

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 09
IF -THEN



Disusun oleh:

MUHAMMAD FAIZ MAULANA

109082500124

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)

}
```

Screenshoot program



```
soal2.go 1  Release Notes: 1.106.0  latsol1.go 1  soal3.go 1  CONSOAL1.go 3 X
CONSOAL1.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 6  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\User-PC\Documents\Semester 1\alpro\minggu9> go run "c:\Users\User-PC\Documents\Semester 1\alpro\minggu9\CONSOAL1.go"
10
10
PS C:\Users\User-PC\Documents\Semester 1\alpro\minggu9> go run "c:\Users\User-PC\Documents\Semester 1\alpro\minggu9\CONSOAL1.go"
-999
999
PS C:\Users\User-PC\Documents\Semester 1\alpro\minggu9>
```

Deskripsi program

Program Go tersebut membaca sebuah bilangan bulat dari input, kemudian jika bilangan itu negatif maka nilainya diubah menjadi kebalikannya (dikalikan -1) sehingga menjadi nilai absolut, dan akhirnya mencetak hasilnya; contohnya, masukan 10 akan dicetak 10, sedangkan masukan -999 akan diubah menjadi 999 lalu dicetak, yang diwujudkan dengan deklarasi variabel int, pemindaian input memakai `fmt.Scan`, percabangan `if` `bilangan < 0 { bilangan = -bilangan }`, dan output dengan `fmt.Println`.

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 {

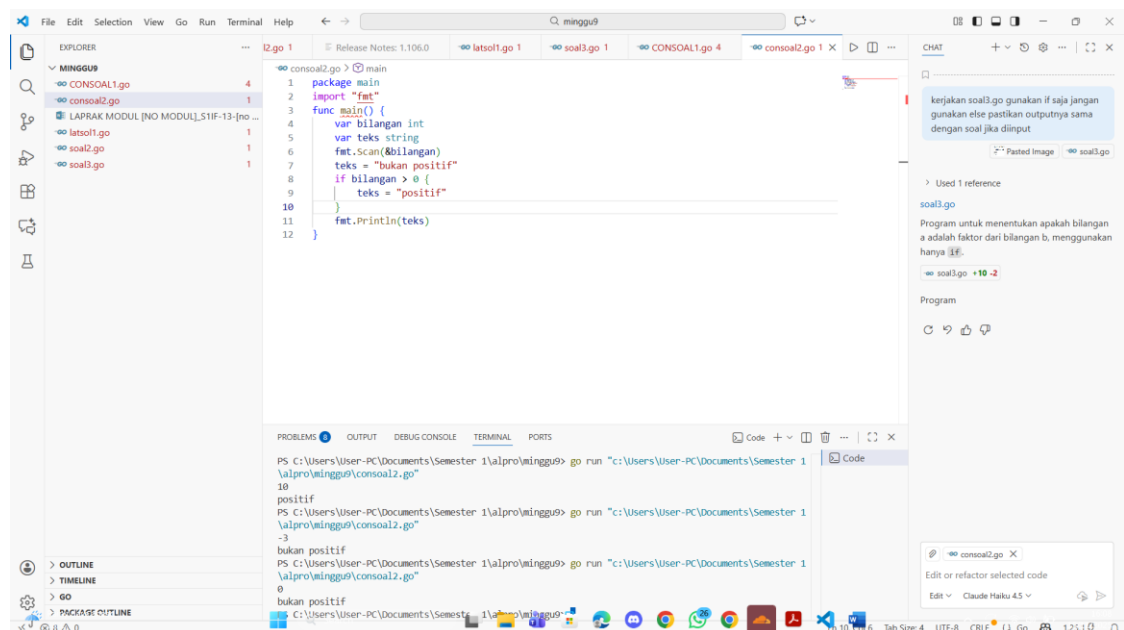
        teks = "positif"

    }

    fmt.Println(teks)

}
```

Screenshoot program



Deskripsi program

Program Go pada gambar membaca satu bilangan bulat dari input, lalu menentukan teks “positif” jika nilainya lebih besar dari nol, dan “bukan positif” untuk nol atau bilangan negatif, kemudian mencetak teks tersebut; implementasinya memakai deklarasi variabel bilangan (int) dan teks (string), pemindaian input dengan `fmt.Scan`, kondisi `if bilangan > 0 { teks = "positif" }` dengan nilai awal teks = "bukan positif", serta output `fmt.Println(teks)`, seperti terlihat dari hasil uji: masukan 10 menghasilkan “positif”, sedangkan -3 dan 0 menghasilkan “bukan positif”.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var bilangan int

    var hasil bool

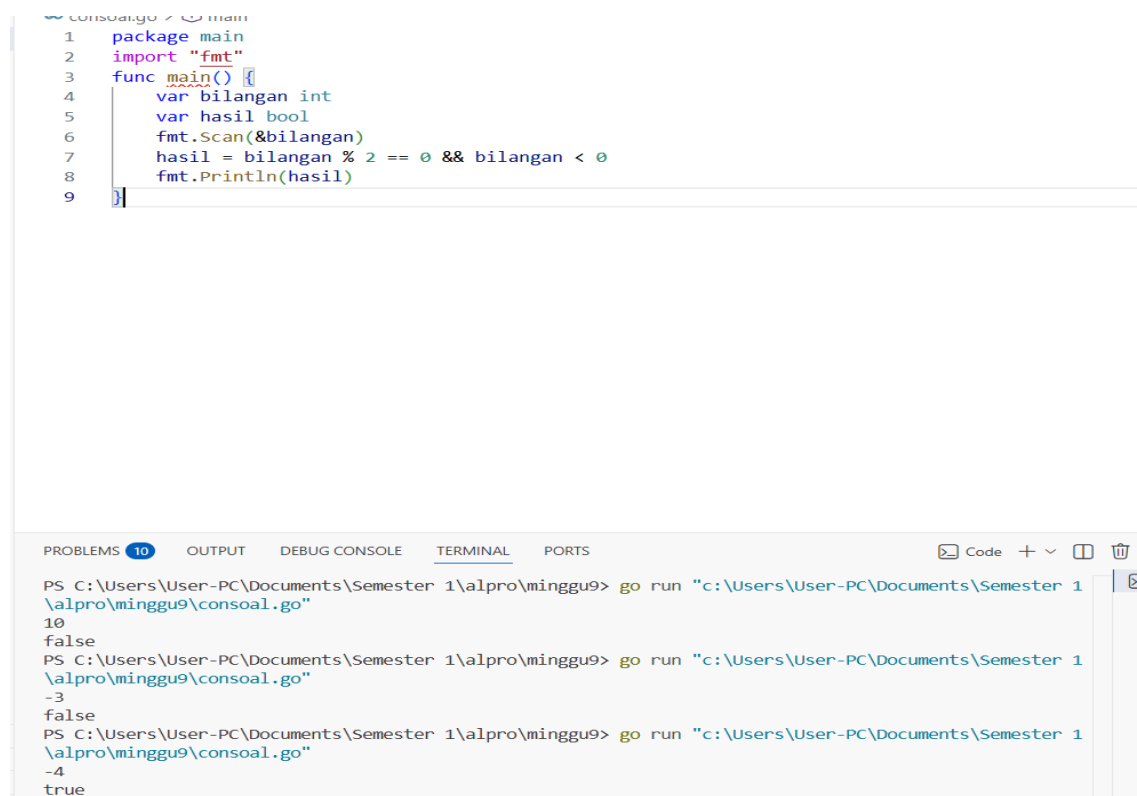
    fmt.Scan(&bilangan)

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

    fmt.Println(hasil)

}
```

Screenshoot program



The screenshot shows the Go program code in the editor and its execution results in the terminal. The code is as follows:

```
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 }
```

The terminal output shows the results of running the program with different inputs:

```
PS C:\Users\User-PC\Documents\Semester 1\alpro\minggu9> go run "c:\Users\User-PC\Documents\Semester 1\alpro\minggu9\consoal.go"
10
false
PS C:\Users\User-PC\Documents\Semester 1\alpro\minggu9> go run "c:\Users\User-PC\Documents\Semester 1\alpro\minggu9\consoal.go"
-3
false
PS C:\Users\User-PC\Documents\Semester 1\alpro\minggu9> go run "c:\Users\User-PC\Documents\Semester 1\alpro\minggu9\consoal.go"
-4
true
```

Deskripsi program

Program Go tersebut membaca sebuah bilangan bulat dari input, lalu menghitung nilai boolean bernama hasil yang akan bernilai true hanya jika bilangan genap sekaligus negatif, yaitu dengan ekspresi logika bilangan $\% 2 == 0 \ \&\& \text{bilangan} < 0$, dan akhirnya mencetak nilai hasil ke layar; karena itu input 10 menghasilkan false (genap tapi tidak negatif), input -3 menghasilkan false (negatif tapi tidak genap), sedangkan input -4 menghasilkan true (genap dan negatif).

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var a, b, c int

    fmt.Scan(&a)

    b = a / 2

    c = a % 2

    if c != 0 {

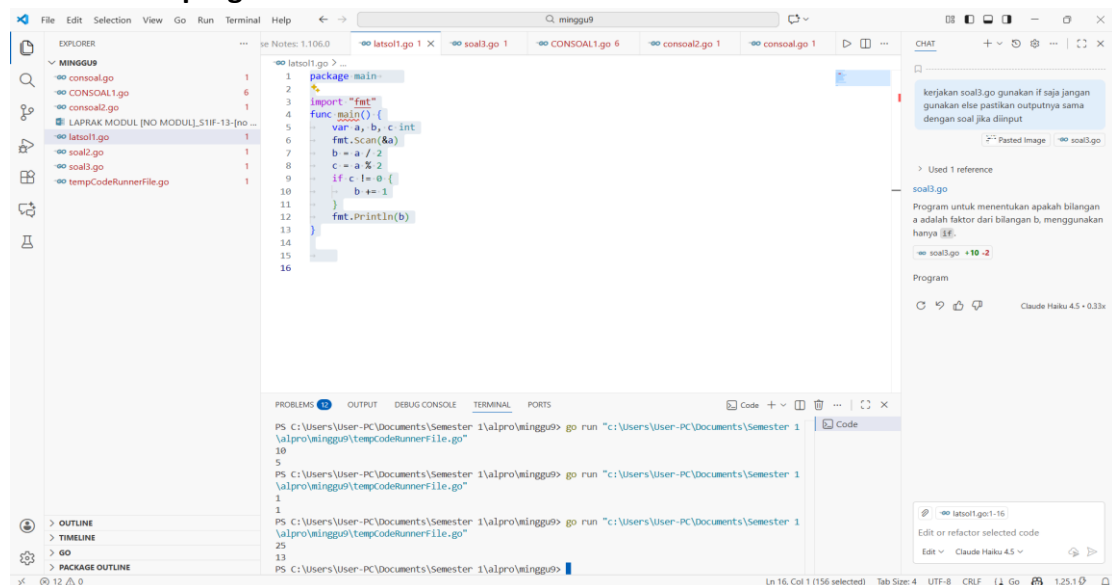
        b += 1

    }

    fmt.Println(b)

}
```

Screenshoot program



Deskripsi program

Program Go pada gambar membaca sebuah nilai integer a , lalu menghitung dua variabel b dan c berdasarkan a : b diisi $a/2$, c diisi $a\%2$ (sisa bagi 2) untuk mendeteksi genap-ganjil; jika c tidak sama dengan 0 (berarti a ganjil) maka b ditambah 1, sehingga b menjadi hasil pembulatan ke atas dari $a/2$ untuk bilangan ganjil, sedangkan tetap $a/2$ untuk bilangan genap, dan akhirnya program mencetak nilai b ke layar, yang terlihat pada uji coba input 10 menghasilkan 5, input 1 menghasilkan 1, dan input 25 menghasilkan 13.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var a int

    fmt.Scan(&a)

    if a%2 == 0 && a < 0 {

        fmt.Println("genap negatif")

    }

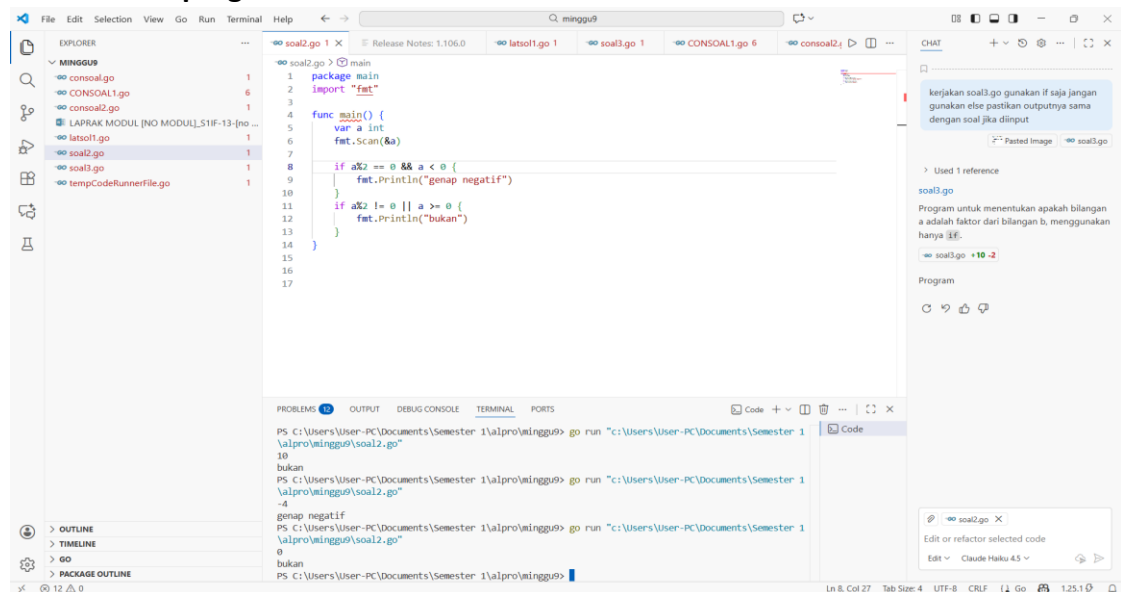
    if a%2 != 0 || a >= 0 {

        fmt.Println("bukan")

    }

}
```

Screenshoot program



Deskripsi program

Program Go tersebut membaca sebuah bilangan bulat a dari input, lalu mengevaluasi dua kondisi terpisah: jika a genap dan negatif maka mencetak genap negatif, sedangkan jika a ganjil atau tidak negatif maka mencetak bukan; karena kedua pemeriksaan ditulis sebagai dua blok if terpisah (bukan if-else), nilai tertentu bisa memicu satu atau dua output—contohnya $a = -4$ hanya memenuhi kondisi pertama sehingga mencetak genap negatif, $a = -3$ hanya memenuhi kondisi kedua sehingga mencetak bukan, sedangkan $a = 0$ memenuhi kondisi kedua sehingga mencetak bukan, dan untuk a genap positif (misalnya 8) juga hanya mencetak bukan

3. Tugas 3

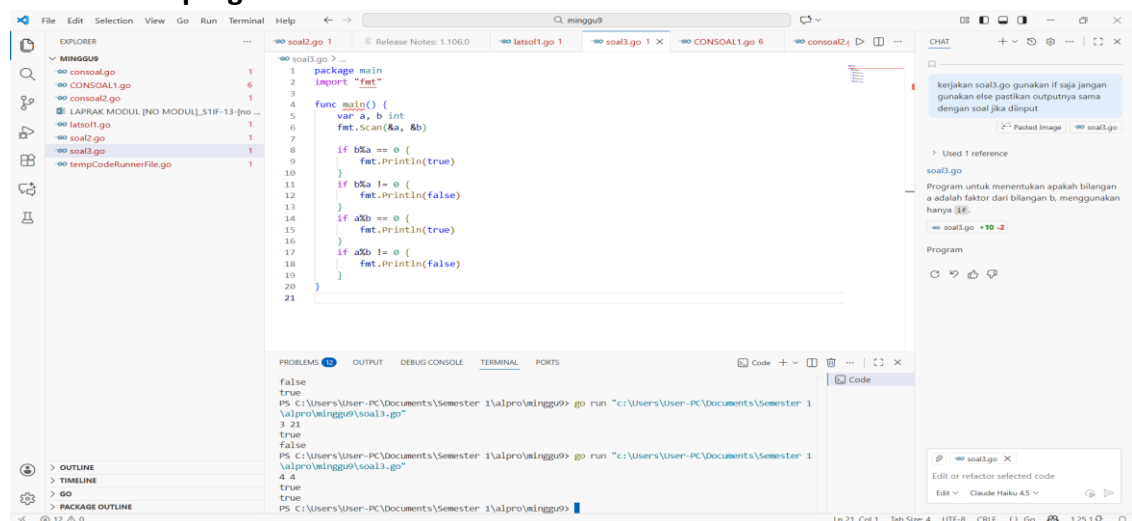
Source code

```
package main
import "fmt"

func main() {
    var a, b int
    fmt.Scan(&a, &b)

    if b%a == 0 {
        fmt.Println(true)
    }
    if b%a != 0 {
        fmt.Println(false)
    }
    if a%b == 0 {
        fmt.Println(true)
    }
    if a%b != 0 {
        fmt.Println(false)
    }
}
```

Screenshoot program



Deskripsi program

Program Go pada gambar membaca dua bilangan bulat a dan b , lalu memeriksa faktor menggunakan empat blok if terpisah: pertama mencetak true jika b habis membagi a ($b\%a == 0$) dan false jika tidak, kemudian mencetak true jika a habis membagi b ($a\%b == 0$) dan false jika tidak; karena tidak memakai if-else, setiap pasangan kondisi menghasilkan dua baris keluaran yang saling melengkapi—misalnya untuk $a=3$, $b=21$ program akan menampilkan true lalu false untuk pemeriksaan $b\%a$, dan true lalu false atau sebaliknya untuk pemeriksaan $a\%b$ tergantung hasil bagi, sehingga pengguna dapat melihat status pembagi dari kedua arah secara eksplisit.