

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 2
REVIEW STRUKTUR KONTROL**



Disusun Oleh :

Natasya Intan Sukma Jiwanti / 2311102279

S1-IF-11-05

Dosen Pengampu :

Arif Amrulloh, S.Kom., M.Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Bahasa Golang atau biasa disebut dengan bahasa Go merupakan bahasa pemrograman yang dikembangkan di Google oleh Robert Griesemer, Rob Pike, dan Ken Thompson pada tahun 2007. Bahasa ini memiliki beberapa kelebihan diantaranya pengapliannya yang cukup mudah.

Bahasa Go memiliki struktur atau kerangka program utama, dimana wajib memiliki package main yang merupakan penanda file yang berisi program utama, dan func main() yang berisi kode utama dari suatu program Go. Bahasa ini memiliki beberapa tipe data, diantaranya:

1. Interger (Bilangan Bulat dapat berupa bilangan positif dan negatif)
2. Uint (Bilangan Bulat positif)
3. Float (Bilangan real, yang bisa diubah dalam bentuk decimal)
4. Bool atau Boolean (logika, hanya terdapat dua kondisi yakni true atau false)
5. Karakter (ASCII)
6. String (Kalimat)

Pada bahasa ini juga terdapat variable yang berfungsi untuk menyimpan nilai. Variabel dapat dinotasikan dengan atau tanpa tipe data dari nilai, lihat contoh dibawah:

- 1) Apabila menggunakan tipe data maka kita harus menggunakan perintah: `var nama_variabel tipe_data`
- 2) Apabila tidak menggunakan tipe data kita dapat langsung mengetikkan: `nama_variabel := nilai_variabel`. Maka variable akan langsung menyesuaikan tipe datanya sesuai data yang disimpan.

Pada praktikan modul 2 ini, juga akan dibahas mengenai array,(yakni penyimpanan data), slice (array tanpa inisialisasi index maupun element),

perulangan for dengan atau tanpa batasan, dan juga operator. Operator yang terdapat dalam bahasa Go diantaranya:

- 1) + , untuk penjumlahan dan konkatenasi 2 string
- 2) - , untuk operasi pengurangan
- 3) * , untuk operasi perkalian
- 4) / , untuk operasi pembagian
- 5) >> atau <<, operator perbandingan
- 6) && untuk dan
- 7) || untuk operator atau
- 8) ! untuk menunjukkan negasi
- 9) = atau ==, operator sama dengan
- 10) % , untuk menunjukkan hasil bagi

II. GUIDED

1. Guided 1

Soal Studi Case

Siswa kelas IPA di salah satu sekolah menengah atas di Indonesia sedang mengadakan praktikum kimia. Di setiap percobaan akan menggunakan 4 tabung reaksi, yang mana susunan warna cairan di setiap tabung akan menentukan hasil percobaan. Siswa diminta untuk mencatat hasil percobaan tersebut. Percobaan dikatakan berhasil apabila susunan warna zat cair pada gelas 1 hingga gelas 4 secara berurutan adalah **'merah', 'kuning', 'hijau', dan 'ungu'** selama 5 kali percobaan berulang.

Buatlah sebuah program yang menerima input berupa warna dari ke 4 gelas reaksi sebanyak 5 kali percobaan. Kemudian program akan menampilkan true apabila urutan warna sesuai dengan informasi yang diberikan pada paragraf sebelumnya, dan false untuk urutan warna yang lainnya.

Sourcecode

```
package main

import (
    "bufio"
    "fmt"
    "os"
    "strings"
)

func main() {
    //urutan warna yang benar
    correctOrder := []string{"merah", "kuning", "hijau",
    "ungu"}

    //membaca input untuk 5 percobaan
    reader := bufio.NewReader(os.Stdin)
    success := true

    for i := 1; i <= 5; i++ {
        fmt.Printf("Percobaan %d: ", i)
```

```

//membaca input dari pengguna
input, _ := reader.ReadString('\n')
input = strings.TrimSpace(input)

//memisahkan input berdasarkan spasi
colors := strings.Split(input, " ")

//mengecek apakah urutan warna sesuai
for j := 0; j < 4; j++ {
    if colors[j] != correctOrder[j] {
        success = false
        break
    }
}

//menampilkan hasil
if success {
    fmt.Println("BERHASIL: true")
} else {
    fmt.Println("BERHASIL: false")
}
}

```

Screenshoot Output

```

PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2b_no1.go"
Percobaan 1: merah kuning ungu hijau
Percobaan 2: merah kuning hijau ungu
Percobaan 3: merah kuning hijau ungu
Percobaan 4: merah kuning hijau ungu
Percobaan 5: merah kuning hijau ungu
BERHASIL: false
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2b_no1.go"
Percobaan 1: merah kuning hijau ungu
Percobaan 2: merah kuning hijau ungu
Percobaan 3: merah kuning hijau ungu
Percobaan 4: merah kuning hijau ungu
Percobaan 5: merah kuning hijau ungu
BERHASIL: true
PS D:\Praktikum Alpro\Praktikum 1> █

```

Deskripsi Program

Pada program diatas, kita menggunakan array untuk menyimpan urutan warna yang benar, yakni dengan urutan **merah, kuning, hijau, dan ungu**. Lalu program akan meminta inputan sebanyak 5 kali percobaan urutan dengan menggunakan perulangan for, dan perintah input. Untuk memisahkan dan memproses data yang diinputkan pertama tama program akan memisahkan setiap inputan warna dalam satu percobaan menggunakan spasi dengan perintah split. Setelah dipisah menggunakan perintah split maka urutannya akan dicek apakah sesuai dengan array atau tidak. Jika tidak maka variabel sukses bernilai false. Lalu setelah user menginputkan kelima hasil percobaan barulah akan muncul nilai sukses apakah true (apabila kelima hasil percobaan sesuai urutan array) atau false (apabila dari kelima hasil percobaan terdapat satu percobaan yang gagal atau tidak sesuai urutan).

2. Guided 2

Soal Studi Case

Diberikan sebuah nilai akhir mata kuliah (NAM) [0..100] dan standar penilaian nilai mata kuliah (NMK) sebagai berikut:

NAM	NMK
$NAM > 80$	A
$72.5 < NAM \leq 80$	AB

$65 < NAM \leq 72.5$	B
$57.5 < NAM \leq 65$	BC
$50 < NAM \leq 57.5$	C
$40 < NAM \leq 50$	D
$NAM \leq 40$	E

Buatlah program yang menerima inputan nilai dan mencetak output NMK nya sesuai dengan tabel diatas!

Sourcecode

```
package main

import "fmt"

func main(){
    var nam float32
    var nmk string

    // Meminta input nilai
    fmt.Print("Masukkan Nilai: ")
    fmt.Scan(&nam)

    //Logika penentuan nilai huruf berdasarkan nilai numerik
    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"
    }

    //menampilkan hasil
    fmt.Printf("Nilai Indeks untuk nilai %.2f adalah %s\n",
nam, nmk)
}
```

Screenshot Output

```
PROBLEMS 16 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 35
Nilai Indeks untuk nilai 35.00 adalah E
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 42
Nilai Indeks untuk nilai 42.00 adalah D
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 58
Nilai Indeks untuk nilai 58.00 adalah BC
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 52
Nilai Indeks untuk nilai 52.00 adalah C
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 65
Nilai Indeks untuk nilai 65.00 adalah BC
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 70
Nilai Indeks untuk nilai 70.00 adalah B
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 80
Nilai Indeks untuk nilai 80.00 adalah AB
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no2.go"
Masukkan Nilai: 90
Nilai Indeks untuk nilai 90.00 adalah A
PS D:\Praktikum Alpro\Praktikum 1> █
```

Deskripsi Program

Pada program diatas kita menggunakan percabangan if, else if, dan else untuk dapat mengelompokkan Nilai Akhir Mahasiswa bertipe data float32 yang diinputkan oleh user merupakan NMK A, B, C, D, atau E, dengan syarat percabangan:

- 1) NMK = A, apabila nilai NAM lebih dari 80
- 2) NMK = AB, apabila nilai NAM diatas 72.6 - 80
- 3) NMK = B, apabila nilai NAM diatas 65 – 72.5
- 4) NMK = BC, apabila nilai NAM diatas 57.5 hingga 65
- 5) NMK = C, apabila nilai NAM diatas 50 hingga 57.5
- 6) NMK = D, apabila nilai NAM diatas 40 hingga 50
- 7) NMK = E, apabila nilai NAM kurang dari 40 atau sama dengan 40

3. Guided 3

Soal Studi Case

Buatlah program yang digunakan untuk menghitung hasil penjumlahan 5 bilangan bulat. Masukan terdiri dari lima bilangan bulat a, b, c, d, dan e. Keluaran berupa bilangan hasil penjumlahan lima bilangan bulat a,b,c,d, dan e.

Sourcecode

```
package main

import (
    "fmt"
)

func main(){
    var a,b,c,d,e int
    var hasil int
    fmt.Print("Inputkan angka: ")
    fmt.Scanln(&a, &b, &c, &d, &e)

    hasil = a+b+c+d+e
    fmt.Println("Hasil Penjumlahan ",a,b,c,d,e, "adalah",hasil)
}
```

Screenshoot Output

```
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\guided3.go"
Inputkan angka: 10 11 12 89 88 20
Hasil Penjumlahan 10 11 12 89 88 adalah 210
PS D:\Praktikum Alpro\Praktikum 1> █
```

Deskripsi Program

Program diatas menggunakan perintah scan untuk menerima inputan bertipe data interger atau bilangan bulat dari user. Untuk menjumlahkannya kita hanya perlu menggunakan + dan mencetak nilainya menggunakann println, yang juga akan membuat baris baru.

I. UNGUIDED

1. Unguided 1

Soal Studi Case

Suatu pita (string) berisi kumpulan nama – nama bunga yang dipisahkan oleh spasi dan '-', contoh pita diilustrasikan seperti berikut ini.

Pita: mawar – melati – tulip – teratai – kamboja – anggrek

Buatlah sebuah program yang menerima input sebuah bilangan bulat positif (dan tidak nol) N, kemudian program akan meminta input berupa nama bunga secara berulang sebanyak N kali dan nama tersebut disimpan ke dalam pita. Lalu modifikasilah program tersebut, dimana proses input akan berhenti apabila user mengetikkan 'SELESAI'. Kemudian tampilkan isi pita beserta banyaknya bunga yang ada di dalam pita.

Sourcecode

```
package main

import (
    "fmt"
)

func main(){
    //urutan pita
    Urutan := [] string{"Mawar", "Melati", "Tulip", "Teratai",
    "Kamboja", "Anggrek"}
    var N int
    fmt.Print("Masukkan bilangan bulat: ")
    fmt.Scan (&N)
    fmt.Println(" ")
    if N == 0{
        fmt.Println("Program tidak menerima input 0!")
    }else{
        for a:= 1; a <= N; a++){
            sisa := a%6
            if sisa == 1{
                fmt.Println ("Bunga ", a ,": " + Urutan[0])
            }else if sisa == 2{
                fmt.Println ("Bunga ", a ,": " + Urutan[1])
            }else if sisa == 3{
```

```

        fmt.Println ("Bunga ", a ,": " + Urutan[2])
    }else if sisa == 4{
        fmt.Println ("Bunga ", a ,": " + Urutan[3])
    }else if sisa == 5{
        fmt.Println ("Bunga ", a ,": " + Urutan[4])
    }else if sisa == 0{
        fmt.Println ("Bunga ", a ,": " + Urutan[5])
    }
}
fmt.Print("Pita : ")
for a:= 1; a <= N; a++){
    sisa := a%6
    if sisa == 1{
        fmt.Print (Urutan[0] + " - ")
    }else if sisa == 2{
        fmt.Print (Urutan[1] + " - ")
    }else if sisa == 3{
        fmt.Print (Urutan[2] + " - ")
    }else if sisa == 4{
        fmt.Print (Urutan[3] + " - ")
    }else if sisa == 5{
        fmt.Print (Urutan[4] + " - ")
    }else if sisa == 0{
        fmt.Print (Urutan[5] + " - ")
    }
}
}
fmt.Println(" ")

//Modifikasi program
fmt.Println("\nMODIFIKASI PROGRAM")
for a:= 1; a>0; a++){
    var Bunga string
    fmt.Print("Bunga ", a, ": ")
    fmt.Scan(&Bunga)
    if Bunga == "SELESAI" || Bunga == "Selesai"{
        break
    }
}
}

```

Screenshoot Output

```
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2b_no2.go"
Masukkan bilangan bulat: 10

Bunga 1 : Mawar
Bunga 2 : Melati
Bunga 3 : Tulip
Bunga 4 : Teratai
Bunga 5 : Kamboja
Bunga 6 : Anggrek
Bunga 7 : Mawar
Bunga 8 : Melati
Bunga 9 : Tulip
Bunga 10 : Teratai
Pita : Mawar - Melati - Tulip - Teratai - Kamboja - Anggrek - Mawar - Melati - Tulip - Teratai -
MODIFIKASI PROGRAM
Bunga 1: Mawar
Bunga 2: Melati
Bunga 3: SELESAI
PS D:\Praktikum Alpro\Praktikum 1> █
```

Deskripsi Program

Pada program diatas, yang akan pertama kali kita lakukan pada program adalah membuat array yang berisikan urutan pita dengan nama nama bunga. Setelah itu kita akan meminta inputan dari user berapa banyak bagian pita yang akan mereka butuhkan. Untuk mencetak pita sesuai urutan dan jumlah yang diinputkan oleh user kita akan menggunakan perulangan for dan percabangan if untuk menyesuaikan urutannya. Kita akan memanfaatkan operator % untuk menentukan bunga mana yang keluar dengan syarat:

- 1) Bilangan yang diinputkan user memiliki % 1 apabila dibagi dengan 6 maka yang keluar adalah array pita indeks ke-0.
- 2) Bilangan yang diinputkan user memiliki % 2 apabila dibagi dengan 6 maka yang keluar adalah array pita indeks ke-1.
- 3) Bilangan yang diinputkan user memiliki % 3 apabila dibagi dengan 6 maka yang keluar adalah array pita indeks ke-2.
- 4) Bilangan yang diinputkan user memiliki % 4 apabila dibagi dengan 6 maka yang keluar adalah array pita indeks ke-3.
- 5) Bilangan yang diinputkan user memiliki % 5 apabila dibagi dengan 6 maka yang keluar adalah array pita indeks ke-4.
- 6) Bilangan yang diinputkan user memiliki % 0 apabila dibagi dengan 6 maka yang keluar adalah array pita indeks ke-5.

Untuk menghubungkan bunga 1 sama lain pada pita kita akan menambahkan "-" dengan menggunakan operator +.

Lalu pada modifikasi program kita akan menggunakan perulangan for tanpa batas, supaya program dapat meminta input dari user tak terbatas, kecuali jika user mengetikkan "SELESAI" atau "Selesai".

2. Unguided 2

Soal Studi Case

Setiap hari Pak Andi membawa banyak barang belanjaan dari pasar dengan mengendarai sepeda motor. Barang belanjaan tersebut dibawa dalam kantong terpal di kiri-kanan motor. Sepeda motor tidak akan oleng jika selisih berat barang di kedua kantong sisi tidak lebih dari 9 kg.

Buatlah program Pak Andi yang menerima input dua buah bilangan real positif yang menyatakan berat total masing-masing isi kantong terpal. Program akan terus meminta input bilangan tersebut hingga salah satu kantong terpal berisi 9kg atau lebih.

Lalu modifikasilah program tersebut, program akan menampilkan true jika selisih kedua isi kantong lebih dari atau sama dengan 9 kg. Program berhenti memproses apabila total berat isi kedua kantong melebihi 150kg atau salah satu kantong beratnya negatif.

Sourcecode

Belum Dimodifikasi

```
package main
import "fmt"

func main(){
    var bebankiri float32
    var bebankanan float32

    for{
        fmt.Print("Masukkan Beban: ")
        fmt.Scan(&bebankiri,&bebankanan)
        if bebankanan >= 9.0 || bebankiri >= 9.0{
            break
        }
    }
}
```

Sudah Dimodifikasi

```
package main
import "fmt"

func main(){
    var bebankiri float32
    var bebankanan float32

    //for{
        //fmt.Print("Masukkan Beban: ")
        //fmt.Scan(&bebankiri,&bebankanan)
        //if bebankanan >= 9.0 || bebankiri >= 9.0{
            //break
        //}
    //}

    //modifikasi
    for{
        fmt.Print("Masukkan Beban: ")
        fmt.Scan(&bebankiri,&bebankanan)
        var nilai bool = false
        total := bebankanan + bebankiri
        if (bebankanan - bebankiri) >= 9 || (bebankiri -
bebankanan) >= 9{
            nilai = true
        }
        if total > 150 || bebankanan <0 || bebankiri <0{
            break
        }
        fmt.Println("Sepeda Pak Andi Oleng : ", nilai)
    }
}
```

Screenshoot Output

Belum dimodifikasi

```
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2b_no3.go"
Masukkan Beban: 8 6
Masukkan Beban: 6 5
Masukkan Beban: 4 8
Masukkan Beban: 7 2
Masukkan Beban: 9 8.76
PS D:\Praktikum Alpro\Praktikum 1> █
```

Sudah dimodifikasi

```
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2b_no3.go"
Masukkan Beban: 56 54
Sepeda Pak Andi Olen : false
Masukkan Beban: 45 65
Sepeda Pak Andi Olen : true
Masukkan Beban: 12.6 17.6
Sepeda Pak Andi Olen : false
Masukkan Beban: 90 12
Sepeda Pak Andi Olen : true
Masukkan Beban: 76 46
Sepeda Pak Andi Olen : true
Masukkan Beban: 100 65
PS D:\Praktikum Alpro\Praktikum 1> 
```

Deskripsi Program

Program diatas merupakan program yang dibuat untuk menginputkan beban sebelah kiri maupun kanan motor pak andi yang menggunakan variabel bebankiri dan bebankanan yang menggunakan tipe data float 32 supaya bisa menerima inputan desimal dari user. Pada sebelum dimodifikasi, program akan terus menerima input dari user tanpa batas dengan syarat beban yang diinputkan oleh user nilainya tidak melebihi 9 baik itu untuk bebankiri maupun bebankanan.

Pada setelah dimodifikasi program akan terus menerima beban dari user tanpa batasan, dengan ketentuan:

- 1) Apabila selisih diantara kedua beban, yakni bebankiri dan bebankanan melebihi atau sama dengan 9kg, maka akan tercetak output pernyataan motor pak andi akan olen adalah benar. Jika tidak melebihi maka akan tercetak output pernyataan motor pak andi akan olen adalah salah.
- 2) Apabila total berat kedua beban sudah melebihi 150 atau salah satu atau kedua beban yang ada di kanan maupun di kiri bernilai minus atau negatif maka program akan berhenti memproses.

3. Unguided 3

Soal Studi Case

$$f(k) = \frac{(4k + 2)^2}{(4k + 1)(4k + 3)}$$

Buatlah sebuah program yang menerima input sebuah bilangan K, kemudian menghitung dan menampilkan nilai f(K) sesuai persamaan di atas.

Sourcecode

```
package main
import "fmt"

func main(){
    var K float32
    fmt.Print("Masukkan Nilai K: ")
    fmt.Scan(&K)
    a := (4*K + 2)
    atas := a * a
    bawah := (4*K+1)*(4*K + 3)
    var nilaif float32 = atas / bawah
    fmt.Print("Nilai F dari K: ", nilaif)
}
```

Screenshot Output

```
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2b_no4.go"
Masukkan Nilai K: 100
Nilai F dari K: 1.0000062
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2b_no4.go"
Masukkan Nilai K: 80
Nilai F dari K: 1.0000097
PS D:\Praktikum Alpro\Praktikum 1> █
```

Deskripsi Program

Program diatas merupakan program yang dibuat untuk menghitung nilai sebuah fungsi yang memiliki rumus $f(K) = \frac{(4K+2)^2}{(4K+1) * (4K+3)}$, dimana K merupakan sebuah bilangan bulat yang diterima program dari user. Program diatas menggunakan tipe data float32 karena hasil perhitungan yang terjadi pada program bernilai desimal. Pertama program akan menghitung proses a, yakni nilai K yang diinputkan oleh user akan dikalikan 4 kemudian ditambahkan 2. Setelah proses tersebut selesai, maka nilainya akan disimpan dalam variabel a sehingga dapat digunakan dalam perhitungan variabel atas yakni $a * a$. Kemudian saya membuat variabel

bawah untuk menyimpan proses perhitungan $(4K+1) * (4K+3)$. Setelah perhitungan variabel atas dan bawah selesai, maka kedua variabel tersebut akan dihitung kembali untuk mendapatkan nilai fungsi f, yakni dengan membagi variabel atas dengan bawah. Kemudian program akan mencetak output sesuai dengan nilai fungsi f.

4. Unguided 4

Soal Studi Case

PT POS membutuhkan aplikasi perhitungan biaya kirim berdasarkan berat parcel. Maka, buatlah program Biaya Pos untuk menghitung biaya pengiriman tersebut dengan ketentuan sebagai berikut:

Dari berat parcel (dalam gram), harus dihitung total berat dalam kg dan sisanya (dalam gram). Biaya jasa pengiriman adalah Rp 10.000,- per kg. Jika sisa berat tidak kurang dari 500 gram, maka tambahan biaya kirim hanya Rp. 5,- per gram saja. Tetapi jika kurang dari 500 gram, maka tambahan biaya akan dibebankan sebesar Rp 15,-per gram. Sisa berat (yang kurang dari 1 kg) digratiskan biayanya apabila total berat ternyata lebih dari 10kg.

Sourcecode

```
package main
import "fmt"

func main(){
    var berat, kilogram, gram, harga1, harga2, totalharga
    uint32
    fmt.Print("Berat Parsel(gram): ")
    fmt.Scan(&berat)
    gram = berat % 1000
    kilogram = (berat - gram) / 1000
    fmt.Println("Detail Berat: ", kilogram , "Kg + ", gram,
    "gr")
    if kilogram <= 10 && gram < 500 {
        harga1 = kilogram * 10000
        harga2 = gram * 15
        totalharga = harga1 + harga2
        fmt.Println("Detail Biaya: ", harga1, " + ", harga2)
```

```

        fmt.Println("Total Biaya: ", totalharga)
    }else if kilogram <= 10 && gram >= 500{
        harga1 = kilogram * 10000
        harga2 = gram * 5
        totalharga = harga1 + harga2
        fmt.Println("Detail Biaya: ", harga1, " + ", harga2)
        fmt.Println("Total Biaya: ", totalharga)
    }else if kilogram > 10{
        harga1 = (kilogram * 10000)
        harga2 = 0
        totalharga = harga1 + harga2
        fmt.Println("Detail Biaya: ", harga1, " + ", harga2)
        fmt.Println("Total Biaya: ", totalharga)
    }
}

```

Screenshoot Output

```

PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\tempCodeRunnerFile.go"
Berat Parsel(gram): 8050
Detail Berat: 8 Kg + 50 gr
Detail Biaya: 80000 + 750
Total Biaya: 80750
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\tempCodeRunnerFile.go"
Berat Parsel(gram): 7750
Detail Berat: 7 Kg + 750 gr
Detail Biaya: 70000 + 3750
Total Biaya: 73750
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\tempCodeRunnerFile.go"
Berat Parsel(gram): 11550
Detail Berat: 11 Kg + 550 gr
Detail Biaya: 110000 + 0
Total Biaya: 110000
PS D:\Praktikum Alpro\Praktikum 1> 

```

Deskripsi Program

Program diatas merupakan program yang digunakan untuk menghitung biaya pengantaran suatu parsel di kantor POS. Program ini menggunakan tipe data uint32 yakni tipe data bilangan bulat positif. Awalnya program akan meminta pengguna untuk menginputkan berat parsel dalam satuan gram. Setelah itu berat dalam parsel ini akan dikonversi menjadi kilogram dimana kita akan membagi berat parsel dalam gram dengan 1000, karena 1kg = 1000 gram. Untuk sisa bagi dari perhitungan tersebut akan tetap dalam satuan gram.

Lalu untuk menghitung total biaya yang perlu dibayarkan oleh pengguna, maka kita menggunakan percabangan if dengan ketentuan sebagai berikut:

- 1) Apabila berat total < 10 kg dan sisa gram < 500 gram maka harga berat parsel setiap 1 kg nya adalah Rp 10.000 dan setiap gram sisaannya adalah Rp 5.

- 2) Apabila berat total < 10 kg dan sisa gram > 500 gram maka harga berat parcel setiap 1 kg nya adalah Rp 10.000 dan setiap gram sisaannya adalah Rp 15.
- 3) Apabila berat total > 10 kg maka harga berat parcel setiap 1 kg nya adalah Rp 10.000 dan setiap gram sisaannya gratis atau Rp 0.

5. Unguided 6

Soal Studi Case

Sebuah bilangan bulat b memiliki faktor bilangan $f > 0$ jika f habis membagi b . Contoh: 2 merupakan faktor dari bilangan 6 karena 6 habis dibagi 2. Buatlah program yang menerima input sebuah bilangan bulat b dan $b > 1$. Program harus dapat mencari dan menampilkan semua faktor dari bilangan tersebut!

Modifikasi Program:

Bilangan bulat $b > 0$ merupakan bilangan prima p jika dan hanya jika memiliki persis dua faktor bilangan saja yaitu 1 dan dirinya sendiri. Lanjutkan program sebelumnya. Setelah menerima masukan sebuah bilangan bulat $b > 0$. Program tersebut mencari dan menampilkan semua faktor bilangan tersebut. Kemudian, program menentukan apakah b merupakan bilangan prima.

Sourcecode

```
package main

import "fmt"

func main(){
    var input, bagi, a uint16
    var jumlah int = 0
    var faktor [] uint16
    var prima bool = false
    fmt.Print("Masukkan Bilangan Bulat: ")
    fmt.Scan (&input)
    for a= 1; a <= input; a++){
        bagi = input % a
        if bagi == 0{
            faktor = append(faktor, a)
        }
    }
}
```

```

        jumlah++
    }
}
fmt.Println("Faktor dari bilangan ", input, " adalah ")
for _, f := range faktor {
    fmt.Print(f, " ")
}
fmt.Println()
if jumlah == 2{
    prima = true
}
fmt.Println("Prima: ", prima)
}

```

Screenshoot Output

```

PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no3.go"
Masukkan Bilangan Bulat: 9
Faktor dari bilangan 9 adalah
1 3 9
Prima: false
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no3.go"
Masukkan Bilangan Bulat: 12
Faktor dari bilangan 12 adalah
1 2 3 4 6 12
Prima: false
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no3.go"
Masukkan Bilangan Bulat: 5
Faktor dari bilangan 5 adalah
1 5
Prima: true
PS D:\Praktikum Alpro\Praktikum 1> go run "d:\Praktikum Alpro\Praktikum 1\Modul 2\2c_no3.go"
Masukkan Bilangan Bulat: 13
Faktor dari bilangan 13 adalah
1 13
Prima: true
PS D:\Praktikum Alpro\Praktikum 1>

```

Deskripsi Program

Program diatas merupakan program untuk menghitung faktor dari suatu bilangan, untuk itu digunakanlah tipe data uint16 yakni bilangan bulat positif. Pertama akan dilakukan inisialisasi variabel seperti input, bagi, dan slice faktor dengan tipe data uint16. Serta terdapat variabel jumlah yaang menggunakan tipe data interger dan logika prima yang aawalnya bernilai false.

Program akan meminta input bilangan bulat yang ingin dicari faktornya dari pengguna. Setelah itu untuk mencari faktor – faktor penyusun dari bilangan tersebut kita akan menggunakan perulangan untuk melakukan proses pembagian input dengan bilangaan bulat lainnya dengan batasan bilangan bulat pembagi <= bilangan input. Setelah itu untuk

menyimpan faktor, kita menggunakan percabangan if dengan syarat hasil bagi = 0, dan perintah append untuk memasukkan bilangan yang habis membagi input ke dalam slice bernama faktor.

Kemudian untuk mencetak apakah bilangan input tersebut merupakan bilangan prima atau bukan kita menggunakan percabangan if dengan syarat faktor bilangan tersebut harus sama dengan 2. Apabila iya maka akan tercetak output prima: true, apabila tidak maka akan tercetak output prima: false.