

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL 9

IF-THEN



Disusun oleh:

RAFIF ARFA DHIAPRAJA

109082500041

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 {

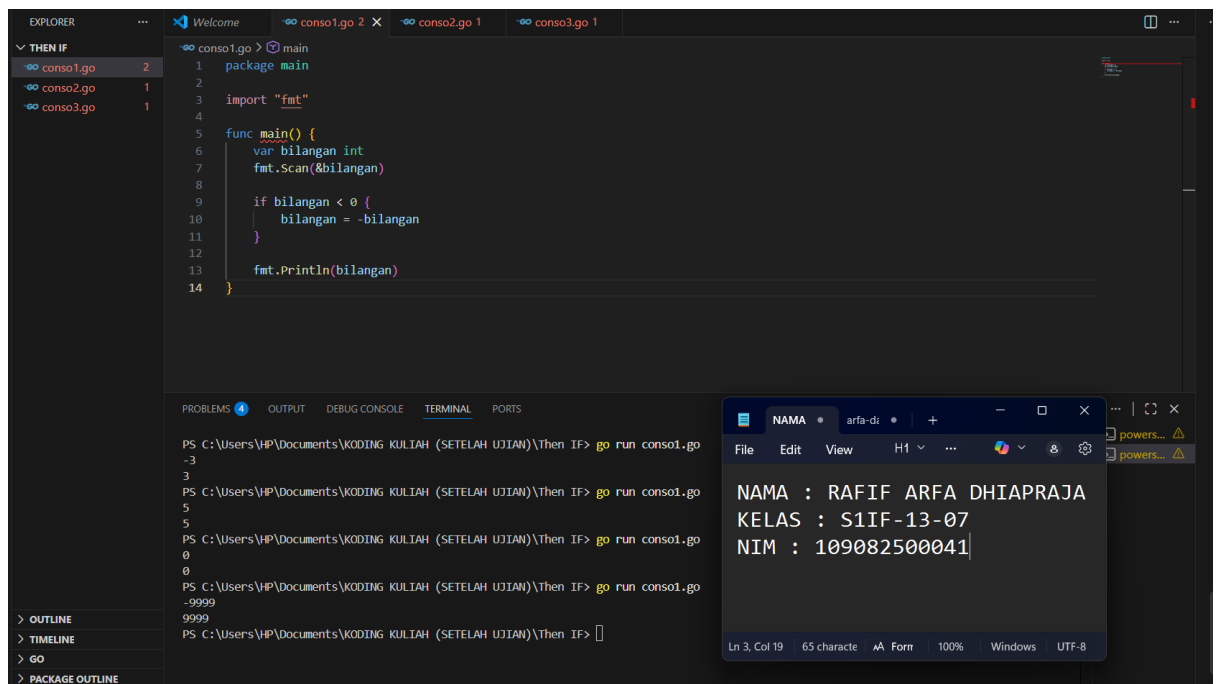
        bilangan = -bilangan

    }

    fmt.Println(bilangan)

}
```

Screenshoot program



Deskripsi program

Pada program ini, saya membuat program untuk menghitung nilai absolut atau nilai mutlak dari sebuah bilangan. Pertama, saya mendeklarasikan satu variabel bernama `bilangan` yang digunakan untuk menyimpan nilai yang dimasukkan oleh pengguna. Setelah itu, program meminta pengguna memasukkan sebuah bilangan bulat. Nilai tersebut kemudian dicek menggunakan kondisi `if`. Pada bagian ini, program memeriksa apakah bilangan yang dimasukkan bernilai negatif. Jika bilangan kurang dari 0, maka program mengubah nilainya menjadi positif dengan cara memberikan nilai `-bilangan` ke variabel yang sama. Jika bilangan yang dimasukkan sudah positif, maka kondisi `if` tidak dijalankan, sehingga nilainya tetap. Setelah proses pengecekan selesai, program menampilkan bilangan tersebut sebagai nilai absolut dengan menggunakan `fmt.Println(bilangan)`.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
```

```

var bilangan int

var hasil string

fmt.Scan(&bilangan)

hasil = "bukan positif"

if bilangan > 0 {

    hasil = "positif"

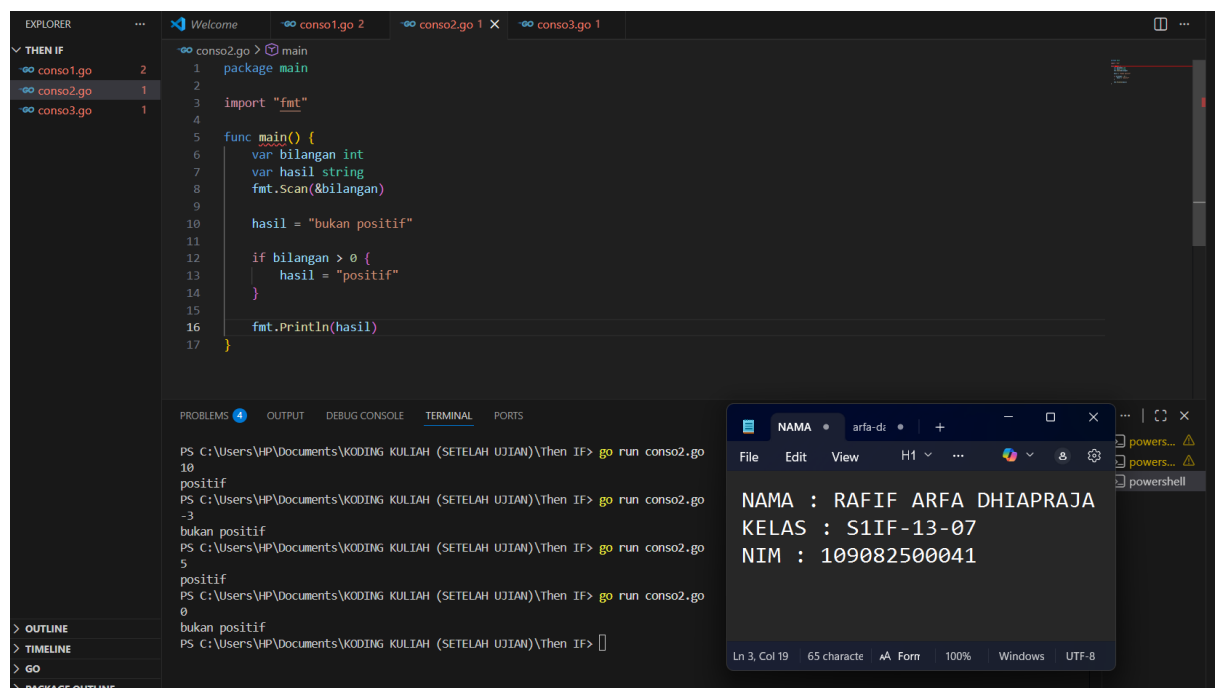
}

fmt.Println(hasil)

}

```

Screenshoot program



Deskripsi program

Pada program ini, saya membuat program untuk menentukan apakah suatu bilangan termasuk positif atau bukan. Pertama, saya mendeklarasikan variabel bilangan untuk menyimpan angka yang diinput pengguna, dan variabel hasil untuk menyimpan teks

yang akan ditampilkan sebagai output. Setelah itu, program membaca sebuah bilangan dari input. Sebelum pengecekan, saya memberi nilai awal pada variabel hasil yaitu "bukan positif". Nilai ini digunakan sebagai kondisi default jika bilangan yang dimasukkan tidak lebih besar dari nol. Kemudian, program menggunakan struktur kontrol if untuk mengecek apakah bilangan lebih besar dari 0. Jika kondisi tersebut benar, maka variabel hasil diubah menjadi "positif". Setelah pengecekan selesai, program menampilkan isi variabel hasil menggunakan `fmt.Println`. Jadi, bila pengguna memasukkan bilangan positif, output yang muncul adalah "positif". Namun jika pengguna memasukkan bilangan nol atau bilangan negatif, output yang ditampilkan adalah "bukan positif".

3. Guided 3

Source Code

```
package main

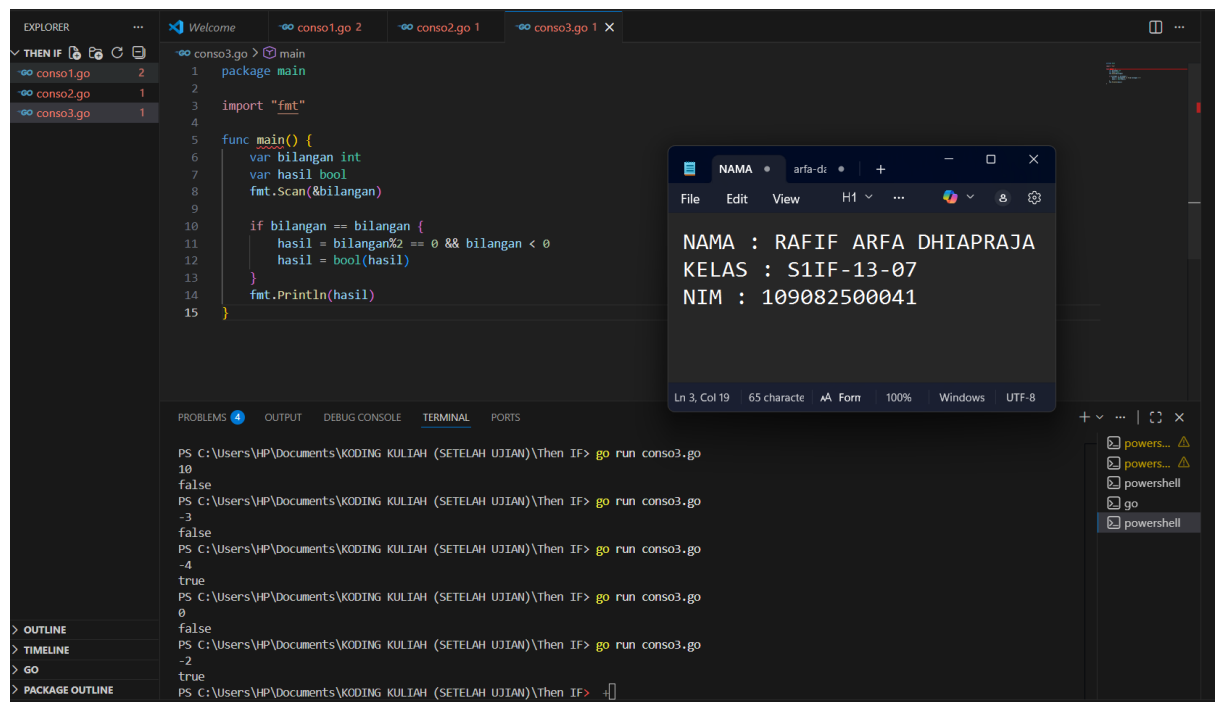
import "fmt"

func main() {
    var bilangan int
    var hasil bool
    fmt.Scan(&bilangan)

    if bilangan == bilangan {
        hasil = bilangan%2 == 0 && bilangan < 0
        hasil = bool(hasil)
    }

    fmt.Println(hasil)
}
```

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan int
7     var hasil bool
8     fmt.Scan(&bilangan)
9
10    if bilangan == bilangan {
11        hasil = bilangan%2 == 0 && bilangan < 0
12        hasil = bool(hasil)
13    }
14    fmt.Println(hasil)
15 }
```

```
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Then IF> go run conso3.go
10
false
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Then IF> go run conso3.go
-3
false
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Then IF> go run conso3.go
-4
true
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Then IF> go run conso3.go
0
false
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Then IF> go run conso3.go
-2
true
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Then IF> +
```

Deskripsi program

Pada program ini, saya membuat program untuk menentukan apakah suatu bilangan memenuhi dua kondisi sekaligus: yaitu bilangan tersebut harus genap dan bernilai negatif. Pertama, saya membuat dua variabel, yaitu bilangan untuk menyimpan input dari pengguna dan hasil untuk menyimpan nilai boolean yang akan menjadi output. Setelah pengguna memasukkan sebuah bilangan, program masuk ke dalam sebuah blok if. Kondisi pada if sebenarnya selalu benar karena membandingkan bilangan dengan dirinya sendiri. Di dalam blok tersebut, variabel hasil diisi dengan ekspresi logika `bilangan%2 == 0 && bilangan < 0`. Ekspresi ini akan menghasilkan nilai benar jika bilangan tersebut genap dan kurang dari nol. Jika salah satu syarat tidak terpenuhi, maka hasil akan bernilai false. Setelah pengecekan selesai, program mencetak nilai boolean hasil menggunakan `fmt.Println`. Jadi, jika pengguna memasukkan bilangan genap negatif seperti -4 atau -10, maka output akan bernilai true. Tetapi untuk bilangan lain seperti bilangan positif, ganjil, atau nol, outputnya akan bernilai false.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var orang, motor int

    fmt.Scan(&orang)

    motor = orang / 2

    if orang % 2 != 0 {

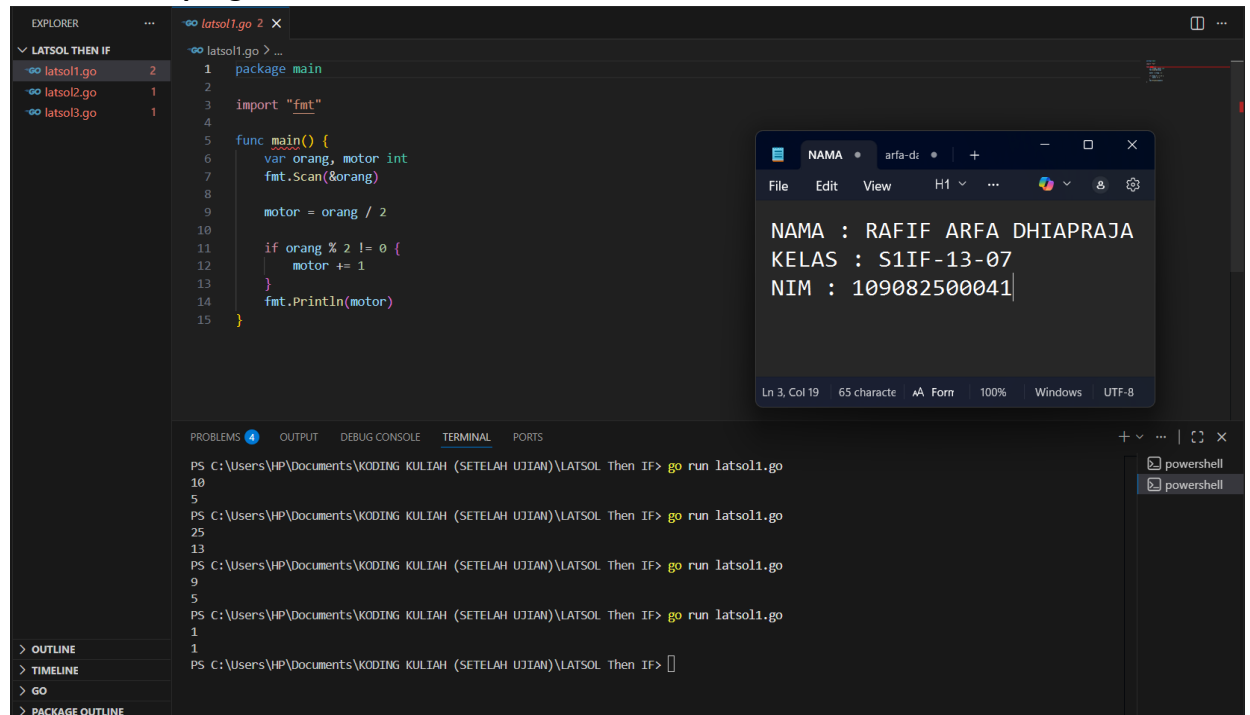
        motor += 1

    }

    fmt.Println(motor)

}
```

Screenshoot program



Deskripsi program

Pada program ini, saya membuat program untuk menghitung jumlah motor yang dibutuhkan untuk touring berdasarkan jumlah orang. Setiap motor bisa digunakan oleh dua orang, yaitu satu pengendara dan satu penumpang. Pertama, saya membuat dua variabel: orang untuk menyimpan jumlah peserta touring, dan motor untuk menyimpan jumlah motor yang akan digunakan. Setelah pengguna memasukkan jumlah orang, program langsung menghitung motor = orang / 2. Ini berarti setiap dua orang dipasangkan ke satu motor. Namun, jika jumlah orangnya ganjil, masih ada satu orang tersisa yang tetap memerlukan satu motor tambahan. Karena itu, saya menggunakan pengecekan if orang % 2 != 0. Jika kondisi ini benar, artinya jumlah orang ganjil, maka motor ditambah satu yaitu dengan rumus motor += 1. Setelah semua perhitungan selesai, program menampilkan jumlah motor yang dibutuhkan menggunakan fmt.Println(motor). Dengan cara ini, berapa pun jumlah orang yang diinputkan, program dapat menentukan jumlah motor secara tepat sesuai aturan satu motor untuk dua orang.

2. Tugas 2

Source code

```
package main
```



```

import "fmt"

func main() {

    var bilangan int

    var text string

    fmt.Scan(&bilangan)

    text = "bukan"

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

        text = "genap negatif"

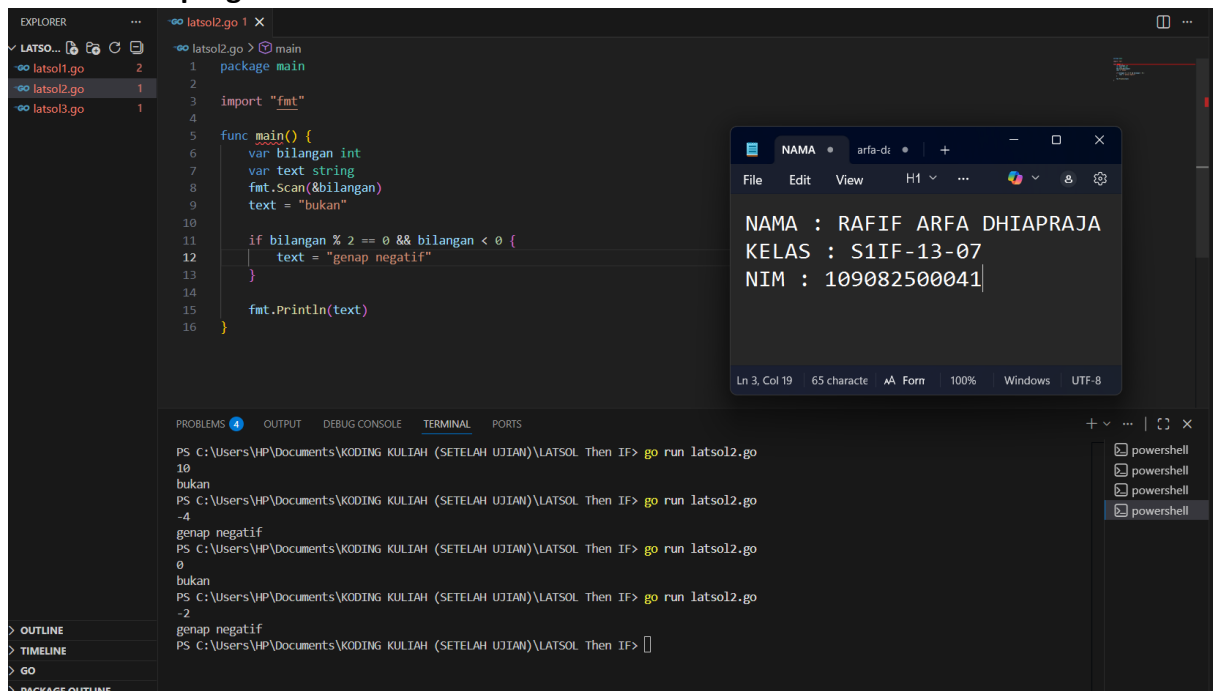
    }

    fmt.Println(text)

}

```

Screenshoot program



Deskripsi program

Program diatas dibuat untuk menyatakan suatu bilangan itu bernilai Genap Negatif atau bukan. Pertama saya mendeklarasi variabel bilangan yang berupa integer, dan text yang berupa string, lalu `fmt.Scan(&bilangan)` untuk membaca input sebuah bilangan dari pengguna. Kemudian saya isi variabel text dengan "bukan" gunanya untuk nilai default text, kemudian saya buat block if yaitu gunanya jika bilangan itu bernilai genap dan kurang dari nol text akan mengeluarkan nilai "genap negative", jadi jika if, dan jikalau blok if tidak terpenuhi maka blok if akan dilewati dan text akan mengisi nilai yang pertama tdi yaitu "bukan" jika kondisi if terpenuhi maka blok if akan dieksekusi dan nilai diganti "genap negative", dan terakhir saya taruh `fmt.Println` untuk menampilkan output text.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var a, b int
    var hasil1, hasil2 bool
    fmt.Scan(&a, &b)

    if b % a == 0 {
        hasil1 = true
    }
    fmt.Println(hasil1)

    if a % b == 0 {
        hasil2 = true
    }
    fmt.Println(hasil2)
}
```

Screenshoot program

The screenshot shows a Go program in VS Code. The Explorer pane on the left shows a project named 'LATSOL THEN IF' with three files: 'latsol1.go', 'latsol2.go', and 'latsol3.go'. The 'latsol3.go' file is selected and its content is displayed in the editor. The code defines a `main` function that takes two integers `a` and `b` as input. It uses `fmt.Scan(&a, &b)` to read the input. It then performs two conditional checks: first, `if b % a == 0`, which sets `hasil1 = true` if the condition is met; second, `if a % b == 0`, which sets `hasil2 = true` if the condition is met. Finally, it prints the results using `fmt.Println(hasil1)` and `fmt.Println(hasil2)`. The Terminal pane at the bottom shows the execution of the program, displaying the output: `10 5` followed by `false`, and `3 21` followed by `true` and `false`. A small window titled 'NAMA' is also visible, displaying the user's details: `NAMA : RAFIF ARFA DHIAPRAJA`, `KELAS : S1IF-13-07`, and `NIM : 109082500041`.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var a, b int
7     var hasil1, hasil2 bool
8     fmt.Scan(&a, &b)
9
10    if b % a == 0 {
11        hasil1 = true
12    }
13    fmt.Println(hasil1)
14
15    if a % b == 0 {
16        hasil2 = true
17    }
18    fmt.Println(hasil2)
19
20 }
```

```
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\LATSOL Then IF> go run latsol3.go
10 5
false
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\LATSOL Then IF> go run latsol3.go
3 21
true
false
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\LATSOL Then IF> go run latsol3.go
4 4
true
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\LATSOL Then IF>
```

Deskripsi program

Pada program ini, saya membuat program untuk menentukan apakah dua bilangan saling menjadi faktor satu sama lain. Program menggunakan dua variabel boolean, yaitu `hasil1` dan `hasil2`, untuk menyimpan hasil pengecekan. Pertama, pengguna memasukkan dua bilangan, yaitu `a` dan `b`. Setelah itu, program melakukan pengecekan pertama dengan kondisi `if b % a == 0`. Kondisi ini digunakan untuk memeriksa apakah `a` adalah faktor dari `b`. Jika kondisi tersebut benar, maka `hasil1` diubah menjadi `true`. Setelah pengecekan pertama selesai, program langsung mencetak nilai `hasil1`. Selanjutnya, program melakukan pengecekan kedua dengan kondisi `if a % b == 0`. Bagian ini bertujuan untuk memeriksa apakah `b` adalah faktor dari `a`. Jika benar, maka `hasil2` diubah menjadi `true`. Setelah pengecekan ini, program mencetak nilai `hasil2`. Dengan cara ini, program dapat menampilkan dua baris keluaran: baris pertama menyatakan apakah `a` merupakan faktor dari `b`, dan baris kedua menyatakan apakah `b` merupakan faktor dari `a`. Jika tidak memenuhi syarat, nilai boolean akan tetap `false`.