

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL PRAKTIKUM 10
ELSE-IF**



Disusun oleh:

Hanifan Bintang Wiraaji

109082500007

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 n int

    var kk bool

    fmt.Scan(&n)

    fmt.Scan(&kk)

    if n >= 17 && kk {

        fmt.Print("Bisa membuat ktp")

    } else {

        fmt.Print("Belum bisa membuat")

    }

}
```

Screenshoot program

The screenshot shows a VS Code editor with a Go file named `main.go`. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     var kk bool
8     fmt.Scan(&n)
9     fmt.Scan(&kk)
10    if n >= 17 && kk {
11        fmt.Print("Bisa membuat ktp")
12    } else {
13        fmt.Print("Belum bisa membuat")
14    }
15 }
16
```

The terminal output shows the execution of the program:

```
ed 1\main.go"
17
true
Bisa membuat ktp
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go
ed 1\main.go"
false
Belum bisa membuat
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go
ed 1\main.go"
15
true
Belum bisa membuat
```

A small window in the foreground shows the output of the program:

```
109082500007
Kelas 7
Hanifan Bintang Wiraaji
```

Deskripsi program

Program ini untuk memeriksa apakah seseorang bisa membuat ktp atau tidak. User akan diminta untuk input umur dan Boolean (true / false) mengenai apakah dia punya kk (kartu keluarga) atau tidak. Kemudian dalam program akan masuk ke pengkodisian if dimana jika user berumur lebih dari atau sama dengan 17 dan memiliki kk yang berarti booleannya True maka user bisa membuat ktp. Selain itu, jika salah satu atau kedua kondisi tidak terpenuhi maka user belum bisa membuat ktp.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var c rune

    fmt.Sprintf("%c", &c)
```

```

if c == 'a' || c == 'i' || c == 'u' || c == 'e' || c == 'o' ||

c == 'A' || c == 'I' || c == 'U' || c == 'E' || c == 'O' {

    fmt.Println("vokal")

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

    fmt.Println("konsonan")

} else {

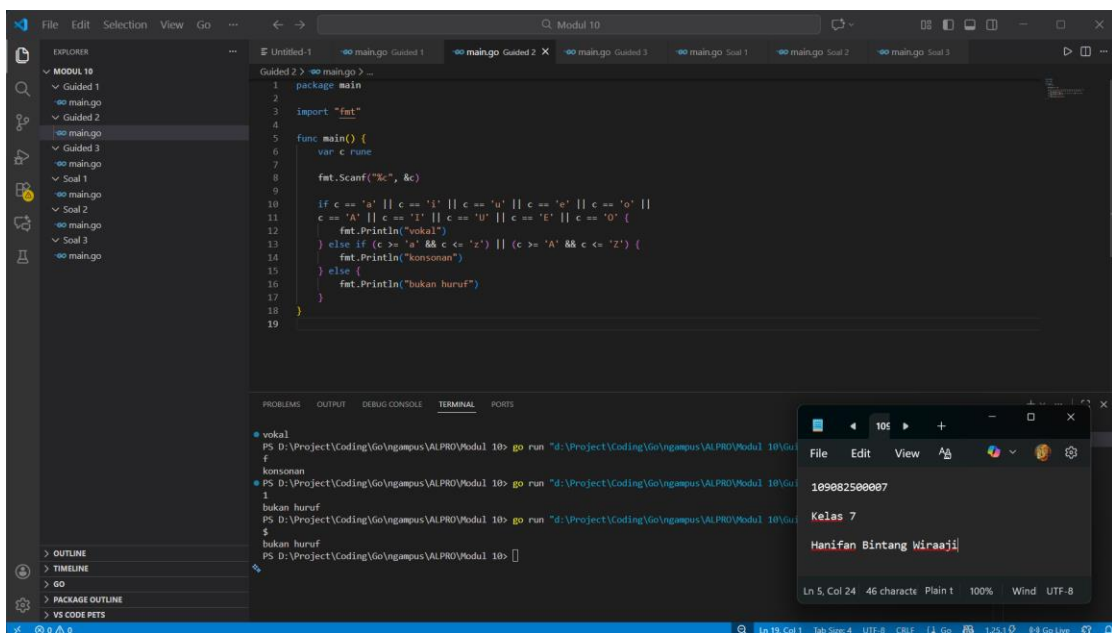
    fmt.Println("bukan huruf")

}

}

```

Screenshoot program



Deskripsi program

Program ini untuk memeriksa apakah suatu huruf adalah vokal, konsonan atau malah bukan huruf. User akan diminta input 1 huruf bebas. Input ini akan dikategorikan sebagai variable rune karena hanya sesimbol dan untuk memudahkan. Kemudian akan masuk ke pengkondisian if yang pertama; akan diperiksa apakah huruf itu masuk ke jenis vokal yang termasuk a, i, u, e, o. Jika hurufnya adalah vokal maka outputnya adalah vokal. Pengkondisian kedua jika bukan huruf vokal tapi masih huruf abjad maka dikategorikan sebagai konsonan, dan outputnya adalah konsonan. Yang terakhir jika

bukan huruf sama sekali maka akan dikategorikan sebagai bukan huruf dan outputnya adalah bukan huruf.

3. Guided 3

Source Code

```
package main

import (
    "fmt"
)

func main() {
    var n int

    fmt.Scan(&n)

    angka1 := n / 1000

    angka2 := n % 1000 / 100

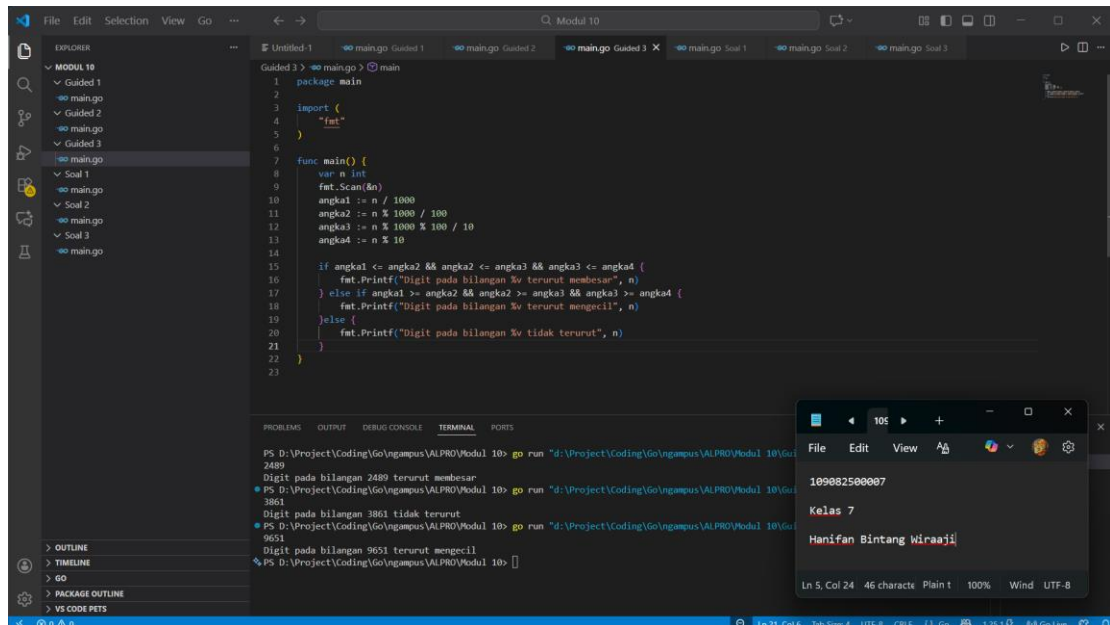
    angka3 := n % 1000 % 100 / 10

    angka4 := n % 10

    if angka1 <= angka2 && angka2 <= angka3 && angka3 <= angka4 {
        fmt.Printf("Digit pada bilangan %v terurut membesar", n)
    } else if angka1 >= angka2 && angka2 >= angka3 && angka3 >= angka4 {
        fmt.Printf("Digit pada bilangan %v terurut mengecil", n)
    } else {
        fmt.Printf("Digit pada bilangan %v tidak terurut", n)
    }
}
```

```
}
```

Screenshoot program



```
Guided 3 > main
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var n int
9     fmt.Scan(&n)
10    angka1 := n / 1000
11    angka2 := n % 1000 / 100
12    angka3 := n % 1000 % 100 / 10
13    angka4 := n % 10
14
15    if angka1 <= angka2 && angka2 <= angka3 && angka3 <= angka4 {
16        fmt.Printf("Digit pada bilangan %v terurut membesar", n)
17    } else if angka1 >= angka2 && angka2 >= angka3 && angka3 >= angka4 {
18        fmt.Printf("Digit pada bilangan %v terurut mengecil", n)
19    } else {
20        fmt.Printf("Digit pada bilangan %v tidak terurut", n)
21    }
22 }
23
```

```
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go\ngampus\ALPRO\Modul 10\Go
2489
Digit pada bilangan 2489 terurut membesar
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go\ngampus\ALPRO\Modul 10\Go
3861
Digit pada bilangan 3861 tidak terurut
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go\ngampus\ALPRO\Modul 10\Go
9651
Digit pada bilangan 9651 terurut mengecil
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10>
```

Deskripsi program

Program ini untuk memeriksa apakah suatu bilangan ribuan (4 angka) adalah bilangan berurut membesar, mengecil atau tidak berurut. User akan diminta memasukan bilangan ribuan (4 angka), kemudian program akan memisahkannya menjadi angka 1 – angka 4, rinciannya dengan modulo dan pembagian yang dilakukan di tiap angka seperti di kodingan yang tertera.

Kemudian program akan masuk ke pengkondisian if pertama yang dimana jika, angka 1 kurang dari sama dengan angka 2, dan angka 2 kurang dari sama dengan angka 3, dan angka 3 kurang dari sama dengan angka 4, maka akan masuk ke pengkondisian pertama dan memberi output bilangan ini berurut membesar. Kemudian pengkondisian kedua, hampir mirip seperti yang pengkondisian pertama hanya saja operator perbandingannya adalah lebih dari sama dengan, jika pengkondisiannya true maka akan masuk ke if kedua yang memberi output bilangan ini sebagai berurut mengecil. Jika bilangan tidak termasuk ke kedua pengkondisian diatas maka akan masuk else yang memberi output bilangan ini tidak berurut.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var b int

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

    fmt.Scan(&b)

    kgberat := b / 1000

    gberat := b % 1000

    fmt.Printf("Detail berat: %v kg + %v gr\n", kgberat, gberat)

    kgbiaya := kgberat * 10000

    gbiaya := gberat

    if gbiaya >= 500 {

        gbiaya = gbiaya * 5

    } else {

        gbiaya = gbiaya * 15

    }

    fmt.Printf("Detail biaya: Rp. %v + Rp. %v\n", kgbiaya, gbiaya)
```

```

var totalbiaya int

if kgberat > 10 {

    totalbiaya = kgbiaya

} else {

    totalbiaya = kgbiaya + gbiaya

}

fmt.Printf("Total biaya: Rp. %v", totalbiaya)

}

```

Screenshoot program

The screenshot shows a Go program in a VS Code editor. The program calculates the total cost based on weight and a base price. It uses a conditional statement to determine if the weight is greater than 10 kg. If so, it uses the base price; otherwise, it adds a surcharge. The program then prints the total cost.

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var b int
7
8     fmt.Print("Berat parsel(gram): ")
9     fmt.Scan(&b)
10
11     kgberat := b / 1000
12     gberat := b % 1000
13     fmt.Printf("Detail berat: %v kg + %v gr\n", kgberat, gberat)
14
15     kgbiaya := kgberat * 10000
16     gbiaya := gberat
17     if gbiaya >= 500 {
18         gbiaya = gbiaya * 5
19     } else {
20         gbiaya = gbiaya * 15
21     }
22     fmt.Printf("Detail biaya: Rp. %v + Rp. %v\n", kgbiaya, gbiaya)
23
24     var totalbiaya int
25     if b > 10000 {
26         totalbiaya = kgbiaya
27     } else {
28         totalbiaya = kgbiaya + gbiaya
29     }
30     fmt.Printf("Total biaya: Rp. %v", totalbiaya)
31 }

```

The terminal output shows the program's execution with the following values:

```

PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go\ngampus\ALPRO\Modul 10\Soal 1.go"
Berat parsel(gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10>

```

A small window in the foreground displays the output of the program, showing the total cost and the name of the user.

```

109082500007
Kelas 7
Hanifan Bintang Wiraaji

```



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var b int
7
8     fmt.Print("Berat parcel(gram): ")
9     fmt.Scan(&b)
10
11     kgberat := b / 1000
12     gberat := b % 1000
13     fmt.Printf("Detail berat: %v kg + %v gr\n", kgberat, gberat)
14
15     kgbiaya := kgberat * 10000
16     gbiaya := gberat
17     if gbiaya >= 500 {
18         gbiaya = gbiaya * 5
19     } else {
20         gbiaya = gbiaya * 15
21     }
22     fmt.Printf("Detail biaya: Rp. %v + Rp. %v\n", kgbiaya, gbiaya)
23
24     var totalbiaya int
25     if b > 10000 {
26         totalbiaya = kgbiaya
27     } else {
28         totalbiaya = kgbiaya + gbiaya
29     }
30     fmt.Printf("Total biaya: Rp. %v", totalbiaya)
31 }
```

PS D:\Project\Coding\Go\ngampus\VALPRO\Modul 10> go run "d:\Project\Coding\Go\ngampus\VALPRO\Modul 10\Soal 1\main.go"

Berat parcel(gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750

PS D:\Project\Coding\Go\ngampus\VALPRO\Modul 10>

Deskripsi program

Program ini untuk menghitung rincian berat dan biaya ongkos sebuah parcel. Perhitungan biayanya adalah sebagai berikut;

1. Tiap 1 kg seharga Rp. 10.000
2. Tiap lebihan yang lebih dari sama dengan 500 gram, akan di hargai Rp. 5 per gram nya
3. Tiap lebihan yang kurang dari 500 gram, akan di hargai Rp. 15 per gram nya
4. Tiap parcel yang beratnya lebih dari 10 kg, user hanya perlu membayar tiap kg nya tidak perlu membayar lebihannya yg kurang dari 1 kg

User akan diminta input berat parcel dalam gram. Kemudian program akan mengkonversi berat parcel menjadi rincian berupa berat dalam kg dan gram sisanya. Kg didapat dari membagi berat parcel dengan 1000, dan gram sisanya didapat dari

modulo berat parcel dengan 1000. Kemudian rincian beratnya ini akan di tampilkan sebagai output dengan keterangan berat dalam (angka) kg + (angka) gram.

Lalu akan diperiksa biaya tiap beratnya. Harga Kg akan dihitung dengan mengkalikan berat kg dengan 10.000. Harga dalam gram akan di hitung dengan pengkondisian. Pertama jika berat lebih dari sama dengan 500, harganya akan dihitung dengan mengkalikan berat gram dengan 5. Kedua jika beratnya kurang dari 500, maka harganya akan dihitung dengan mengkalikan berat gram dengan 15. Kemudian hasil rincian biaya tiap berat akan di tampilkan sebagai output dengan keterangan detail biaya dalam harga kg + harga gram.

Terakhir total biaya, disini akan ada 2 pengkondisian. Pertama jika beratnya lebih dari 10 kg atau 10000 gram maka lebihan yang kurang dari 1 kg akan di gratiskan, yang berarti total biaya nya adalah harga kg saja. Kemudian pengkondisian keduanya adalah else atau jika beratnya kurang dari 10 kg atau 10000 gram harga akan di total antara harga kg dan gram. Kemudian total biaya tadi akan di tampilkan sebagai output terakhir.

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 {

        nam = "A"

    }

    if nam > 72.5 {

        nam = "AB"

    }

}
```

```
if nam > 65 {  
  
    nam = "B"  
  
}  
  
if nam > 57.5 {  
  
    nam = "BC"  
  
}  
  
if nam > 50 {  
  
    nam = "C"  
  
}  
  
if nam > 40 {  
  
    nam = "D"  
  
} else if nam <= 40 {  
  
    nam = "E"  
  
}  
  
fmt.Println("Nilai mata kuliah: ", nmk)  
  
}
```

The fixed 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 if nam <= 40 {

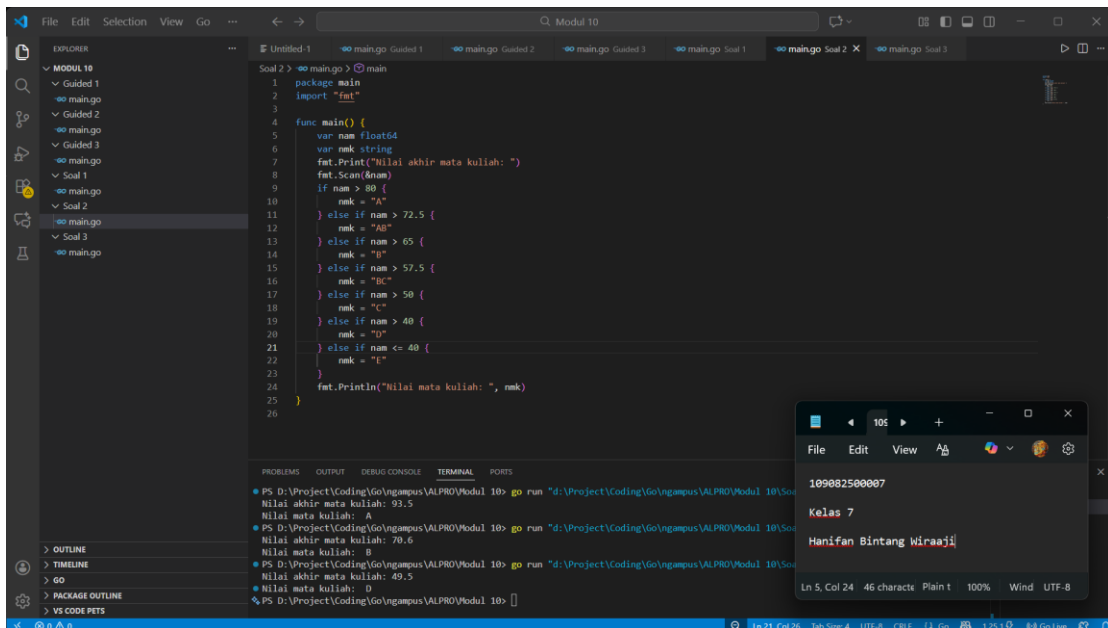
    nmk = "E"

}

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

}
```

Screenshoot program



Deskripsi program

Program ini untuk melihat nilai nmk dari nam yang di berikan. User akan diminta menginput nilai nam nya, kemudian program akan masuk pengkondisian untuk melihat rentang skor nmknnya dari nilai nam, jika lebih 80 akan A, lebih dari 72.5 akan AB, dan seterusnya. Kemudian hasil nmk nya ditampilkan.

Pertanyaan mengenai program

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

Jawab: Isi kodingannya eror semua, tapi kalo di benerin dikit tanpa merubah if nya, Hasilnya D, tidak sesuai dengan spesifikasi soal

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

Jawab: Kesalahannya cukup banyak;

- Tanda kutipnya tidak sesuai dengan standar vs code
- 5 if berjejer menurun yang tidak berguna karena yg terpakai ujung2nya yang paling bawah, seharusnya if ditulis dengan format if, kemudian else if, dan diakhiri else
- Variable yang diubah oleh if salah, seharusnya nmk karena dia adalah variable string, dan juga karena nmk adalah rentang nilai A, B dll, jadi penggunaan nam pada source code salah

Penjelasan alur program, User diminta input nam yang adalah float64, setelah kode dibenarkan, nam akan diperiksa menurun secara terstruktur di if else, sehingga menghasilkan nmk yang sesuai dengan nam.

- Perbaiki program tersebut! Ujilah dengan masukan: 93.5; 70.6; dan 49.5. Seharusnya keluaran yang diperoleh adalah 'A', 'B', dan 'D'.

Jawab: kode sudah dibenarkan, hasilnya ada di screenshot program

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {

    var b, prima int

    fmt.Print("Bilangan: ")

    fmt.Scan(&b)

    if b > 1 {

        fmt.Print("Faktor: ")

        for i := 1; i <= b; i++ {

            faktor := b % i

            if faktor == 0 {

                fmt.Print(i, " ")

                prima += i

            }

        }

        fmt.Println("")

    }

    if b > 0 {

        bilPrima := false

        prima -= 1

    }

}
```

```

    if prima == b {

        bilPrima = true

    }

    fmt.Printf("Prima: %v", bilPrima)

}

}

```

Screenshoot program

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var b, prima int
7     fmt.Print("Bilangan: ")
8     fmt.Scan(&b)
9     if b > 1 {
10         fmt.Print("Faktor: ")
11         for i := 1; i <= b; i++ {
12             faktor := b % i
13             if faktor == 0 {
14                 fmt.Print(i, " ")
15                 prima += i
16             }
17         }
18         fmt.Println("")
19     }
20     if b > 0 {
21         bilPrima := false
22         prima -= 1
23         if prima == b {
24             bilPrima = true
25         }
26         fmt.Printf("Prima: %v", bilPrima)
27     }
28 }
29

```

```

PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go\ngampus\ALPRO\Modul 10\Soal 3.go"
Bilangan: 12
Faktor: 1 2 3 4 6 12
Prima: false
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10> go run "d:\Project\Coding\Go\ngampus\ALPRO\Modul 10\Soal 3.go"
Bilangan: 7
Faktor: 1 7
Prima: true
PS D:\Project\Coding\Go\ngampus\ALPRO\Modul 10>

```

Deskripsi program

Program ini untuk memeriksa apakah suatu bilangan adalah bilangan prima dan menyajikan faktornya. User diminta input angka, program akan memeriksa factor bilangan tersebut. Akan ada perulangan dengan i yang bernilai 1 dan terus bertambah sampai sama dengan angka yg diinput, di dalamnya akan menghitung modulo angka yang diinput dengan i , jika hasilnya sama dengan 0 maka di tampilkan sebagai factor. Ini akan terus berulang sampai semua factor tertulis, atau sampai i mencapai sejumlah bilangan yg diinput.

Kemudian untuk bilangan primanya, jadi dalam perulangan tadi tiap bilangan factor muncul akan di akumulasi dalam variable prima. Kemudian jika angka yang diinput adalah bilangan prima, contoh 7 adalah angka yang diinput user, maka variable prima akan berisi angka 8, jumlah akumulasi dari factor 7 yang hanya 1 dan 7. Kemudian variable prima akan dikurangi dengan 1, kemudian masuk pengkondisian if, jika

variable prima sama dengan angka yang diinput user maka akan ditampilkan keterangan true pada prima.