

**LAPORAN PRAKTIKUM  
PEMROGRAMAN BERORIENTASI OBJEK**

**MODUL II  
REVIEW STRUKTUR KONTROL**



Oleh:

**IKRAM IRIANSYAH**

2311102184

S1-IF-11-02

**S1 TEKNIK INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

**2024**

## I. DASAR TEORI

Review struktur kontrol dalam bahasa Go (Golang) berkaitan dengan berbagai elemen yang digunakan untuk mengontrol aliran eksekusi program. Struktur kontrol mengatur bagaimana blok kode tertentu dieksekusi berdasarkan kondisi atau perulangan. [1] Berikut beberapa dasar teori dari struktur kontrol dalam bahasa Go:

### 1. Pernyataan Kondisional (**if, else if, else**)

- Digunakan untuk mengeksekusi blok kode tertentu jika kondisi tertentu terpenuhi.
- Bentuk dasar.

```
if kondisi {  
    // blok kode dieksekusi jika kondisi true  
} else if kondisi_lain {  
    // blok kode dieksekusi jika kondisi_lain true  
} else {  
    // blok kode dieksekusi jika tidak ada kondisi yang terpenuhi  
}
```

- Go juga mendukung pendeklarasian variabel dalam pernyataan if :

```
if nilai := hitungNilai(); nilai > 10 {  
    fmt.Println("Nilai lebih besar dari 10")  
}
```

### 2. Pernyataan Switch (**Switch**)

- Alternatif dari beberapa pernyataan if else, mempermudah pembacaan kode.
- Tidak seperti bahasa lain, switch Go secara otomatis tidak membutuhkan break untuk menghentikan eksekusi.
- Bentuk dasar.

```
switch ekspresi {  
case nilai1:  
    // blok kode jika ekspresi == nilai1  
case nilai2:  
    // blok kode jika ekspresi == nilai2  
default:  
    // blok kode jika tidak ada nilai yang cocok  
}
```

- **Switch** juga bisa digunakan tanpa ekspresi untuk mengevaluasi kondisi :

```
switch {  
  case x > 10:  
    fmt.Println("x lebih besar dari 10")  
  case x < 5:  
    fmt.Println("x lebih kecil dari 5")  
  default:  
    fmt.Println("x berada di antara 5 dan 10")  
}
```

### 3. Perulangan (**for**)

- Bahasa Go hanya memiliki satu jenis perulangan, yaitu for. Namun, for ini fleksibel dan dapat digunakan untuk berbagai bentuk perulangan.
- Bentuk dasar for

```
for i := 0; i < 10; i++ {  
  fmt.Println(i)  
}
```

- Perulangan tanpa kondisi (**infinite loop**)

```
for {  
  fmt.Println("Perulangan tanpa akhir")  
}
```

- Digunakan untuk mengiterasi elemen dalam slice, array, map, atau channel :

```
for index, value := range collection {  
  fmt.Println(index, value)  
}
```

#### 4. Pernyataan **break** dan **continue**

- **break** digunakan untuk menghentikan eksekusi dari perulangan atau **switch**.

```
for i := 0; i < 10; i++ {  
    if i == 5 {  
        break // keluar dari loop saat i == 5  
    }  
}
```

- **continue** digunakan untuk melewati eksekusi kode yang tersisa dalam iterasi dan langsung lanjut ke iterasi berikutnya.

```
for i := 0; i < 10; i++ {  
    if i % 2 == 0 {  
        continue // loncat ke iterasi berikutnya jika i genap  
    }  
    fmt.Println(i) // hanya akan mencetak bilangan ganjil  
}
```

#### 5. Pernyataan (**goto**)

- Go mendukung **goto**, meskipun jarang digunakan karena dapat membuat kode sulit diikuti. **goto** memindahkan eksekusi program ke label yang ditentukan.

```
func main() {  
    i := 0  
loop:  
    if i < 5 {  
        fmt.Println(i)  
        i++  
        goto loop  
    }  
}
```

#### 6. Pernyataan (**defer**)

- **defer** digunakan untuk menunda eksekusi fungsi sampai fungsi yang menampung **defer** selesai.

```
func contoh() {  
    defer fmt.Println("Ini dieksekusi terakhir")  
    fmt.Println("Ini dieksekusi pertama")  
}
```

- **defer** sering digunakan untuk menutup resource seperti file atau koneksi jaringan, memastikan resource tersebut ditutup meskipun terjadi error.

## 7. **Panic dan Recover**

- **Panic** digunakan untuk menghentikan eksekusi program secara tiba-tiba. Biasanya digunakan untuk menangani kesalahan yang tidak bisa ditangani di runtime.

```
panic("Terjadi kesalahan!")
```

- **Recover** digunakan untuk mengontrol dan mengatasi panic, sehingga program bisa kembali berjalan.

```
func contohPanic() {  
    defer func() {  
        if r := recover(); r != nil {  
            fmt.Println("Recover dari panic:", r)  
        }  
    }()  
    panic("Terjadi panic")  
}
```

## II. GUIDED

### A. Modul 2A

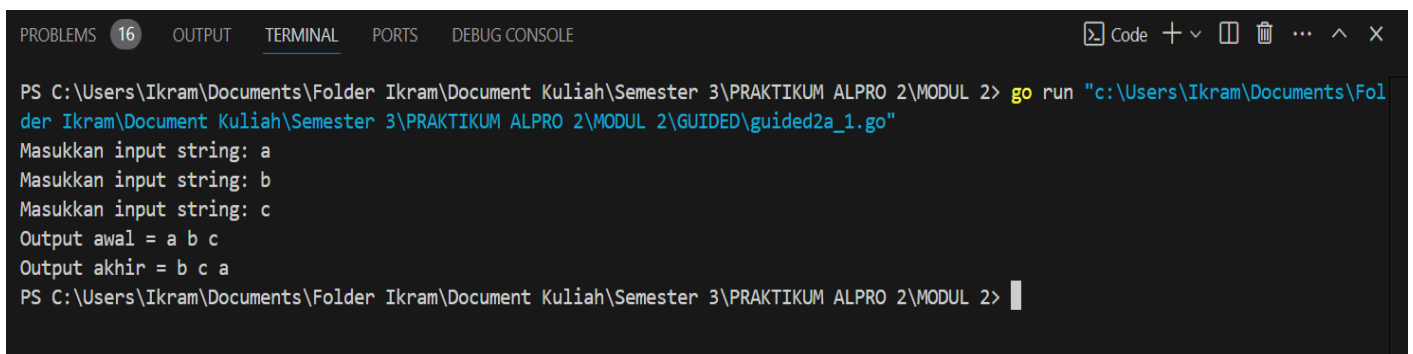
1. Telusuri program berikut dengan cara mengkompilasi dan mengeksekusi program. Silakan masukan data yang sesuai sebanyak yang diminta program. Perhatikan keluaran yang diperoleh. Coba terangkan apa sebenarnya yang dilakukan program tersebut?

```
package main

import "fmt"

func main() {
    var satu, dua, tiga string
    var temp string

    fmt.Print("Masukkan input string: ")
    fmt.Scanln(&satu)
    fmt.Print("Masukkan input string: ")
    fmt.Scanln(&dua)
    fmt.Print("Masukkan input string: ")
    fmt.Scanln(&tiga)
    fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)
    temp = satu
    satu = dua
    dua = tiga
    tiga = temp
    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
}
```



```
PROBLEMS 16 OUTPUT TERMINAL PORTS DEBUG CONSOLE
Code + - [ ] [ ] ... ^ X

PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Fol
der Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\GUIDED\guided2a_1.go"
Masukkan input string: a
Masukkan input string: b
Masukkan input string: c
Output awal = a b c
Output akhir = b c a
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> |
```

Program ini melakukan penggeseran nilai variable . pada awal program menerima tiga input melalui fmt.Scan dan disimpan pada variable satu, dua, dan tiga. Lalu nilai variable awal dicetak. Setelah itu, variable dibalik dengan cara variable temp diisi dengan nilai a, variable a diisi dengan b, variable b diisi dengan c, dan variable c diisi dengan nilai temp sehingga Ketika output akhir dicetak nilai akan bergeser menjadi b dulu lalu c baru a.

2. Tahun kabisat adalah tahun yang habis dibagi 400 atau habis dibagi 4 tetapi tidak habis dibagi 100. Buatlah sebuah program yang menerima input sebuah bilangan bulat dan memeriksa apakah bilangan tersebut merupakan tahun kabisat (true) atau bukan (false).

```
package main

import "fmt"

func main() {
    var tahun int
    fmt.Print("Tahun : ")
    fmt.Scan(&tahun)
    fmt.Printf("Kabisat : %t \n", (tahun%4 == 0 && (tahun%100 != 0 || tahun%400 == 0)))
}
```

```
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\GUIDED\guided2a_2.go"
Tahun : 2000
Kabisat : true
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\GUIDED\guided2a_2.go"
Tahun : 2016
Kabisat : true
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\GUIDED\guided2a_2.go"
Tahun : 2018
Kabisat : false
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> |
```

Program ini merupakan program untuk mengecek tahun kabisat. Pada awal program akan diminta inputan user yang akan disimpan pada variable tahun setelah itu akan dicetak menggunakan printf dengan %t menandakan bahwa tipenya adalah Boolean pengecekan dilakukan dengan memodulus tahun dengan 4 lalu memberikan operator AND yang didalam operasi akan dicek apakah tahun tidak habis dibagi serratus atau habis dibagi 400 ini dilakukan karena “Tidak semua tahun yang bisa dibagi dengan 100 adalah tahun kabisat, kecuali tahun tersebut bisa dibagi dengan 400”. Misalkan kita input 2000 maka akan berniali false jika tanpa dicek dengan modulus 400.

3. Buat program Bola yang menerima input jari-jari suatu bola (bilangan bulat). Tampilkan Volume dan Luas kulit bola.  $volumebola = \frac{4}{3}\pi r^3$  dan  $luasbola = 4\pi r^2$  ( $\pi \approx 3.1415926535$ ).

```
package main

import (
    "fmt"
    "math"
)

func main() {

    var jari, volume, luasKulit float64
    fmt.Print("Jeari = ")
    fmt.Scan(&jari)
    volume = math.Pi * (4.0 / 3.0) * math.Pow(jari, 3)
    luasKulit = 4 * math.Pi * math.Pow(jari, 2)
    fmt.Printf("Bola dengan jejari %v memiliki volume %.4f dan luas kulit %.4f\n", jari, volume, luasKulit)
}
```

```
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\GUIDED\guided2a_3.go"
Jejari = 5
Bola dengan jejari 5 memiliki volume 523.5988 dan luas kulit 314.1593
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> |
```



Program ini digunakan untuk menghitung volume dan luas kulit dari sebuah bola. Pada awal, program ini akan menerima inputan jari jari yang disimpan pada variable jari. Lalu akan dilakukan operasi matematika sesuai rumus volume bola dan disimpan pada variable volume. Setelah itu, dilakukan operasi untuk luaskulit bola dan disimpan pada variable luasKulit. Lalu hasilnya dicetak dengan printf, "%.4f" pada printf digunakan untuk membatasi 4 digit setelah koma.

### III. UNGUIDED

#### A. Modul 2B

1. 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 lainnya.

```
package main

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

func main() {

    correctOrder := []string{"merah", "kuning",
                             "hijau", "ungu"}

    reader := bufio.NewReader(os.Stdin)
    success := true

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

        input, _ := reader.ReadString('\n')
        input = strings.TrimSpace(input)

        colors := strings.Split(input, " ")
```

```

        for j :=0; j < 4; j++ {
            if colors[j] !=correctOrder[j]{
                success = false
                break
            }
        }

        if !success {
            break
        }
    }

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

```

```

PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2b_1.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 C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2b_1.go"
Percobaan 1:merah kuning hijau ungu
Percobaan 2:merah kuning hijau ungu
Percobaan 3:merah kuning hijau ungu
Percobaan 4:ungu kuning hijau merah
BERHASIL : false
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2>

```

Program ini digunakan untuk memeriksa apakah hasil percobaan suda sesuai. Pada awal program, pengguna diminta menginputkan hasil percobaan pada ke 4 gelas sebanyak 5 kali. Ini disimpan pada array 2 dimensi dengan nama variable ujiKimia. Setelah selesai menginputkan akan digunakan nested loop untuk mengecek setiap inputan. Inputan percobaan yang seharusnya telah dipetakan berdasarkan index pada variable warna. Nested loop paling luar mengakses percobaan ke-n dan yang dalamnya mengakses isi dari gelas. Jika terdapat satu saja yang tidak sesuai nilai beda akan berubah menjadi false yang akan memicu if pada nested loop dalam untuk menghentikan loop bagian dalam dan merubah nilai i pada loop

terluar menjadi 5 (untuk menghentikan loop karena  $i < 5$ , alternatif bisa menggunakan `break`). Sehingga program akan mencetak Berhasil: false jika berbeda, dan Berhasil: true jika sama semua dari percobaan 1 – 5. Break disini dilakukan agar proses menjadi lebih cepat mengingat jika satu saja berbeda maka percobaan langsung gagal, maka jika ada 1 saja yang berbeda untuk menentukan percobaan berhasil atau tidak kita tidak perlu mengecek bagian setelahnya.

2. 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.

```
package main

import "fmt"

func main() {
    // Deklarasi variabel untuk menyimpan pita dan banyaknya
    bunga
    var pita string        // Variabel 'pita' digunakan untuk
    menggabungkan nama bunga
    var banyakBunga int    // Variabel 'banyakBunga' untuk
    menghitung jumlah bunga yang dimasukkan

    // Menggunakan loop untuk meminta input bunga dari pengguna
    secara berulang
    for {
        var bunga string
        // Meminta input nama bunga
        fmt.Print("Bunga (ketik 'SELESAI' untuk berhenti): ")
        fmt.Scanln(&bunga) // Membaca input dari pengguna

        // Jika pengguna mengetik "SELESAI", maka keluar dari loop
        if bunga == "SELESAI" {
            break
        }

        // Jika pita sudah memiliki isi, tambahkan pemisah " - "
        if pita != "" {
            pita += " - "
        }
        // Menambahkan nama bunga ke pita
        pita += bunga
    }
```

```

        // Meningkatkan jumlah bunga yang dimasukkan
        banyakBunga++
    }

    // Menampilkan hasil akhir
    fmt.Println("Pita:", pita)                // Menampilkan pita yang
    berisi nama-nama bunga
    fmt.Println("Banyaknya  bunga:",  banyakBunga) //
    Menampilkan jumlah bunga yang dimasukkan
}

```

```

PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2b_2.go"
Bunga (ketik 'SELESAI' untuk berhenti): Kertas
Bunga (ketik 'SELESAI' untuk berhenti): Mawar
Bunga (ketik 'SELESAI' untuk berhenti): Tulip
Bunga (ketik 'SELESAI' untuk berhenti): SELESAI
Pita: Kertas - Mawar - Tulip
Banyaknya bunga: 3
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2>

```

Program ini akan membuat pita dengan bunga yang kita inputkan ditambah dengan “-“. Pada awal akan diminta inputan banyaknya Bunga yang akan dimasukkan dengan `fmt.Scan`. Lalu disimpan pada variable `banyak`. Setelah itu dilakukan perulangan sebanyak nilai variable `banyak`. Selama perulangan akan diminta menginputkan nilai bunga yang disimpan pada variable `bunga` sekaligus melakukan penggabungan string yang disimpan pada variable `pita`. Lalu setelah perulangan selesai, nilai `pita` akan dicetak.

3. 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 9 kg atau lebih.

```

package main

import (
    "bufio" // Mengimpor paket bufio untuk membaca input
    dari pengguna
    "fmt"    // Mengimpor paket fmt untuk menampilkan
    output

```

```

"os" // Mengimpor paket os untuk menggunakan input
dari sistem operasi
"strconv" // Mengimpor paket strconv untuk konversi
string ke tipe data numerik
"strings" // Mengimpor paket strings untuk manipulasi
string
)

func main() {
    reader := bufio.NewReader(os.Stdin) // Membuat reader
    untuk membaca input pengguna

    // Loop untuk meminta input pengguna secara berulang
    for {
        fmt.Print("Masukan berat belanjaan di kedua kantong:
")
        input, _ := reader.ReadString('\n') // Membaca input
        pengguna
        input = strings.TrimSpace(input) // Menghapus spasi
        dan karakter newline di awal dan akhir

        // Memisahkan input menjadi dua bagian berdasarkan
        spasi
        berat := strings.Split(input, " ")

        // Mengecek apakah input terdiri dari dua angka
        if len(berat) != 2 {
            fmt.Println("Input tidak valid. Masukkan dua
angka.") // Menampilkan pesan kesalahan jika input tidak
            valid
            continue // Melanjutkan
            ke iterasi berikutnya untuk meminta input ulang
        }

        // Mengonversi input dari string ke float64
        berat1, _ := strconv.ParseFloat(berat[0], 64)
        berat2, _ := strconv.ParseFloat(berat[1], 64)

        // Jika salah satu berat negatif, sepeda tidak oleng dan
        keluar dari loop
        if berat1 < 0 || berat2 < 0 {
            fmt.Println("Sepeda motor pak Andi akan oleng:
false") // Menampilkan bahwa sepeda tidak oleng
            break // Keluar dari
            loop
        }
    }
}

```

```

        // Jika total berat lebih dari 150 kg, sepeda tidak oleng
        dan keluar dari loop
        if berat1+berat2 > 150 {
            fmt.Println("Sepeda motor pak Andi akan oleng:
false") // Sepeda tidak oleng karena berat berlebihan
            break // Keluar dari
loop
        }

        // Menghitung selisih berat antara kedua kantong
        selisih := berat1 - berat2
        if selisih < 0 {
            selisih = -selisih // Mengubah selisih menjadi nilai
absolut (selisih selalu positif)
        }

        // Jika selisih berat lebih dari atau sama dengan 9 kg,
        sepeda akan oleng
        if selisih >= 9 {
            fmt.Println("Sepeda motor pak Andi akan oleng:
true")
        } else {
            fmt.Println("Sepeda motor pak Andi akan oleng:
false")
        }
    }

    // Pesan yang ditampilkan setelah proses selesai
    fmt.Println("Proses selesai.")
}

```

```

PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2b_3.go"
Masukan berat belanjaan di kedua kantong: 5 10
Sepeda motor pak Andi akan oleng: false
Masukan berat belanjaan di kedua kantong: 55.6 70.2
Sepeda motor pak Andi akan oleng: true
Masukan berat belanjaan di kedua kantong: 72.3 66.9
Sepeda motor pak Andi akan oleng: false
Masukan berat belanjaan di kedua kantong: 59.5 98.7
Sepeda motor pak Andi akan oleng: false
Proses selesai.
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2>

```

Program ini digunakan untuk mengetahui apakah selisihnya dua kantong kurang dari Sembilan dan apakah salah satu kantong lebih dari sama dengan Sembilan. Pada awal program akan menerima inputan berat1 dan berat2. Jika berat salah satu kantong lebih dari 9 maka perulangan akan berhenti. Disini kita tidak perlu mengecek apakah selisih lebih dari 9, karena beban dibatasi hanya 9 kg setiap kantong sehingga tidak mungkin memiliki selisih lebih dari 9.

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

```
package main

import (
    "fmt"
)

func main() {
    // Deklarasi variabel k dengan tipe float64
    var k float64
    // Menampilkan pesan untuk memasukkan nilai K
    fmt.Print("Nilai K = ")
    // Membaca input dari pengguna dan menyimpannya ke variabel k
    fmt.Scanln(&k)

    // Hitung f(K) dengan rumus:  $f(k) = ((4*k + 2)^2) / ((4*k + 1) * (4*k + 3))$ 
    f_k := (4*k + 2) * (4*k + 2) / ((4*k + 1) * (4*k + 3))

    // Menampilkan hasil perhitungan f(K) dengan presisi 10 angka di belakang koma
    fmt.Printf("Nilai f(K) = %.10f\n", f_k)
}
```

```
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2b_4.go"
Nilai K = 100
Nilai f(K) = 1.0000061880
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> █
```

Program ini digunakan untuk menghitung fungsi matematika seperti yang ditentukan di modul. Pada awal program pengguna diminta melakukan inputan yang akan dimasukkan ke variable k. setelah itu, nilai k dimasukkan ke persamaan sesuai yang ada di modul lalu nilainya diinsiasi ke variable hasil. Setelah itu dicetak dengan printf dengan format verbs “%.10f” untuk mengambil 10 digit angka setelah koma.

**Modifikasi program sebelumnya yang menerima input integer  $K$  dan menghitung  $\sqrt{2}$  untuk  $K$  tersebut. Hampiran  $\sqrt{2}$  dituliskan dalam ketelitian 10 angka di belakang koma.**

```
package main

import (
    "fmt"
)

func main() {
    // Deklarasi variabel k dengan tipe float64
    var k float64
    // Menampilkan pesan untuk memasukkan nilai K
    fmt.Print("Nilai K = ")
    // Membaca input dari pengguna dan menyimpannya ke variabel k
    fmt.Scanln(&k)

    // Hitung f(K) dengan rumus:  $f(k) = ((4*k + 2)^2) / ((4*k + 1) * (4*k + 3))$ 
    f_k := (4*k + 2) * (4*k + 2) / ((4*k + 1) * (4*k + 3))

    // Menampilkan hasil perhitungan f(K) dengan presisi 10 angka di belakang koma
    fmt.Printf("Akar 2 ( $\sqrt{2}$ ) = %.10f\n", f_k)
}
```

```
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2b_4.go"
Nilai K = 100
Nilai f(K) = 1.0000061880
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2b_4.go"
Nilai K = 10
Akar 2 ( $\sqrt{2}$ ) = 1.0005672150
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> |
```



Pada modifikasinya, kita diminta untuk membuat program menghitung nilai akar 2 dengan harapan sehingga kita membutuhkan perulangan sebanyak  $k+1$  kali dan setiap perulangan fungsi matematika diatas nilai  $x$  nya akan diisi dengan nilai  $i$  pada perulangan lalu dikalikan dengan nilai variable hasil sebelumnya. Setelah perulangan selesai, nilai hasil dicetak dengan printf dengan format verbs “%.10f” untuk mengambil 10 digit angka setelah koma.

## B. MODUL 2C

1. PT POS membutuhkan aplikasi perhitungan biaya kirim berdasarkan berat parsel. Maka, buatlah program BiayaPos untuk menghitung biaya pengiriman tersebut dengan ketentuan sebagai berikut!

```
package main

import "fmt"

// Fungsi untuk menghitung biaya pengiriman
func hitungBiaya(beratGram int) (int, int, int) {
    // Konversi berat dari gram ke kg
    beratKg := beratGram / 1000
    sisaGram := beratGram % 1000

    // Hitung biaya pengiriman per kg
    biayaKg := beratKg * 10000

    // Hitung biaya pengiriman untuk sisa gram
    var biayaSisa int
    if sisaGram >= 500 {
        biayaSisa = sisaGram * 5
    } else if sisaGram > 0 && beratKg <= 10 {
        biayaSisa = sisaGram * 15
    } else {
        biayaSisa = 0
    }

    return biayaKg, biayaSisa, beratKg
}

func main() {
    var beratGram int
    fmt.Print("Masukkan berat parsel (dalam gram): ")
    fmt.Scanln(&beratGram)
```

```

        // Panggil fungsi hitungBiaya untuk mendapatkan biaya
        pengiriman
        biayaKg,    biayaSisa,    beratKg    :=
        hitungBiaya(beratGram)

        // Tampilkan hasil perhitungan
        fmt.Printf("Total berat: %d kg %d gram\n", beratKg,
        beratGram%1000)
        fmt.Printf("Biaya    pengiriman:    Rp.    %d\n",
        biayaKg+biayaSisa)
        fmt.Printf(" - Biaya per kg: Rp. %d\n", biayaKg)
        fmt.Printf(" - Biaya sisa gram: Rp. %d\n", biayaSisa)
    }

```

```

PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2c_1.go"
Masukkan berat parsel (dalam gram): 8500
Total berat: 8 kg 500 gram
Biaya pengiriman: Rp. 82500
- Biaya per kg: Rp. 80000
- Biaya sisa gram: Rp. 2500
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2>

```

Program ini digunakan untuk menghitung biaya pengiriman. Pada awal program, user diminta inputan berat parsel dalam bentuk gram yang akan disimpan ke dalam variable beratParsel. Disini akan dipisahkan berat kilo dan gram, untuk berat kilo didapat dari pembagian nilai beratParsel dengan 1000, dan untuk gram dari beratParsel di modulus dengan 1000 ini akan diinisiasi pada variable parselKG dan parselGram. Lalu untuk mendapat nilai biayaKG kita mengalikan parselKG dengan 10000, untuk penghitungan biaya gramnya menggunakan if statement jika lebih dari sama dengan 15 maka untuk nilai variable biayaGram (yang menampung biaya gram) yaitu parselGram dikali dengan 5 dan jika tidak maka dikali 15. pada detail berat yang ditampilkan adalah nilai variable parselKG dan parselGram, Pada detail biaya yang ditampilkan adalah biayaKG dan biayaGram, pada total biaya yang ditampilkan merupakan hasil penjumlahan dari nilai variable biayaKG dan biayaGram.

2. Program berikut menerima input sebuah bilangan riil yang menyatakan NAM. Program menghitung NMK dan menampilkannya.

```
package main

import "fmt"

func main() {
    // Deklarasi variabel untuk menyimpan nilai numerik dan nilai huruf
    var nam float32 // Variabel 'nam' untuk menyimpan nilai numerik
    dengan tipe float32
    var nmk string // Variabel 'nmk' untuk menyimpan nilai huruf (indeks)

    // Meminta input nilai dari pengguna
    fmt.Print("Masukkan nilai: ")
    fmt.Scan(&nam) // Membaca input nilai numerik dari pengguna

    // Logika penentuan nilai huruf berdasarkan nilai numerik
    if nam >= 80 {
        nmk = "A" // Jika nilai 80 atau lebih, indeksnya A
    } else if nam >= 70 {
        nmk = "B" // Jika nilai 70 atau lebih, indeksnya B
    } else if nam >= 60 {
        nmk = "C" // Jika nilai 60 atau lebih, indeksnya C
    } else if nam >= 50 {
        nmk = "D" // Jika nilai 50 atau lebih, indeksnya D
    } else if nam >= 40 {
        nmk = "E" // Jika nilai 40 atau lebih, indeksnya E
    } else {
        nmk = "F" // Jika nilai di bawah 40, indeksnya F
    }

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

Jawablah pertanyaan-pertanyaan berikut:

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

Jawab : error, alasannya karena variable nam bertipe data float64 bukan string.

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

Jawab : seharusnya hasil nilainya disimpan pada variable nmk, if statement digabungkan menjadi if else statement, dan else terakhir untuk nilai E bisa digantikan dengan else mengingat tidak ada kondisi lain setelah e. Alurnya dengan melakukan pengecekan nam dari atas atau nilai lebih dari 80, lalu untuk statement menggunakan batas bawah dari range nilai. Sehingga batas atas dari setiap nilai ikut ke nmk bawahnya. Misalkan pada statement  $nam > 72.5$  untuk ab dan  $nam > 65$  untuk b maka nilai b dapat dicapai jika nilainya di range  $65 < nilai \leq 72.5$ .

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

Jawab :

```
package main

import "fmt"

func main() {
    // Deklarasi variabel untuk menyimpan nilai numerik
    dan nilai huruf
    var nam float64 // Variabel 'nam' untuk menyimpan nilai
    numerik dengan tipe float64
    var nmk string // Variabel 'nmk' untuk menyimpan nilai
    huruf (indeks)

    // Meminta input nilai dari pengguna
    fmt.Print("Nilai akhir mata kuliah: ")
    fmt.Scanln(&nam) // Membaca input nilai numerik dari
    pengguna
```

```
// Logika penentuan nilai huruf berdasarkan nilai
numerik
if nam >= 80 {
    nmk = "A" // Jika nilai 80 atau lebih, indeksinya A
} else if nam >= 72.5 {
    nmk = "AB" // Jika nilai 72.5 atau lebih, indeksinya B
} else if nam >= 65 {
    nmk = "B" // Jika nilai 65 atau lebih, indeksinya C
} else if nam >= 57.5 {
    nmk = "C" // Jika nilai 57.5 atau lebih, indeksinya D
} else if nam >= 40 {
    nmk = "D" // Jika nilai 40 atau lebih, indeksinya E
}

// Menampilkan hasil penentuan nilai huruf (indeks)
fmt.Println("Nilai mata kuliah: ", nmk)
}
```

```
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Fol
der Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\UNGUIDED\unguided2c_2.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Fol
der Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\UNGUIDED\unguided2c_2.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Fol
der Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\UNGUIDED\unguided2c_2.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> |
```

Untuk cara kerjanya, sama seperti yang di deskripsikan pada poin b jadi program akan menerima inputan nilai dan disimpan pada variable nam. Alurnya dengan melakukan pengecekan nam dari atas atau nilai lebih dari 80, lalu untuk statement

menggunakan batas bawah dari range nilai. Sehingga batas atas dari setiap nilai ikut ke nmk bawahnya. Misalkan pada statement  $nam > 72.5$  untuk ab dan  $nam > 65$  untuk b maka nilai b dapat dicapai jika nilainya di range  $65 < nilai \leq 72.5$ . lalu nilai variable nmk dicetak bersama dengan string “Nilai mata kuliah : “

3. Buatlah program yang menerima input sebuah bilangan bulat b dan  $b > 1$ . Program harus dapat mencari dan menampilkan semua faktor dari bilangan tersebut!

```

package main

import (
    "fmt"
)

func main() {
    // Deklarasi variabel untuk menyimpan input bilangan
    var b int

    // Meminta pengguna untuk memasukkan bilangan
    fmt.Print("Bilangan: ")
    fmt.Scanln(&b)

    // Mencetak faktor-faktor dari bilangan
    fmt.Print("Faktor: ")
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Print(i, " ")
        }
    }

    // Mengecek apakah bilangan adalah bilangan prima
    prima := true
    if b <= 1 {
        prima = false
    } else {
        // Loop untuk mencari pembagi selain 1 dan b
        for i := 2; i*i <= b; i++ {
            if b%i == 0 {
                prima = false
                break
            }
        }
    }

    // Menampilkan apakah bilangan prima atau bukan
    fmt.Println("\nPrima:", prima)
}

```

```

PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2c_3.go"
Bilangan: 12
Faktor: 1 2 3 4 6 12
Prima: false
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2> go run "c:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2\unguided2c_3.go"
Bilangan: 7
Faktor: 1 7
Prima: true
PS C:\Users\Ikram\Documents\Folder Ikram\Document Kuliah\Semester 3\PRAKTIKUM ALPRO 2\MODUL 2>

```

Program ini digunakan untuk mencari nilai bilangan prima, pada awal program pengguna diminta input angka yang akan disimpan pada variable number. Cara mencari faktornya dengan melakukan perulangan sebanyak bilangan tersebut dimulai dari angka 1. Jika bilangan yang diinput di modulo dengan nilai i pada perulangan habis dibagi maka bilangan tersebut faktor dan akan dicetak pada terminal. Sebelum mencari faktor akan dilakukan pengecekan apakah bilangan yang diinputkan kurang dari atau sama dengan 1 jika iya maka akan dicetak bilangan harus lebih dari 1 alu program dihentikan dengan return.

**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.**

```

package main

import (
    "fmt"
)

func main() {
    // Deklarasi variabel untuk menyimpan input bilangan
    var b int

    // Meminta pengguna untuk memasukkan bilangan
    fmt.Print("Bilangan: ")
    fmt.Scanln(&b)
}

```

```

// Mencetak faktor-faktor dari bilangan
fmt.Print("Faktor: ")
for i := 1; i <= b; i++ {
    if b%i == 0 {
        fmt.Print(i, " ")
    }
}

// Mengecek apakah bilangan adalah bilangan prima
prima := true
if b <= 1 {
    prima = false
} else {
    // Loop untuk mencari pembagi selain 1 dan b
    for i := 2; i*i <= b; i++ {
        if b%i == 0 {
            prima = false
            break
        }
    }
}

// Menampilkan apakah bilangan prima atau bukan
fmt.Println("\nPrima:", prima)
}

```

Pada lanjutan program hanya dilakukan sedikit penambahan yaitu penambahan variable countFaktor lalu melakukan increment countFaktor setiap faktor dari bilangan ditemukan. Dan terakhir penambahan print dengan operator logika yang akan mengecek apakah banyaknya faktor hanya dua, jika iya maka bilangan tersebut prima (bilangan prima sendiri hanya memiliki faktor bilangan 1 dan bilangan itu sendiri, bilangan selain prima otomatis punya lebih dari 2 faktor karena bilangan 1 dan bilangan itu sendiri ikut terhitung).