

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 10**

**ELSE IF**



**Disusun oleh:**

**NABIL NAILUR RIDHO**

**109082530008**

**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 a bool

    fmt.Scan(&usia)

    fmt.Scan(&a)

    if usia >= 17 && a == true{

        fmt.Print("bisa membuat ktp")

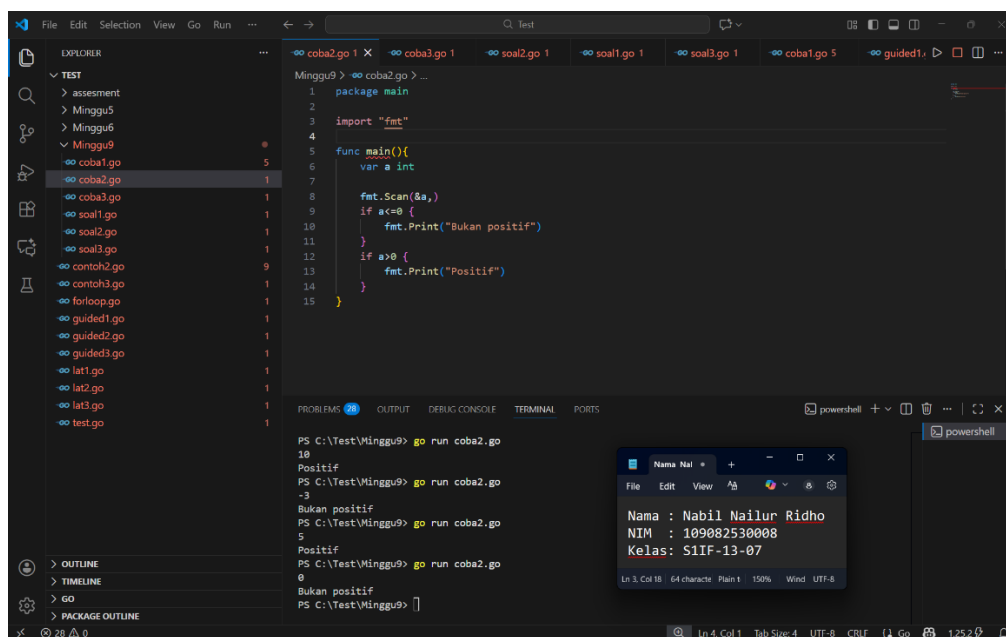
    } else {

        fmt.Print("belum bisa memmbuat ktp")

    }

}
```

### Screenshoot program



### Deskripsi program

Program Go ini untuk memilih kondisi untuk memvalidasi persyaratan pembuatan Kartu Tanda Penduduk (KTP). Sistem menerima dua parameter masukan, yakni usia pengguna dan status kelengkapan berkas (dalam bentuk *boolean*). Menggunakan operator logika AND (&&), program menetapkan aturan ketat bahwa izin pembuatan KTP hanya diberikan jika usia minimal 17 tahun dan status boolean bernilai true secara bersamaan. Jika salah satu syarat saja tidak terpenuhi, maka output akan menyatakan bahwa pengguna belum layak membuat KTP.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main(){

    var huruf rune

    fmt.Print("Masukkan Huruf: ")

    fmt.Scanf("%c", &huruf)


    if huruf == 'A' || huruf == 'I' || huruf == 'U' ||
huruf == 'E' || huruf == 'O' || huruf == 'a' || huruf ==
'i' || huruf == 'u' || huruf == 'e' || huruf == 'o' {

        fmt.Print("vokal")

    } else if (huruf >= 'a' && huruf <= 'z') || (huruf
>= 'A' && huruf <= 'Z') {

        fmt.Print("konsonan")

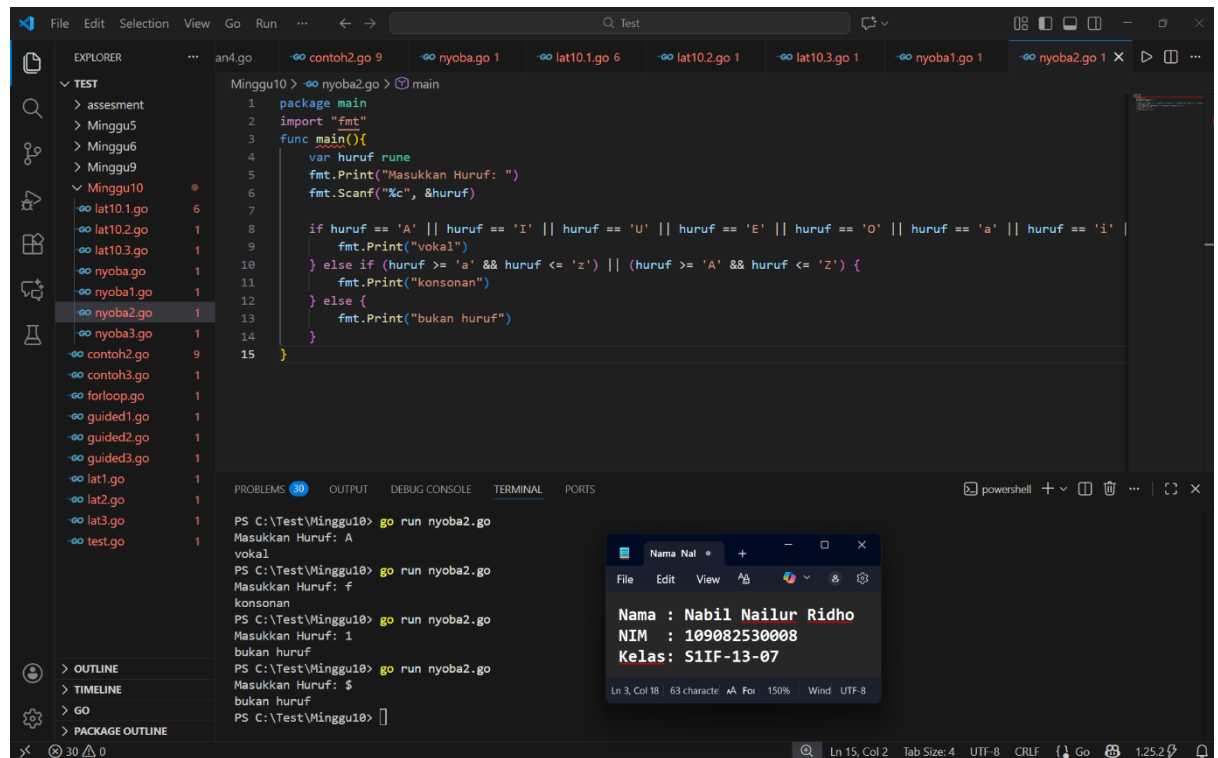
    } else {

        fmt.Print("bukan huruf")

    }

}
```

## Screenshot program



### Deskripsi program

Program Go ini buat mengidentifikasi dan mengelompokkan karakter input ke dalam kategori vokal, konsonan, atau karakter non-alfabet. Pengecekan dimulai dengan membandingkan input terhadap daftar huruf vokal (baik huruf besar maupun kecil) menggunakan operator OR. Apabila input bukan vokal, program melanjutkannya dengan memeriksa apakah karakter tersebut berada dalam rentang huruf abjad standar (a-z atau A-Z) untuk diklasifikasikan sebagai konsonan. Karakter di luar kedua definisi tersebut, seperti angka atau tanda baca, akan ditandai sebagai "bukan huruf".

### 3. Guided 3

## Source Code

```
package main

import "fmt"

func main(){

    var bilangan, d1, d2, d3, d4 int
```

```
var urutan string

fmt.Print("Bilangan: ")

fmt.Scan(&bilangan)

d1 = bilangan / 1000

d2 = (bilangan % 1000) / 100

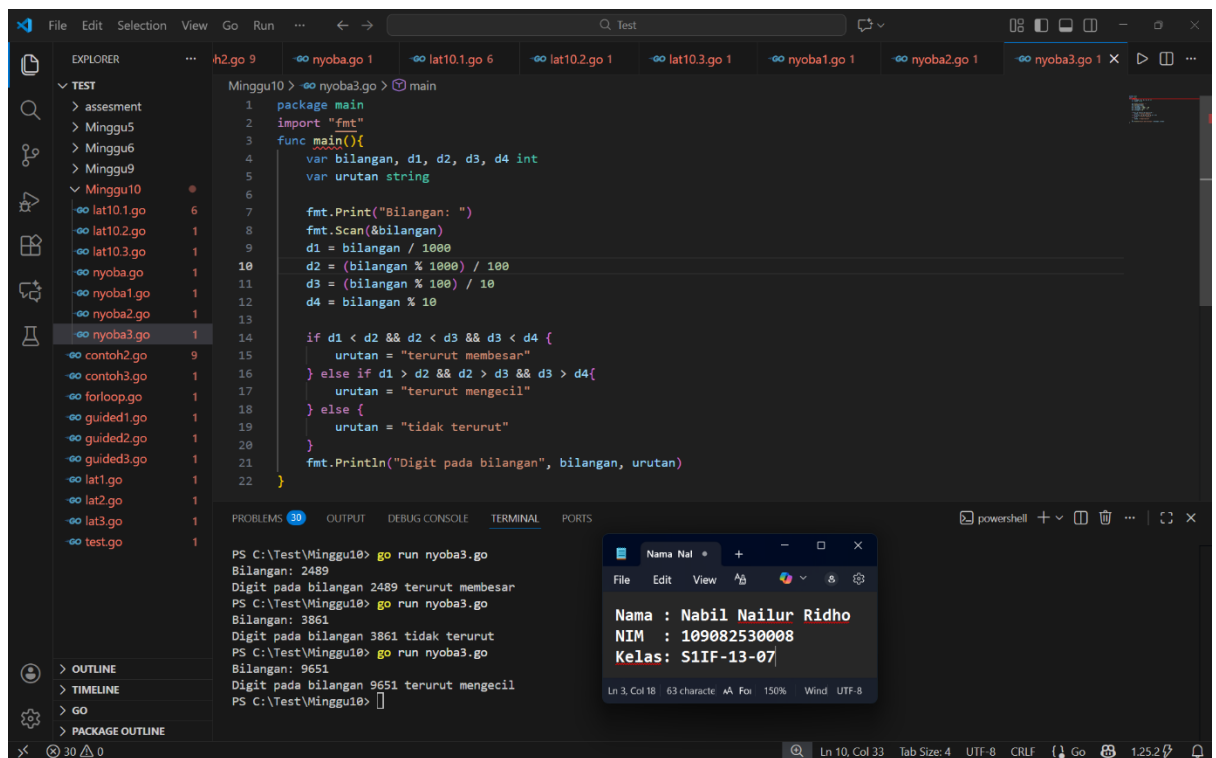
d3 = (bilangan % 100) / 10

d4 = bilangan % 10

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

fmt.Println("Digit pada bilangan", bilangan,
urutan)
}
```

## Screenshoot program



```
1 package main
2 import "fmt"
3 func main(){
4     var bilangan, d1, d2, d3, d4 int
5     var urutan string
6
7     fmt.Print("Bilangan: ")
8     fmt.Scan(&bilangan)
9     d1 = bilangan / 1000
10    d2 = (bilangan % 1000) / 100
11    d3 = (bilangan % 100) / 10
12    d4 = bilangan % 10
13
14    if d1 < d2 && d2 < d3 && d3 < d4 {
15        urutan = "terurut membesar"
16    } else if d1 > d2 && d2 > d3 && d3 > d4 {
17        urutan = "terurut mengecil"
18    } else {
19        urutan = "tidak terurut"
20    }
21    fmt.Println("Digit pada bilangan", bilangan, urutan)
22 }
```

```
PS C:\Test\Minggu10> go run nyoba3.go
Bilangan: 2489
Digit pada bilangan 2489 terurut membesar
PS C:\Test\Minggu10> go run nyoba3.go
Bilangan: 3861
Digit pada bilangan 3861 tidak terurut
PS C:\Test\Minggu10> go run nyoba3.go
Bilangan: 9651
Digit pada bilangan 9651 terurut mengecil
PS C:\Test\Minggu10>
```

Nama : Nabil Nailur Ridho  
NIM : 109082530008  
Kelas: S1IF-13-07

## Deskripsi program

Program Go ini buat membedah struktur sebuah bilangan bulat empat digit untuk menentukan pola urutan angkanya. Proses dimulai dengan teknik ekstraksi digit, di mana bilangan utama dipecah menjadi satuan-satuan terpisah (d1 hingga d4) melalui kombinasi operasi pembagian dan modulus. Setelah setiap digit didapatkan, program mengevaluasi hubungan antar digit tersebut secara sekuensial. Hasil evaluasi akan menyimpulkan apakah digit-digit tersebut tersusun dalam format yang konsisten membesar, konsisten mengecil, atau tidak memiliki keteraturan pola sama sekali.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {

    var parsel, kilogram, gram, harga, biaya, total int

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

    fmt.Scan(&parsel)

    kilogram = parsel / 1000

    gram = parsel % 1000

    harga = kilogram * 10000

    if parsel > 10000 {

        biaya = 0

        total = harga

    } else if gram >= 500 {

        biaya = gram * 5

        total = harga + biaya

    } else if gram < 500 {

        biaya = gram * 15

        total = harga + biaya

    }

}
```

```

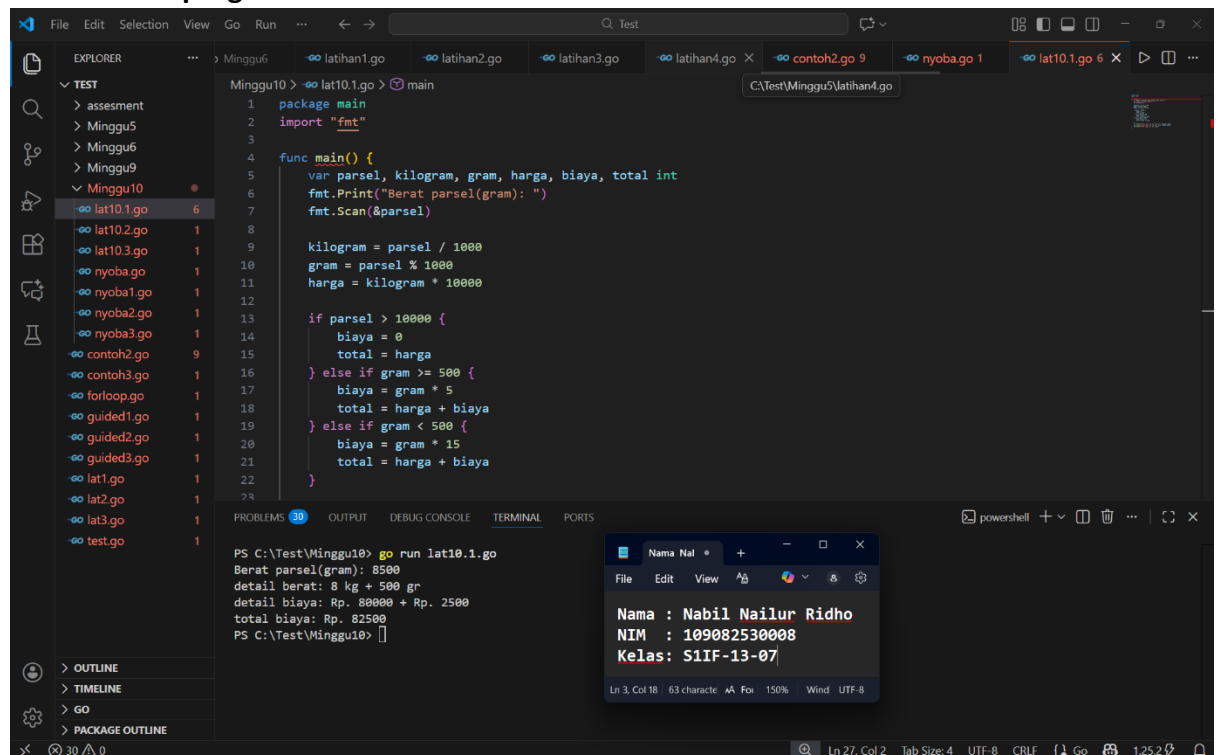
    fmt.Printf("detail berat: %d kg + %d gr\n", kilogram,
gram)

    fmt.Printf("detail biaya: Rp. %d + Rp. %d\n", harga,
biaya)

    fmt.Printf("total biaya: Rp. %d", total)
}

```

## Screenshoot program



## Deskripsi program

Program ini berfungsi untuk menghitung biaya pengiriman parcel berdasarkan berat total dalam gram. Pertama, program menerima input berat parcel lalu mengubahnya menjadi satuan kilogram dan sisa gram. Selanjutnya, harga dasar dihitung dari kilogram dikali 10.000. Setelah itu, program menentukan biaya tambahan berdasarkan jumlah gram: jika berat lebih dari 10.000 gram maka tidak ada biaya tambahan, jika  $\text{gram} \geq 500$  maka biaya adalah  $\text{gram} \times 5$ , dan jika  $\text{gram} < 500$  biaya adalah  $\text{gram} \times 15$ . Program kemudian menampilkan rincian berat, perhitungan biaya tambahan, dan total biaya pengiriman.

## 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"

    }

}
```

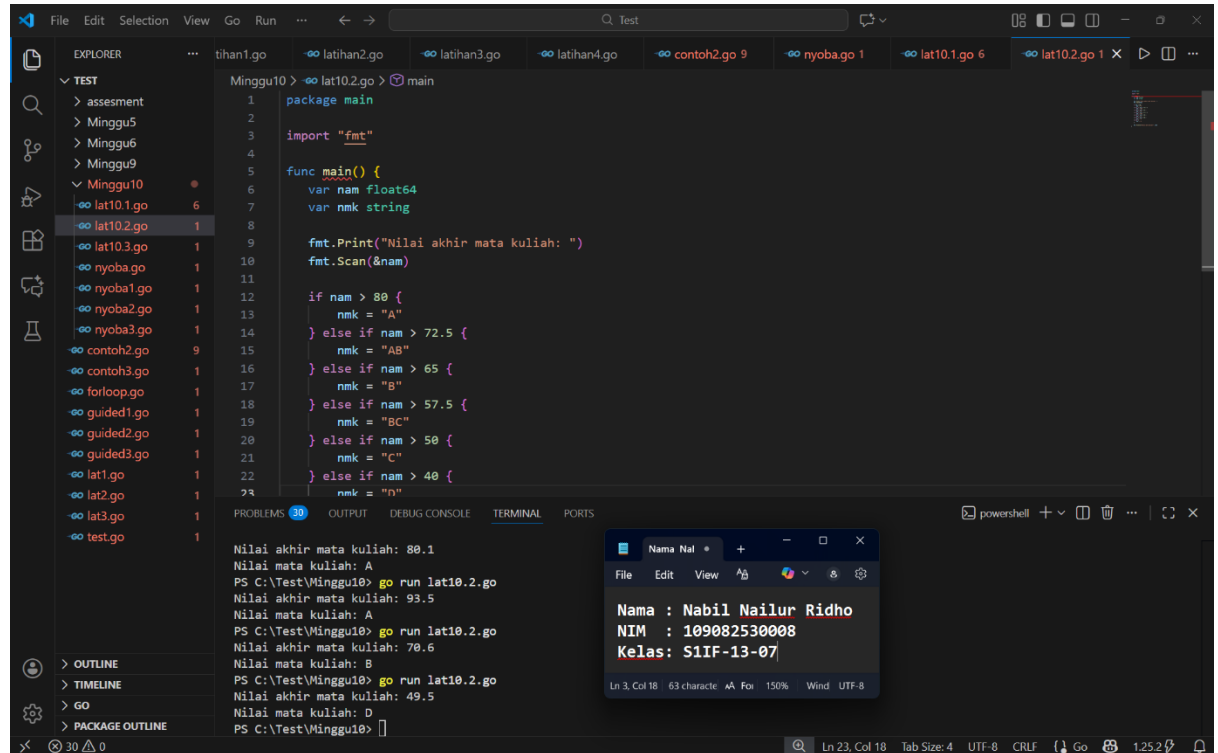
```

fmt.Println("Nilai mata kuliah:", nmk)

}

```

## Screenshoot program



## Deskripsi program

Program ini buat mengonversi nilai akhir mata kuliah menjadi nilai huruf berdasarkan rentang tertentu. Pengguna memasukkan nilai numerik, kemudian program membandingkannya melalui serangkaian kondisi untuk menentukan nilai huruf: nilai di atas 80 menjadi A, di atas 72.5 menjadi AB, di atas 65 menjadi B, di atas 57.5 menjadi BC, di atas 50 menjadi C, di atas 40 menjadi D, dan sisanya E. Program ini memanfaatkan struktur *if-else* bertingkat, sehingga setiap nilai numerik dapat diklasifikasikan secara tepat sesuai standar penilaian.

a. Jika nam diberikan adalah 80.1, apa keluaran dari program tersebut? Apakah eksekusi program tersebut sesuai spesifikasi soal?

Jawaban : maka keluarannya nmk : A, dan sesuai, tetapi jika belum di perbaiki maka keluarannya Adalah nam : D, dan tidak sesuai.

b. Apa saja kesalahan dari program tersebut? Mengapa demikian? Jelaskan alur program seharusnya!

Jawaban: Program memiliki kesalahan sintaks karena variabel nam (tipe *float*) keliru digunakan untuk menyimpan data huruf (*string*), padahal seharusnya menggunakan variabel nmk, dan terdapat kesalahan logika fatal akibat penggunaan if terpisah tanpa else yang menyebabkan nilai huruf terus tertimpa oleh kondisi di bawahnya (misalnya nilai 80.1 yang seharusnya "A" akan berubah menjadi "D"). Perbaikannya adalah mengganti target variabel menjadi nmk dan mengubah struktur kendali menjadi if-else if agar program berhenti mengecek setelah satu kondisi terpenuhi.

c. Perbaiki program tersebut! Ujilah dengan masukan: 93.5; 70.6; dan 49.5.

Seharusnya keluaran yang diperoleh adalah 'A', 'B', dan 'D'.

Jawaban: Sudah di perbaiki seperti pada gambar yang terdapat di atas soal dan deskripsi program berikut.

### 3. Tugas 3

#### Source code

```
package main

import "fmt"

func main() {
    var n, i, faktor int
    var prima bool
    fmt.Print("masukkan bilangan: ")
    fmt.Scan(&n)
    faktor = 0

    fmt.Print("faktor: ")
    for i = 1; i <= n; i++ {
        if n%i == 0 {
            fmt.Print(" ", i)
            faktor++
        }
    }

    fmt.Println()

    fmt.Print("Prima: ")
    if faktor == 2 {
        prima = true
    }
    fmt.Print(prima)
}
```

#### Screenshoot program

```
1 package main
2 import "fmt"
3
4 func main() {
5     var n, i, faktor int
6     var prima bool
7     fmt.Println("masukkan bilangan: ")
8     fmt.Scan(&n)
9     faktor = 0
10
11     fmt.Println("faktor: ")
12     for i = 1; i <= n; i++ {
13         if n%i == 0 {
14             fmt.Print(" ", i)
15             faktor++
16         }
17     }
18
19     fmt.Println()
20
21     fmt.Println("Prima: ")
22     if faktor == 2 {
23         prima = true
24     }
25 }
```

PS C:\Test\Minggu10> go run lat10.3.go  
masukkan bilangan: 12  
faktor: 1 2 3 4 6 12  
Prima: false  
PS C:\Test\Minggu10> go run lat10.3.go  
masukkan bilangan: 7  
faktor: 1 7  
Prima: true  
PS C:\Test\Minggu10>

Nama : Nabil Nailur Ridho  
NIM : 109082530008  
Kelas: S1IF-13-07

## Deskripsi program

Program ini buat menentukan apakah sebuah bilangan merupakan bilangan prima serta menampilkan semua faktor dari bilangan tersebut. Pengguna memasukkan bilangan bulat, lalu program menggunakan perulangan dari 1 hingga n untuk mencari semua nilai yang dapat membagi n (faktor). Setiap faktor ditampilkan dan dihitung jumlahnya. Setelah perulangan selesai, jika jumlah faktor tepat dua (yaitu 1 dan dirinya sendiri), maka bilangan tersebut dinyatakan prima; jika tidak, maka bukan bilangan prima. Program ini menggabungkan konsep perulangan, pengecekan modulus, dan logika sederhana untuk mengidentifikasi bilangan prima.