

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL 2
REVIEW STRUKTUR KONTROL**



Oleh:

GALIH TRISNA

2311102050

IF-11-02

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024**

I. DASAR TEORI

A. Struktur Program Go

Struktur dasar program Go terdiri dari beberapa bagian, antara lain:

- Package Main
Setiap file go harus berada dalam sebuah package
- Function Main
Fungsi yang dieksekusi saat program dimulai

B. Tipe Data dan Instruksi Dasar

a. Data dan Variabel

- Tipe data : integer (int, int8, int16, int32, int64), float (float32, float64), boolean, dan string.
- Variable
Dengan var “var age int = 30” dan dengan shorthand “age := 30”

b. Instruksi dasar

- Operasi Matematis : ‘+’, ‘-’, ‘*’, ‘/’, ‘%’
- Operasi Logika : ‘&&’, ‘||’, ‘!’

c. Konstanta Simbolik

Merupakan konstanta atau variabel yang tidak dapat diubah
“const pi = 3.14”

C. Struktur Kontrol Perulangan

a. Bentuk While Loop

Pada Go tidak memiliki while, namun tetap dapat dilakukan dengan menggunakan for tanpa kondisi inisialisasi

```
i := 0
for i < 5 {
    fmt.Println(i)
    i++
}
```

b. Bentuk Repeat Until

Pada Go tidak memiliki while, namun tetap dapat dilakukan dengan menggunakan for dan kondisi break

```
i := 0
for {
    fmt.Println(i)
    i++
    if i >= 5 {
```

```
        break
    }
}
```

D. Struktur Kontrol Percabangan

a. Bentuk If Else

Dalam go, tidak memerlukan diapit oleh tanda kurung

```
if age > 18 {
    fmt.Println("Adult")
} else {
    fmt.Println("Not an adult")
}
```

b. Bentuk Switch Case

```
switch day := 2; day {
case 1:
    fmt.Println("Monday")
case 2:
    fmt.Println("Tuesday")
default:
    fmt.Println("Unknown day")
}
```

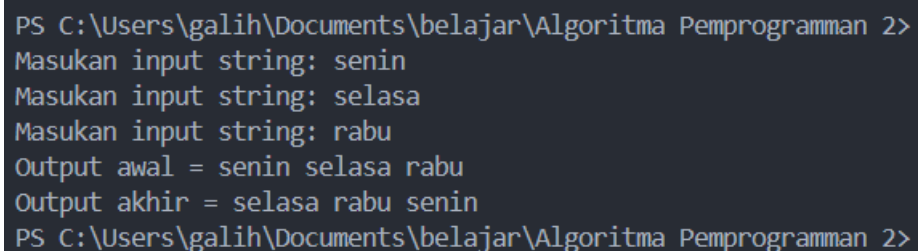
II. GUIDED

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
        temp string
    )
    fmt.Print("Masukan input string: ")
    fmt.Scanln(&satu)
    fmt.Print("Masukan input string: ")
    fmt.Scanln(&dua)
    fmt.Print("Masukan 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)
}
```



```
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
Masukan input string: senin
Masukan input string: selasa
Masukan input string: rabu
Output awal = senin selasa rabu
Output akhir = selasa rabu senin
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
```

program tersebut adalah program mengembalikan urutan dari susunan angka atau kata. User diminta untuk menginputkan string 3 kali yang akan

dimasukan ke dalam variable satu, dua, dan tiga. Kemudian program menampilkan variabel yang user inputkan dengan urutan variable satu, dua, dan tiga. kemudian program akan mengisi value variable dari temp dengan variable satu, kemudian mengisi variable satu dengan variable dua, mengisi variable dua dengan variable tiga, dan mengisi variable tiga dengan variable temp. Kemudian program akan menampilkan variabel dengan urutan variable satu, dua, dan tiga.

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). (Contoh input/output, Teks bergaris bawah adalah input dari user):

1	Tahun: <u>2016</u> Kabisat: true
2	Tahun: <u>2000</u> Kabisat: true
3	Tahun: <u>2018</u> Kabisat: false

```
package main
import "fmt"
```

```
func main(){
    var tahun int
    var kabisat bool
    kabisat = false
    fmt.Print("Tahun: ")
    fmt.Scanln(&tahun)
    if tahun%4==0 && tahun%100!=0 {
        kabisat = true
    }
    println("Kabisat: ",kabisat)
}
```

```
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
Tahun: 2024
Kabisat: true
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
Tahun: 2025
Kabisat: false
```

Program diatas adalah program untuk menentukan apakah tahun yang diinputkan oleh user merupakan tahun kabisat atau bukan. Program dimulai dengan mendeklarasikan variabel integer tahun dan variabel boolean kabisat dengan value false. Kemudian user diminta menginputan value untuk variabel tahun. Kemudian program menggunakan percabangan if else untuk menentukan tahun tersebut kabisat atau bukan dengan cara jika tahun sisa hasil baginya dengan 4 sama dengan 0 dan tahun sisa hasil baginya dengan 100 tidak sama dengan 0 maka variable kabisat diberi value true. Kemudian program akan mencetak value variabel dari kabisat.

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 $luasbola = 4\pi r^2$ ($\pi \approx 3.1415926535$). (Contoh input/output, Teks bergaris bawah adalah input dari user):

<pre>Jejari = 5 Bola dengan jejari 5 memiliki volume 523.5988 dan luas kulit 314.1593</pre>

```
package main
import "fmt"
import "math"
```

```
func main(){
    var jari, luas, volume float64
    fmt.Print("Jejari = ")
    fmt.Scanln(&jari)
    volume = 4*math.Pi*math.Pow(jari,3)/3
    luas = 4*math.Pi*math.Pow(jari,2)
    fmt.Println("Bola dengan jejari ",jari," memiliki volume ",volume," dan
luas kulit ",luas)
}
```

```
PS C:\Users\galih\Documents\belajar\Algoritma Pemrograman 2> go run "c:\Users\galih\Documents\belajar\Algoritma Pemrograman 2\1-011024\2a\3\main.go"
Jejari = 5
Bola dengan jejari 5 memiliki volume 523.5987755982989 dan luas kulit 314.1592653589793
PS C:\Users\galih\Documents\belajar\Algoritma Pemrograman 2> |
```

Program tersebut merupakan program untuk menghitung volume dan luas dari permukaan bola berdasarkan input yang diberikan oleh user. User menginput jari jari dengan tipe data float. Kemudian variabel volume diisi dengan value yang berasal dari perhitungan rumus $4 \times \text{math.Pi} \times \text{math.Pow}(\text{jari}, 3) / 3$ dan untuk variabel luas yaitu $4 \times \text{math.Pi} \times \text{math.Pow}(\text{jari}, 2)$. Dalam rumus ini menggunakan bantuan fungsi `math.Pi` untuk π dan `math.Pow` untuk perpangkatan. Kemudian program akan menampilkan value dari variabel jari, volume, dan luas

4. Dibaca nilai temperatur dalam derajat Celsius. Nyatakan temperatur tersebut dalam Fahrenheit $Celsius = (Fahrenheit - 32) \times \frac{5}{9}$ Reamur $Reamur = Celsius \times \frac{4}{5}$ Kelvin $Kelvin = (Fahrenheit + 459.67) \times \frac{5}{9}$ (Contoh input/output, Teks bergaris bawah adalah input dari user):

```
Temperatur Celsius: 50
Derajat Fahrenheit: 122
```

Lanjutkan program di atas, sehingga temperatur dinyatakan juga dalam derajat Reamur dan Kelvin. (Contoh input/output, Teks bergaris bawah adalah input dari user):

```
Temperatur Celsius: 50
Derajat Reamur: 40
Derajat Fahrenheit: 122
Derajat Kelvin: 323
```

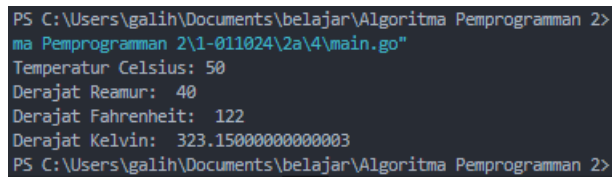
```
package main
import "fmt"
```

```
func main(){
    var celcius, reamur, fahrenheit, kelvin float64
    fmt.Print("Temperatur Celsius: ")
    fmt.Scanln(&celcius)
    reamur = celcius/5*4
```

```

fahrenheit = celcius*9/5+32
kelvin = (fahrenheit+459.67)/9*5
fmt.Println("Derajat Reamur: ", reamur)
fmt.Println("Derajat Fahrenheit: ", fahrenheit)
fmt.Println("Derajat Kelvin: ", kelvin)
}

```



```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrograman 2>
ma Pemrograman 2\1-011024\2a\4\main.go
Temperatur Celsius: 50
Derajat Reamur: 40
Derajat Fahrenheit: 122
Derajat Kelvin: 323.15000000000003
PS C:\Users\galih\Documents\belajar\Algoritma Pemrograman 2>

```

Program diatas merupakan program untuk mengonversi suhu dari celcius ke reamur, fahrenheit, dan kelvin. User diminta untuk mengisi variable float celcius. Kemudian value variable reamur didapatkan dari perhitungan $\text{celcius} / 5 * 4$. Value variable fahrenheit didapatkan dari perhitungan $(\text{celcius} * 9 / 5) + 32$. Value variable kelvin didapatkan dari perhitungan $((\text{fahrenheit} + 459.67) / 9) * 5$. Kemudian hasil dari konversi tersebut akan di tampilkan oleh program.

5. Tipe karakter sebenarnya hanya apa yang tampak dalam tampilan. Di dalamnya tersimpan dalam bentuk biner 8 bit (byte) atau 32 bit (rune) saja. Buat program ASCII yang akan membaca 5 buah data integer dan mencetaknya dalam format karakter. Kemudian membaca 3 buah data karakter dan mencetak 3 buah karakter setelah karakter tersebut (menurut tabel ASCII) Masukan terdiri dari dua baris. Baris pertama berisi 5 buah data integer. Data integer mempunyai nilai antara 32 s.d. 127. Baris kedua berisi 3 buah karakter yang berdampingan satu dengan yang lain (tanpa dipisahkan spasi). Keluaran juga terdiri dari dua baris. Baris pertama berisi 5 buah representasi karakter dari data yang diberikan, yang berdampingan satu dengan lain, tanpa dipisahkan spasi. Baris kedua berisi 3 buah karakter (juga tidak dipisahkan oleh spasi).

No.	Masukan	Keluaran
1	66 97 103 117 115 SNO	Bagus TOP

```
package main
```

```
import "fmt"
```

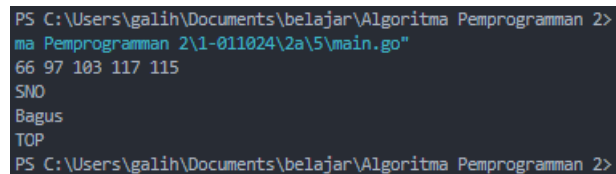


```

func main() {
    var nums [5]int
    var chars [3]rune
    for i := 0; i < 5; i++ {
        fmt.Scan(&nums[i])
    }
    fmt.Scanln()
    for i := 0; i < 3; i++ {
        fmt.Scanf("%c", &chars[i])
    }

    for i := 0; i < 5; i++ {
        fmt.Printf("%c", nums[i])
    }
    fmt.Println()
    for i := 0; i < 3; i++ {
        fmt.Printf("%c", chars[i]+1)
    }
}

```



```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
ma Pemrogramman 2\1-011024\2a\5\main.go
66 97 103 117 115
SNO
Bagus
TOP
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>

```

Program diatas merupakan program meminta user untuk memasukan 5 angka dan 3 karakter, kemudian mengubah dan menampilkan hasilnya. Program mendeklarasikan array nums untuk menyimpan 5 bilangan bulat dan array chars untuk menyimpan 3 karakter bertipe rune. User akan memasukan 5 angka kedalam array nums dan juga memasukan 3 karakter ke dalam chars. Kemudian program menampilkan 5 angka dalam bentuk karakter ASCII yang sesuai. Setelah itu program akan menampilkan 3 karakter dari char namun nilai pada setiap karakter di tambah 1 sehingga menghasilkan karakter selanjutnya dalam urutan unicode

III. UNGUIDED

2B

1. 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 berturut-turut 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 lainnya. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

Percobaan 1:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
Percobaan 2:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
Percobaan 3:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
Percobaan 4:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
Percobaan 5:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
BERHASIL: true				
Percobaan 1:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
Percobaan 2:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
Percobaan 3:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
Percobaan 4:	<u>ungu</u>	<u>kuning</u>	<u>hijau</u>	<u>merah</u>
Percobaan 5:	<u>merah</u>	<u>kuning</u>	<u>hijau</u>	<u>ungu</u>
BERHASIL: false				

```
package main
```

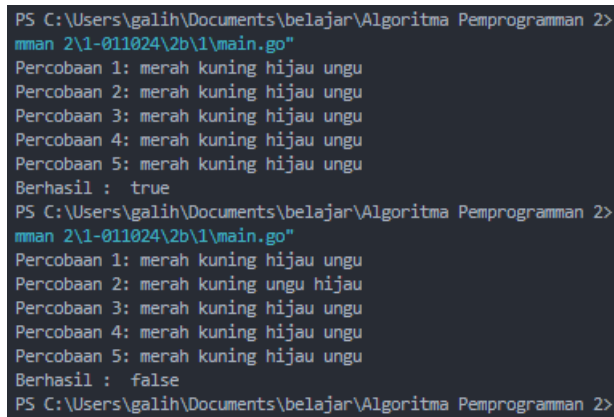
```
import "fmt"
```

```
func main() {  
    var warna [4]string  
    var warna_urutan [4]string  
    warna_urutan = [4]string{"merah","kuning","hijau","ungu"}  
    var berhasil bool  
    berhasil = true  
    for i := 0; i < 5; i++ {  
        fmt.Print("Percobaan ",i+1,"": "  
        for i := 0; i < 4; i++ {
```

```

        fmt.Scan(&warna[i])
    }
    for i := 0; i < 4; i++{
        if warna[i] != warna_urutan[i] {
            berhasil = false
        }
    }
}
fmt.Println("Berhasil : ",berhasil)
}

```



```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmman 2\1-011024\2b\1\main.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\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmman 2\1-011024\2b\1\main.go
Percobaan 1: merah kuning hijau ungu
Percobaan 2: merah kuning ungu hijau
Percobaan 3: merah kuning hijau ungu
Percobaan 4: merah kuning hijau ungu
Percobaan 5: merah kuning hijau ungu
Berhasil : false
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>

```

Program ini adalah program yang mengecek urutan warna apakah sama dengan urutan yang telah ditentukan atau tidak sebanyak 5 kali berturut turut. Terdapat variabel array warna_urutan yang berisi urutan warna yang benar yaitu merah, kuning, hijau, ungu. Kemudian user akan menginputkan urutan yang user miliki sebanyak 5 percobaan ke variabel array warna. Kemudian terdapat juga variabel boolean berhasil yang berisi value true. Dalam setiap percobaan, program akan mengecek apakah urutan pada percobaan tersebut sesuai atau tidak. Jika tidak maka variabel boolean berhasil akan diisi dengan false. Kemudian program menampilkan apakah variabel berhasil true atau false

2. 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.

(Petunjuk: gunakan operasi penggabungan string dengan operator “+”).
Tampilkan isi pita setelah proses input selesai. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

N: <u>3</u>	N : <u>0</u>
Bunga 1: <u>Kertas</u>	Pita :
Bunga 2: <u>Mawar</u>	
Bunga 3: <u>Tulip</u>	
Pita: Kertas - Mawar - Tulip -	

Modifikasi program sebelumnya, proses input akan berhenti apabila user mengetikkan ‘SELESAI’. Kemudian tampilkan isi pita beserta banyaknya bunga yang ada di dalam pita Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

Bunga 1: <u>Kertas</u>	Bunga 1: <u>SELESAI</u>
Bunga 2: <u>Mawar</u>	Pita :
Bunga 3: <u>Tulip</u>	Bunga: 0
Bunga 4: <u>SELESAI</u>	
Pita: Kertas - Mawar - Tulip -	
Bunga: 3	

package main

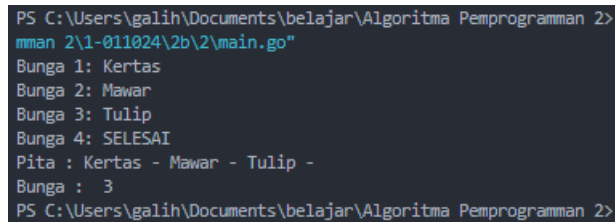
import "fmt"

```
func main() {
    var n int
    var bunga [20]string
    n = 0
    for {
        fmt.Print("Bunga ", n+1, ": ")
        fmt.Scan(&bunga[n])
        if bunga[n] == "SELESAI" {
            break
        }
        n++
    }
    fmt.Print("Pita : ")
    for i := 0; i < n; i++ {
```

```

        fmt.Print(bunga[i], " - ")
    }
    fmt.Println("\nBunga : ", n)
}

```



```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mman 2\1-011024\2b\2\main.go
Bunga 1: Kertas
Bunga 2: Mawar
Bunga 3: Tulip
Bunga 4: SELESAI
Pita : Kertas - Mawar - Tulip -
Bunga : 3
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>

```

Program diatas merupakan program untuk mencatat nama nama bunga sampai user mengetikan SELESAI. Program mendeklarasikan variable n untuk menghitung jumlah bunga yang dimasukkan dan variabel array bunga untuk tempat menyimpan nama nama bunga. User akan diminta memasukan nama bunga terus menerus didalam perulangan sampai user memasukan SELESAI untuk berhenti memasukan nama bunga. Kemudian program akan menampilkan nama nama bunga dari array bunga dan kemudian menampilkan jumlah bunga yang telah di masukan user yang berasal dari variabel n

3. 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 9 kg atau lebih. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

Masukan berat belanjaan di kedua kantong: <u>5.5</u> <u>1.0</u> Masukan berat belanjaan di kedua kantong: <u>7.1</u> <u>8.5</u> Masukan berat belanjaan di kedua kantong: <u>2</u> <u>6</u> Masukan berat belanjaan di kedua kantong: <u>9</u> <u>5.8</u> Proses selesai.

Pada modifikasi 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 150 kg atau salah satu kantong beratnya negatif. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

```
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
Proses selesai.
```

```
package main
```

```
import "fmt"
```

```
func main() {
    var oleng bool
    var kanan, kiri float64
    for {
        oleng = false
        fmt.Println("\nMasukan berat belanjaan di kedua kantong: ")
        fmt.Scanln(&kanan, &kiri)
        if kiri > 0 && kanan > 0 {
            if kiri > kanan && kiri-kanan >= 9 {
                oleng = true
            }
            if kanan > kiri && kanan-kiri >= 9 {
                oleng = true
            }
        }
        if kiri+kanan > 150 {
            fmt.Println("Program Selesai.")
            break
        } else {
            fmt.Println("Sepeda motor pak Andi akan oleng: ",oleng)
        }
    }
}
```

```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrograman 2>
mmman 2\1-011024\2b\3\main.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
Program Selesai.
PS C:\Users\galih\Documents\belajar\Algoritma Pemrograman 2>

```

Program diatas adalah program yang menentukan apakah sepeda motor pak andi akan oleng atau tidak. Program mendeklarasikan variabel boolean oleng, float kanan, dan float kiri. kemudian terdapat perulangan yang didalam nya user diminta untuk menginputkan berat kantong kanan dan kantong kiri. kemudian terdapat percabangan untuk memeriksa apakah berat kiri dan kanan lebih dari 0 atau tidak. Jika iya kemudian terdapat percabangan lagi didalamnya yaitu untuk mengecek apakah selisih dari bobot kiri dan kanan lebih atau sama dengan 9 maka variabel oleng diberi value true. Program tersebut akan terus berulang sampai user memasukan berat kantong kiri dan kanan jika ditambahkan menghasilkan berat lebih dari 150.

SA

4. Diberikan sebuah persamaan sebagai berikut ini.

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

Buatlah sebuah program yang menerima input sebuah bilangan sebagai K, kemudian menghitung dan menampilkan nilai f(K) sesuai persamaan di atas. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

Nilai K = <u>100</u> Nilai f(K) = 1.0000061880

$\sqrt{2}$ merupakan bilangan irasional. Meskipun demikian, nilai tersebut dapat dihipir dengan rumus berikut

$$\sqrt{2} = \prod_{k=0}^{\infty} \frac{(4k + 2)^2}{(4k + 1)(4k + 3)}$$

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. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

1	Nilai K = 10 Nilai akar 2 = 1.4062058441
2	Nilai K = 100 Nilai akar 2 = 1.4133387072
3	Nilai K = 1000 Nilai akar 2 = 1.4141252651

package main

```
import "fmt"
func main() {
    var k int
    fmt.Print("Nilai K = ")
    fmt.Scanln(&k)
    var result float64 = 1.0
    for i := 0; i <= k; i++ {
        result *= float64((4*i + 2) * (4*i + 2)) / float64((4*i + 1) * (4*i + 3))
    }
    fmt.Printf("Nilai akar 2 = %.10f\n", result)
}
```

```
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmn 2\1-011024\2b\4\main.go
Nilai K = 10
Nilai akar 2 = 1.4062058441
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmn 2\1-011024\2b\4\main.go
Nilai K = 100
Nilai akar 2 = 1.4133387072
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmn 2\1-011024\2b\4\main.go
Nilai K = 1000
Nilai akar 2 = 1.4141252651
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
```

Program diatas adalah program untuk menghitung hasil dari sebuah rumus persamaan, user akan menginputkan nilai k.kemudia k akan dihitung dengan rumus persamaan berikut

$$\sqrt{2} = \prod_{k=0}^{\infty} \frac{(4k + 2)^2}{(4k + 1)(4k + 3)}$$

Kemudian program setelah selesai mennghitung akan menampilkan hasilnya.

2C

1. PT POS membutuhkan aplikasi perhitungan biaya kirim berdasarkan berat parcel. Maka, buatlah program BiayaPos 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 1kg) digratiskan biayanya apabila total berat ternyata lebih dari 10kg. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

1	Contoh #1 Berat parcel (gram): <u>8500</u> Detail berat: 8 kg + 500 gr Detail biaya: Rp. 80000 + Rp. 2500 Total biaya: Rp. 82500
2	Contoh #2 Berat parcel (gram): <u>9250</u> Detail berat: 9 kg + 250 gr Detail biaya: Rp. 90000 + Rp. 3750 Total biaya: Rp. 93750
3	Contoh #3 Berat parcel (gram): <u>11750</u> Detail berat: 11 kg + 750 gr Detail biaya: Rp. 110000 + Rp. 3750 Total biaya: Rp. 110000

package main

import "fmt"

func main() {

var beratParsel, kg, sisaGram, biayaDasar, biayaTambahan , totalBiaya
int

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

fmt.Scanln(&beratParsel)

kg = beratParsel / 1000

sisaGram = beratParsel % 1000

```

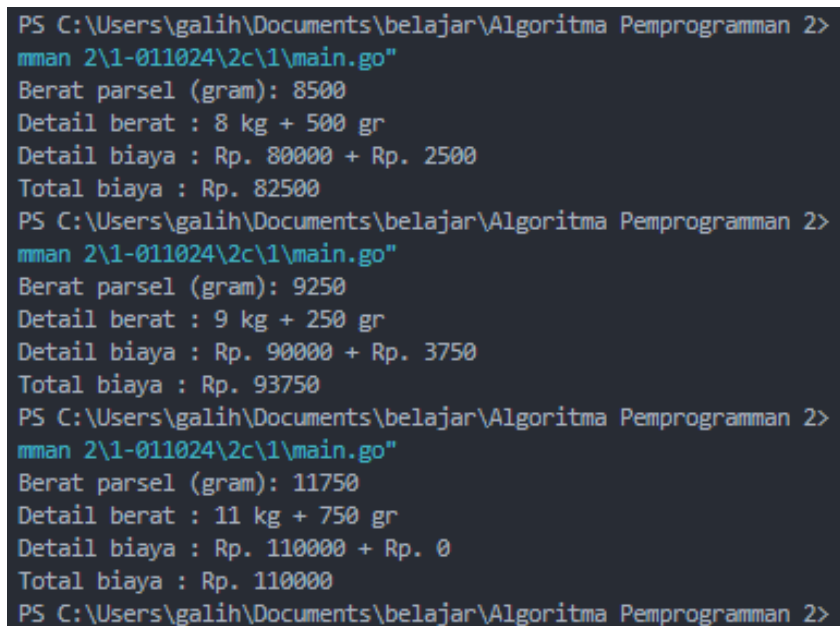
biayaDasar = kg * 10000

if sisaGram >= 500 {
    biayaTambahan = sisaGram * 5
} else if sisaGram > 0 {
    biayaTambahan = sisaGram * 15
}

if kg > 10 && sisaGram < 1000 {
    biayaTambahan = 0
}

totalBiaya = biayaDasar + biayaTambahan
fmt.Println("Detail berat :",kg,"kg +", sisaGram, "gr")
fmt.Println("Detail biaya : Rp.",biayaDasar, "+ Rp.", biayaTambahan)
fmt.Println("Total biaya : Rp.",totalBiaya)
}

```



```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmman 2\1-011024\2c\1\main.go"
Berat parcel (gram): 8500
Detail berat : 8 kg + 500 gr
Detail biaya : Rp. 80000 + Rp. 2500
Total biaya : Rp. 82500
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmman 2\1-011024\2c\1\main.go"
Berat parcel (gram): 9250
Detail berat : 9 kg + 250 gr
Detail biaya : Rp. 90000 + Rp. 3750
Total biaya : Rp. 93750
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
mmman 2\1-011024\2c\1\main.go"
Berat parcel (gram): 11750
Detail berat : 11 kg + 750 gr
Detail biaya : Rp. 110000 + Rp. 0
Total biaya : Rp. 110000
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>

```

Program diatas merupakan program untuk menghitung biaya parcel berdasarkan beratnya dalam satuan gram. Program akan meminta user untuk menginputkan berat parcel. Lalu dengan menggunakan modulus, program dapat menentukan jumlah kg dan sisa gram dari berat total. Kemudian program akan menghitung biaya dasar berdasarkan berat kg

yaitu 10.000 per kg. Kemudian sisa gram jika lebih atau sama dnegan 500 gram makan dikenakan biaya tambahan 5 per gram. Sedangkan jika kurang dari 500 maka dikenakan biaya 15 gram. Namun jika berat parcel lebih dari 10kg maka biaya tambahan nya akan dihapus atau menjadi 0. Kemudian program menampilkan total biaya yang didapatkan dari penjumlahan biaya dasar dengan biaya tambahan.

2. 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 <= 80	AB
65 < NAM <= 72.5	B
57.5 < NAM <= 65	BC
50 < NAM <= 57.5	C
40 < NAM <= 50	D
NAM <= 40	E

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

```
package main
import "fmt"
func main() {
    var nam float64
    var nmk string
    fmt.Print("Nilai akhir mata kuliah: ")
    fmt.Scanln(&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)
}
```

Jawablah pertanyaan-pertanyaan berikut:

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

Program tidak dapat dijalankan karena kesalahan variabel. Seharusnya variabel yang digunakan untuk menyimpan nilai yaitu nmk yang bertipe data string bukan nam yang bertipe data float

- b. Apa saja kesalahan dari program tersebut? Mengapa demikian? Jelaskan alur program seharusnya!
- “ ” seharusnya " "
 - penggunaan percabangan yang salah seharusnya menggunakan if elseif else alih alih menggunakan if if if
 - variable yang digunakan untuk menyimpan nilai seharusnya menggunakan nmk dengan tipe data string bukan nam yang bertipe data float
- c. Perbaiki program tersebut! Ujilah dengan masukan: 93.5; 70.6; dan 49.5. Seharusnya keluaran yang diperoleh adalah ‘A’, ‘B’, dan ‘D’.

```
package main
import "fmt"
func main() {
    var nam float64
    var nmk string
    fmt.Print("Nilai akhir mata kuliah: ")
    fmt.Scanln(&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)
}

```

```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> go run .\1-011024\2c\2\tempCodeRunnerFile.go
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> go run .\1-011024\2c\2\tempCodeRunnerFile.go
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> go run .\1-011024\2c\2\tempCodeRunnerFile.go
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>

```

3. 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! Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

Bilangan: <u>12</u>	Bilangan: <u>7</u>
Faktor: 1 2 3 4 6 12	Faktor: 1 7

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. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

Bilangan: <u>12</u>	Bilangan: <u>7</u>
Faktor: 1 2 3 4 6 12	Faktor: 1 7
Prima: false	Prima: true

```

package main

import "fmt"

func main(){
    var b int
    var f []int
    fmt.Print("Bilangan : ")
    fmt.Scan(&b)
    if b > 0 {
        for i := 1; i <= b; i++ {
            if b % i == 0 {
                f = append(f, i)
            }
        }
    }
    fmt.Println("Faktor :",f)
    if len(f) == 2 {
        fmt.Println("Prima: true")
    } else {
        fmt.Println("Prima: false")
    }
}

```

Program diatas adalah program untuk mencari faktor pada sebuah bilangan dan menentukan apakah bilangan prima atau bukan. Program mendeklarasikan variabel integer b dan array integer f. Kemudian user diminta untuk menginputkan bilangan yang akan disimpan ke variabel b. Jika b lebih dari 0 maka akan terjadi perulangan yaitu untuk mengisi array f dengan indeks perulangan jika b memiliki sisa hasil bagi dengan i sama dengan 0. Kemudian program menampilkan nilai array f. Kemudian jika panjang array nya 2 maka program menampilkan prima : true sedangkan jika tidak maka prima : false