dari x. Misalnya, jika pengguna memasukkan 2 8, maka hasil yang muncul adalah true false, karena 2 adalah faktor dari 8, tetapi 8 bukan faktor dari 2.