

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 10  
IF-THEN**



**Disusun oleh:**

**Tasyifa`ul hana**

**109082500212**

**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"

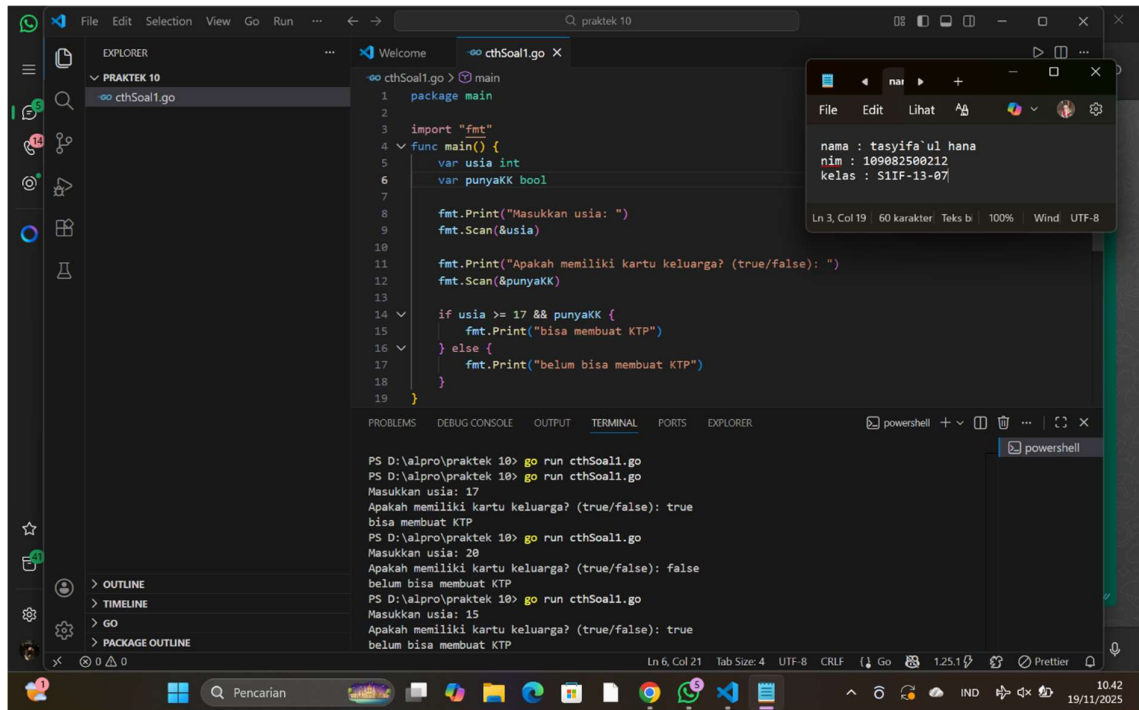
func main() {
    var usia int
    var punyaKK bool

    fmt.Print("Masukkan usia: ")
    fmt.Scan(&usia)

    fmt.Print("Apakah memiliki kartu keluarga? (true/false): ")
    fmt.Scan(&punyaKK)

    if usia >= 17 && punyaKK {
        fmt.Print("bisa membuat KTP")
    } else {
        fmt.Print("belum bisa membuat KTP")
    }
}
```

## Screenshoot program



The screenshot shows a Go program in a file named `cthSoal1.go` within a workspace named `PRAKTEK 10`. The program prompts the user for their age and whether they have a family card, then checks if they are eligible for a KTP based on these inputs.

```
1 package main
2
3 import "fmt"
4 func main() {
5     var usia int
6     var punyaKK bool
7
8     fmt.Print("Masukkan usia: ")
9     fmt.Scan(&usia)
10
11     fmt.Print("Apakah memiliki kartu keluarga? (true/false): ")
12     fmt.Scan(&punyaKK)
13
14     if usia >= 17 && punyaKK {
15         fmt.Print("bisa membuat KTP")
16     } else {
17         fmt.Print("belum bisa membuat KTP")
18     }
19 }
```

The terminal output shows three test runs:

```
PS D:\alpro\praktek 10> go run cthSoal1.go
PS D:\alpro\praktek 10> go run cthSoal1.go
Masukkan usia: 17
Apakah memiliki kartu keluarga? (true/false): true
bisa membuat KTP
PS D:\alpro\praktek 10> go run cthSoal1.go
Masukkan usia: 20
Apakah memiliki kartu keluarga? (true/false): false
belum bisa membuat KTP
PS D:\alpro\praktek 10> go run cthSoal1.go
Masukkan usia: 15
Apakah memiliki kartu keluarga? (true/false): true
belum bisa membuat KTP
```

## Deskripsi program

Program diatas meminta pengguna untuk meminta dua input dari pengguna, yaitu usia (angka) dan kk (bernilai true/false yang menunjukkan apakah memiliki Kartu Keluarga). Setelah menerima input, program mengecek apakah usia pengguna minimal 17 tahun dan memiliki KK. Jika kedua syarat itu terpenuhi, program akan menampilkan pesan "bisa membuat KTP", sedangkan jika salah satu syarat tidak terpenuhi, program menampilkan "belum bisa membuat KTP". Program ini pada dasarnya digunakan untuk menentukan apakah seseorang memenuhi syarat pembuatan KTP berdasarkan usia dan kepemilikan KK.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {

    var huruf string

    fmt.Print("Masukan satu huruf: ")

    fmt.Scan(&huruf)


    if huruf == "A" || huruf == "I" || huruf == "U" ||
huruf == "E" || huruf == "O" {

        fmt.Print("vokal")

    } else if (huruf >= "A" && huruf <= "Z") {

        fmt.Print("konsonan")

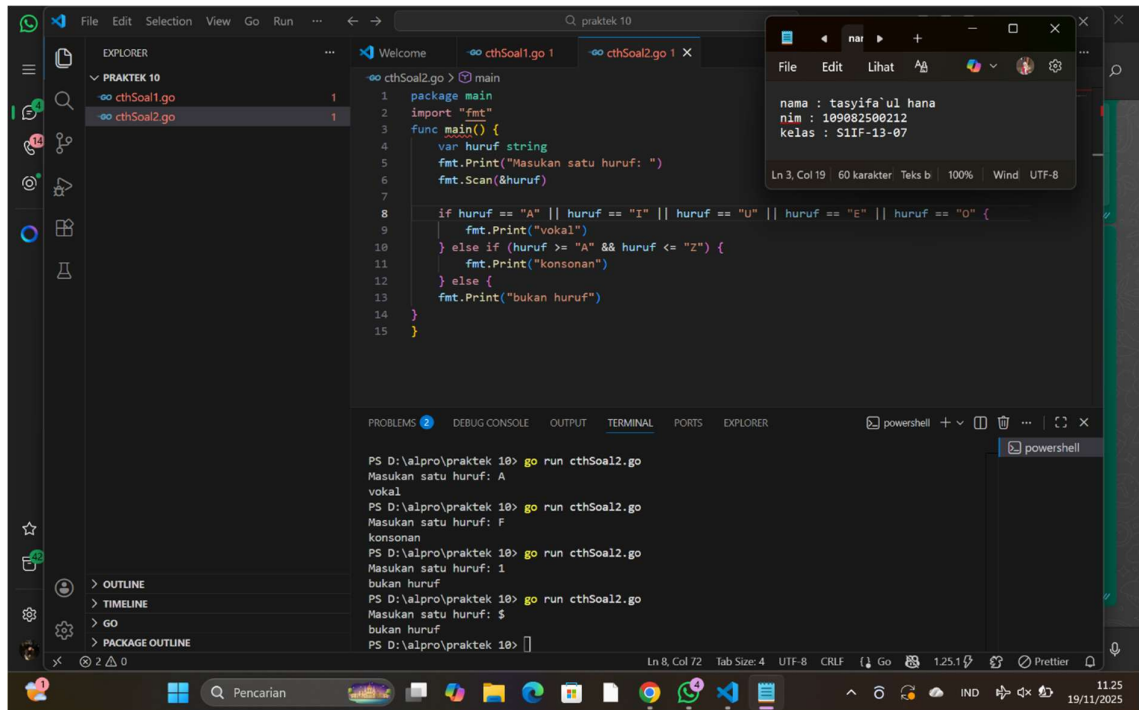
    } else {

        fmt.Print("bukan huruf")

    }

}
```

## Screenshoot program



```
package main
import "fmt"
func main() {
    var huruf string
    fmt.Print("Masukan satu huruf: ")
    fmt.Scan(&huruf)

    if huruf == "A" || huruf == "I" || huruf == "U" || huruf == "E" || huruf == "O" {
        fmt.Print("vokal")
    } else if (huruf >= "A" && huruf <= "Z") {
        fmt.Print("konsonan")
    } else {
        fmt.Print("bukan huruf")
    }
}
```

```
PS D:\alpro\praktek 10> go run cthSoal2.go
Masukan satu huruf: A
vokal
PS D:\alpro\praktek 10> go run cthSoal2.go
Masukan satu huruf: F
konsonan
PS D:\alpro\praktek 10> go run cthSoal2.go
Masukan satu huruf: 1
bukan huruf
PS D:\alpro\praktek 10> go run cthSoal2.go
Masukan satu huruf: $
bukan huruf
PS D:\alpro\praktek 10>
```

## Deskripsi Program

Program diatas meminta pengguna untuk membaca satu karakter dari pengguna lalu memeriksa apakah karakter itu termasuk huruf, kemudian menentukan apakah huruf tersebut adalah vokal atau konsonan. karakter dibandingkan dengan daftar huruf vokal (A, I, U, E, O). Jika karakter merupakan huruf dan termasuk vokal, program akan menampilkan "vokal"; jika karakter merupakan huruf tetapi bukan vokal, program menampilkan "konsonan"; namun jika karakter bukan huruf sama sekali, program menampilkan "bukan huruf". Program ini membantu mengidentifikasi jenis karakter yang dimasukkan pengguna dengan sederhana.

### 3. Guided 3

#### Source Code

```
package main

import "fmt"

func main() {

    var bilangan, d1, d2, d3, d4 int
    var teks string

    fmt.Print("Bilangan: ")

    fmt.Scan(&bilangan)

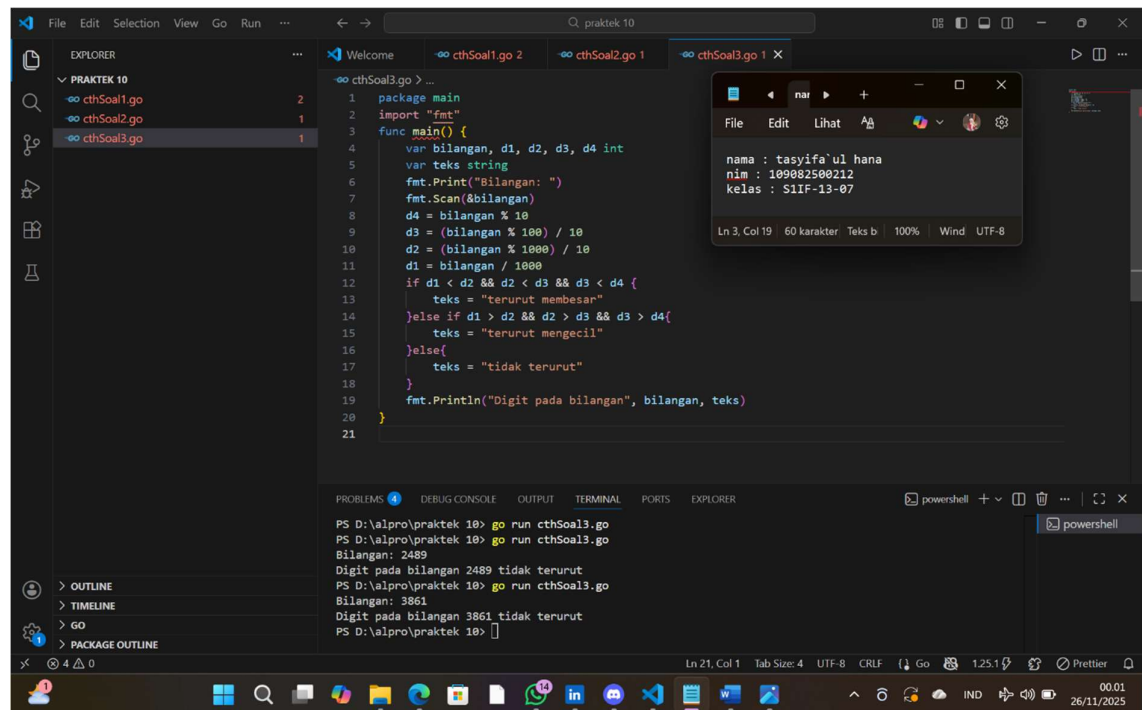
    d4 = bilangan % 10
    d3 = (bilangan % 100) / 10
    d2 = (bilangan % 1000) / 10
    d1 = bilangan / 1000

    if d1 < d2 && d2 < d3 && d3 < d4 {
        teks = "terurut membesar"
    }else if d1 > d2 && d2 > d3 && d3 > d4{
        teks = "terurut mengecil"
    }else{
        teks = "tidak terurut"
    }

    fmt.Println("Digit pada bilangan", bilangan, teks)

}
```

## Screenshoot program



```
1 package main
2 import "fmt"
3 func main() {
4     var bilangan, d1, d2, d3, d4 int
5     var teks string
6     fmt.Print("Bilangan: ")
7     fmt.Scan(&bilangan)
8     d4 = bilangan % 10
9     d3 = (bilangan % 100) / 10
10    d2 = (bilangan % 1000) / 100
11    d1 = bilangan / 1000
12    if d1 < d2 && d2 < d3 && d3 < d4 {
13        teks = "terurut membesar"
14    } else if d1 > d2 && d2 > d3 && d3 > d4 {
15        teks = "terurut mengecil"
16    } else {
17        teks = "tidak terurut"
18    }
19    fmt.Println("Digit pada bilangan", bilangan, teks)
20 }
21
```

nama : tasyifa'ul hana  
nim : 109082500212  
kelas : S1IF-13-07

PS D:\alpro\praktek 10> go run cthSoal3.go  
Bilangan: 2489  
Digit pada bilangan 2489 tidak terurut  
PS D:\alpro\praktek 10> go run cthSoal3.go  
Bilangan: 3861  
Digit pada bilangan 3861 tidak terurut  
PS D:\alpro\praktek 10>

## Deskripsi program

Program diatas meminta pengguna untuk memasukkan sebuah bilangan empat digit, lalu memisahkan setiap digitnya menjadi d1 (ribuan), d2 (ratusan), d3 (puluhan), dan d4 (satuan). Setelah digit-digit tersebut diambil, program membandingkannya untuk menentukan apakah susunan angkanya terurut membesar ( $d1 < d2 < d3 < d4$ ), terurut mengecil ( $d1 > d2 > d3 > d4$ ), atau tidak terurut jika tidak memenuhi kedua kondisi tersebut. Terakhir, program menampilkan hasil berupa digit bilangan tersebut beserta keterangan apakah digit-digitnya berurutan membesar, mengecil, atau tidak berurutan.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {

    var berat int

    fmt.Print("Berat parsel (gram): ")

    fmt.Scan(&berat)

    kg := berat / 1000
    sisa := berat % 1000

    biaya := kg * 10000

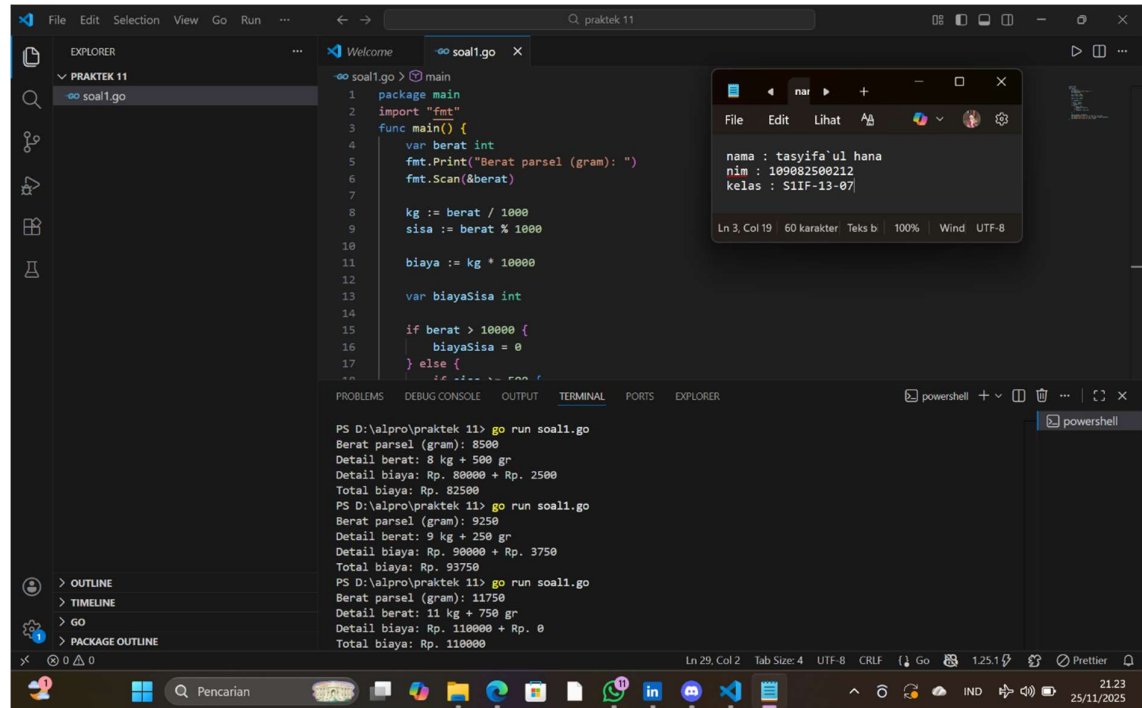
    var biayaSisa int

    if berat > 10000 {
        biayaSisa = 0
    } else {
        if sisa >= 500 {
            biayaSisa = sisa * 5
        } else {
            biayaSisa = sisa * 15
        }
    }

}
```



## Screenshoot program



```
1 package main
2 import "fmt"
3 func main() {
4     var berat int
5     fmt.Print("Berat parcel (gram): ")
6     fmt.Scan(&berat)
7
8     kg := berat / 1000
9     sisa := berat % 1000
10
11     biaya := kg * 10000
12
13     var biayaSisa int
14
15     if berat > 10000 {
16         biayaSisa = 0
17     } else {
```

```
nama : tasyifa'ul hana
nim : 109082500212
kelas : S1IF-13-07
```

```
PS D:\alpro\praktek 11> go run soal1.go
Berat parcel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS D:\alpro\praktek 11> go run soal1.go
Berat parcel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
PS D:\alpro\praktek 11> go run soal1.go
Berat parcel (gram): 11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 0
Total biaya: Rp. 110000
```

## Deskripsi program

Program diatas meminta pengguna untuk menghitung biaya pengiriman parcel berdasarkan berat yang dimasukkan dalam gram. Mula-mula, program mengubah berat gram menjadi kilogram dan sisa gram, lalu menghitung biaya kilogram dengan tarif Rp10.000 per kg. Setelah itu, program menentukan biaya untuk sisa gram: jika total berat lebih dari 10 kg maka sisa gram tidak dikenakan biaya, tetapi jika 10 kg atau kurang, sisa yang  $\geq 500$  gram dikenakan Rp5 per gram, sedangkan sisa yang  $< 500$  gram dikenakan Rp15 per gram. Terakhir, program menjumlahkan biaya kilogram dan biaya sisa, kemudian menampilkan rincian berat serta total biaya yang harus dibayar.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {

    var nam float64

    var nmk string


    fmt.Print("Nilai akhir mata kuliah: ")

    fmt.Scan(&nam)


    if nam > 80 {

        nmk = "A"

    } else if nam > 72.5 {

        nmk = "AB"

    } else if nam > 65 {

        nmk = "B"

    } else if nam > 57.5 {

        nmk = "BC"

    } else if nam > 50 {

        nmk = "C"

    } else if nam > 40 {

        nmk = "D"

    } else {

        nmk = "E"

    }

}
```

## Screenshoot program

```
1 package main
2 import "fmt"
3 func main() {
4     var nam float64
5     var nmk string
6
7     fmt.Print("Nilai akhir mata kuliah: ")
8     fmt.Scan(&nam)
9
10
11     if nam > 80 {
12         nmk = "A"
13     } else if nam > 72.5 {
14         nmk = "AB"
15     } else if nam > 65 {
16         nmk = "B"
17     } else if nam > 57.5 {
18         nmk = "BC"
19     } else if nam > 50 {
20         nmk = "C"
21     } else if nam > 40 {
22         nmk = "D"
23     } else {
24         nmk = "E"
25     }
26 }
```

PS D:\valpro\praktek 11> go run soal2.go  
Nilai akhir mata kuliah: 93.5  
Nilai mata kuliah: A  
PS D:\valpro\praktek 11> go run soal2.go  
Nilai akhir mata kuliah: 7.6  
Nilai mata kuliah: E  
PS D:\valpro\praktek 11> go run soal2.go  
Nilai akhir mata kuliah: 70.6  
Nilai mata kuliah: B

nama : tasyifa'ul hana  
nim : 109082500212  
kelas : S1IF-13-07

## Deskripsi program

Program diatas meminta pengguna untuk menentukan nilai huruf mata kuliah berdasarkan nilai akhir yang dimasukkan pengguna. Setelah pengguna mengetikkan nilai akhirnya, program membandingkan nilai tersebut dengan beberapa rentang nilai untuk menentukan kategori huruf, mulai dari A untuk nilai di atas 80, AB untuk nilai di atas 72.5, B untuk di atas 65, BC untuk di atas 57.5, C untuk di atas 50, D untuk di atas 40, dan E untuk nilai 40 ke bawah. Setelah kategori nilai ditentukan, program menampilkan hasil berupa nilai huruf yang sesuai dengan nilai akhir tersebut.

### 3. Tugas 3

#### Source code

```
package main
import "fmt"
func main() {
    var b int
    fmt.Print("Bilangan: ")
    fmt.Scan(&b)

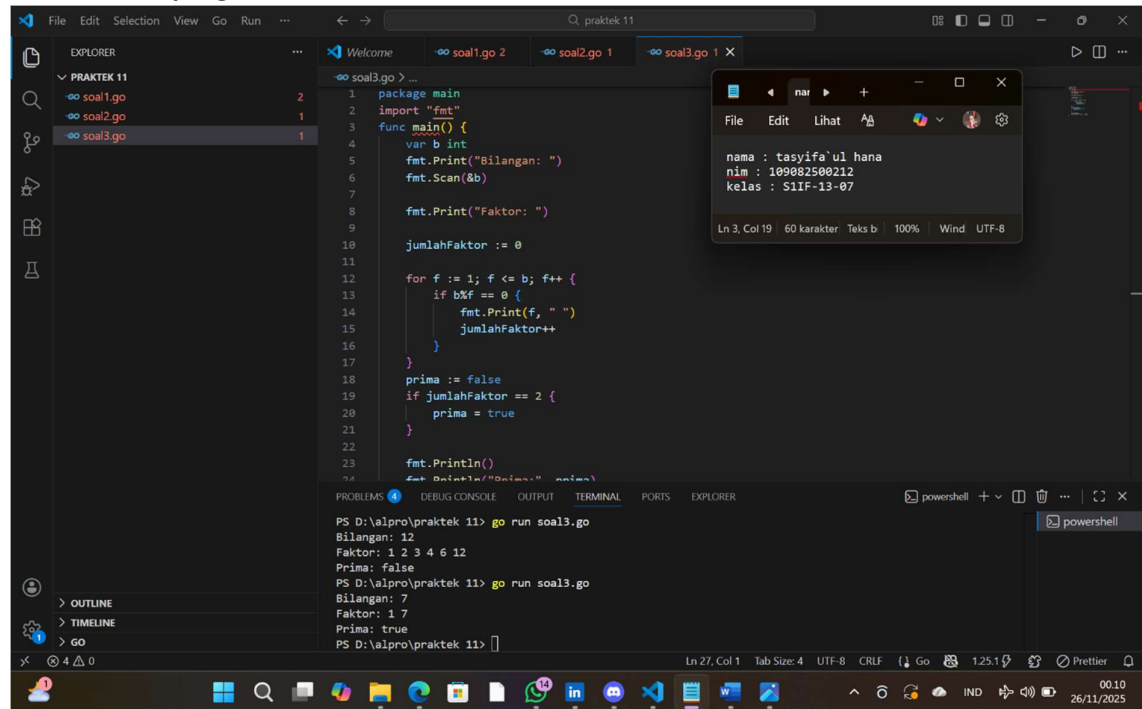
    fmt.Print("Faktor: ")

    jumlahFaktor := 0

    for f := 1; f <= b; f++ {
        if b%f == 0 {
            fmt.Print(f, " ")
            jumlahFaktor++
        }
    }
    prima := false
    if jumlahFaktor == 2 {
        prima = true
    }

    fmt.Println()
    fmt.Println("Prima:", prima)
}
```

## Screenshoot program



The screenshot shows a Go program in a text editor. The code defines a function `main` that takes an integer `b` as input. It prints the input, finds all factors of `b` by checking numbers from 1 to `b`, and prints the factors. It also checks if the number is prime by counting the factors; if there are exactly two factors, it is prime. Finally, it prints whether the number is prime or not.

```
1 package main
2 import "fmt"
3 func main() {
4     var b int
5     fmt.Print("Bilangan: ")
6     fmt.Scan(&b)
7
8     fmt.Print("Faktor: ")
9
10    jumlahFaktor := 0
11
12    for f := 1; f <= b; f++ {
13        if b%f == 0 {
14            fmt.Print(f, " ")
15            jumlahFaktor++
16        }
17    }
18    prima := false
19    if jumlahFaktor == 2 {
20        prima = true
21    }
22
23    fmt.Println()
24    fmt.Println("Bilangan " + strconv.Itoa(b) + " adalah prima: ")
25}
```

The terminal output shows the program being run twice. In the first run, the input is 12, and the output shows factors 1 2 3 4 6 12, and it is not prime. In the second run, the input is 7, and the output shows factors 1 7, and it is prime.

```
PS D:\alpro\praktek 11> go run soal3.go
Bilangan: 12
Faktor: 1 2 3 4 6 12
Prima: false
PS D:\alpro\praktek 11> go run soal3.go
Bilangan: 7
Faktor: 1 7
Prima: true
PS D:\alpro\praktek 11>
```

## Deskripsi program

Program diatas meminta pengguna untuk memasukkan sebuah bilangan, lalu mencari dan menampilkan semua faktor dari bilangan tersebut dengan cara mengecek satu per satu angka dari 1 hingga bilangan itu sendiri. Setiap angka yang dapat membagi bilangan dengan sisa nol akan dicetak sebagai faktor, sekaligus dihitung jumlah faktornya. Setelah perulangan selesai, program menentukan apakah bilangan tersebut merupakan bilangan prima, yaitu jika jumlah faktornya tepat dua. Terakhir, program menampilkan hasil apakah bilangan yang dimasukkan termasuk bilangan prima atau bukan.