

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II
REVIEW STRUKTUR KONTROL**



Disusun Oleh :

Egi Umar Ferdhika / 2311102277

11-IF-06

Dosen Pengampu :

Abednego Dwi Septiadi, S.Kom., M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Go (Golang) adalah bahasa pemrograman yang dikembangkan oleh Google, dirancang untuk efisiensi, performa tinggi, dan kesederhanaan. Dalam pengembangan aplikasi, model digunakan sebagai representasi dari data atau entitas dalam sistem, terutama dalam arsitektur Model-View-Controller (MVC), di mana model bertanggung jawab untuk menangani data dan logika bisnis. Pada Go, model diimplementasikan menggunakan struct, yang memungkinkan pengelompokan variabel dengan tipe berbeda ke dalam satu entitas. Sebagai contoh, struct User dapat digunakan untuk merepresentasikan pengguna dengan properti seperti ID, Name, Email, dan CreatedAt. Struct dalam Go juga mendukung penggunaan tag, seperti tag `json:"name"`, yang mengatur bagaimana field tersebut dipetakan ketika di-encode atau di-decode dari JSON.

Selain itu, Go memungkinkan penambahan metode pada struct untuk menjalankan logika bisnis tertentu, seperti memvalidasi data. Dalam pengembangan aplikasi yang melibatkan database, library seperti GORM memudahkan pemetaan struct ke tabel dalam database, di mana field struct otomatis dipetakan ke kolom tabel.

Struktur model biasanya ditempatkan dalam folder models, dan digunakan dalam controller atau service untuk memanipulasi data, serta berinteraksi dengan database melalui ORM atau query SQL manual. Dengan fitur-fitur ini, Go menawarkan pendekatan yang sederhana dan efisien dalam membangun model yang scalable untuk berbagai aplikasi modern.

II. GUIDED

1. Berisi source code dan output dari kegiatan praktikum yang telah dilaksanakan. Source Code diberi penjelasan maka akan menjadi nilai ++

a. Soal Studi Case

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?

```
1 package main
2 import "fmt"
3
4 func main() {
5     var (
6         satu, dua, tiga string
7         temp string
8     )
9     fmt.Print("Masukan input string: ")
10    fmt.Scanln(&satu)
11    fmt.Print("Masukan input string: ")
12    fmt.Scanln(&dua)
13    fmt.Print("Masukan input string: ")
14    fmt.Scanln(&tiga)
15    fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)
16    temp = satu
17    satu = dua
18    dua = tiga
19    tiga = temp
20    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
21 }
```

Sourcecode

```
package main
import (
    "fmt"
    "math"
)

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)

}

```

Screenshoot Output

```

PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run "d:\Kuliah\SEM 3\alpro 2\kode\kode(2)\guided1_modul2.go"
Masukan input string: 5
Masukan input string: 6
Masukan input string:3
Output awal = 5 6 3
Output akhir = 6 3 5

```

Deskripsi Program

Program ini meminta pengguna untuk memasukkan tiga string, kemudian menampilkan urutan awal dari string-string tersebut. Setelah itu, program akan melakukan rotasi nilai, di mana string pertama disimpan ke yang kedua, yang kedua ke yang ketiga, dan yang ketiga ke yang pertama, menggunakan variabel temp untuk menyimpan sementara nilai. Akhirnya, program mencetak urutan string yang sudah dirotasi. Dengan input tiga string dari pengguna, lalu proses menukar urutan string, dan output urutan string sebelum dan sesudah ditukar.

b. Soal Studi Case

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

Sourcecode

```
package main
```

```

import "fmt"

func main() {

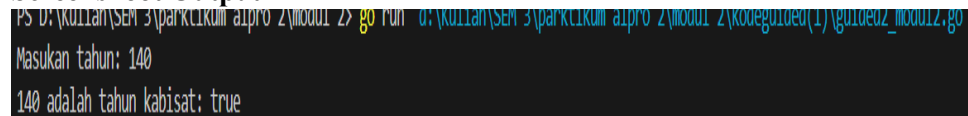
    var tahun int

    fmt.Print("Masukan tahun: ")
    fmt.Scanln(&tahun)

    if (tahun%400 == 0) || (tahun%4 == 0 &&
tahun%100 != 0) {
        fmt.Println(tahun, "adalah tahun kabisat:
true")
    } else {
        fmt.Println(tahun, "adalah tahun kabisat:
false")
    }
}

```

Screenshoot Output



```

PS D:\KULIAH\SEM 3\parker\alpro 2\modul 2\kodeguide(1)\guide02_modul2.go
Masukan tahun: 140
140 adalah tahun kabisat: true

```

Deskripsi Program

Program ini berfungsi untuk menentukan apakah suatu tahun adalah tahun kabisat. Program dimulai dengan meminta pengguna memasukkan sebuah angka tahun. Setelah itu, program memeriksa apakah tahun tersebut memenuhi syarat tahun kabisat. Syaratnya adalah tahun harus habis dibagi 400, atau habis dibagi 4 tetapi tidak habis dibagi 100. Jika salah satu dari syarat tersebut terpenuhi, program akan mencetak bahwa tahun tersebut adalah tahun kabisat (true). Jika tidak, program akan mencetak bahwa tahun tersebut bukan tahun kabisat (false). Misalnya, jika pengguna memasukkan tahun 2024, program akan menampilkan bahwa 2024 adalah tahun kabisat: true.

c. Soal Studi Case

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):

Jejari = 5

Bola dengan jejari 5 memiliki volume 523.5988 dan luas kulit 314.1593

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var jariJari float64
    fmt.Print("Masukan jari-jari bola: ")
    fmt.Scanln(&jariJari)

    volume := (4.0 / 3.0) * math.Pi *
    math.Pow(jariJari, 3)
    luasPermukaan := 4 * math.Pi *
    math.Pow(jariJari, 2)

    fmt.Printf("Volume bola dengan jari-jari %.2f
    adalah %.2f\n", jariJari, volume)
    fmt.Printf("Luas permukaan bola dengan jari-
    jari %.2f adalah %.2f\n", jariJari, luasPermukaan)
}
```

Screenshoot Output

```
PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run "d:\Kuliah\SEM 3\parktikum alpro 2\modul 2\kodeguided(2)\guided3.modul2.go"
Masukan jari-jari bola: 32
Volume bola dengan jari-jari 32.00 adalah 137258.28
Luas permukaan bola dengan jari-jari 32 adalah 12867.96
```

Deskripsi Program

Program ini menghitung volume dan luas permukaan bola berdasarkan jari-jari yang dimasukkan oleh pengguna. Setelah pengguna memasukkan nilai jari-jari, program menggunakan rumus $\frac{4}{3} \times \pi \times r^3$ untuk menghitung volume bola, dan rumus $4 \times \pi \times r^2$ untuk menghitung luas permukaan bola, di mana r adalah jari-

jari. Hasil perhitungan volume dan luas permukaan bola kemudian ditampilkan dengan format dua angka desimal. Sebagai contoh, jika pengguna memasukkan jari-jari 5, program akan menghitung dan menampilkan volume dan luas permukaan bola tersebut.

III. UNGUIDED

1. Berisi source code dan output dari kegiatan praktikum yang telah dilaksanakan. Source Code diberi penjelasan maka akan menjadi nilai ++

2a. 1. Soal Studi Case

4. Dibaca nilai temperatur dalam derajat Celsius. Nyatakan temperatur tersebut dalam Fahrenheit

$$Celsius = (Fahrenheit - 32) \times \frac{5}{9} \quad Reamur = Celsius \times \frac{4}{5} \quad Kelvin = (Fahrenheit + 459.67) \times \frac{5}{9}$$

(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
```

Sourcecode

```
package main

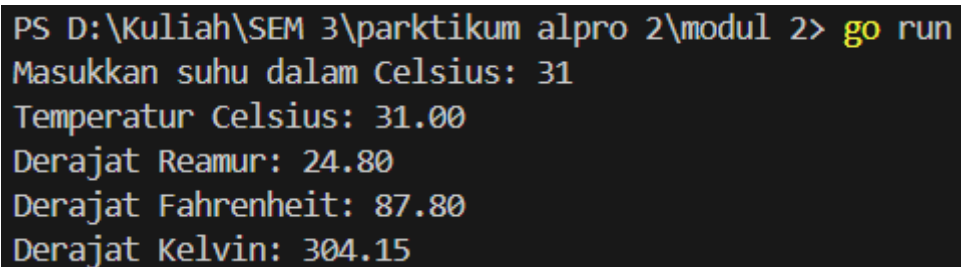
import (
    "fmt"
)

func main() {
    var celsius float64
    fmt.Print("Masukkan suhu dalam Celsius: ")
    fmt.Scanln(&celsius)

    reamur := celsius * 4 / 5
    fahrenheit := (celsius * 9 / 5) + 32
    kelvin := celsius + 273.15
```

```
        fmt.Printf("Temperatur Celsius: %.2f\n",  
celsius)  
        fmt.Printf("Derajat Reamur: %.2f\n", reamur)  
        fmt.Printf("Derajat Fahrenheit: %.2f\n",  
fahrenheit)  
        fmt.Printf("Derajat Kelvin: %.2f\n", kelvin)  
    }
```

Screenshoot Output



```
PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run  
Masukkan suhu dalam Celsius: 31  
Temperatur Celsius: 31.00  
Derajat Reamur: 24.80  
Derajat Fahrenheit: 87.80  
Derajat Kelvin: 304.15
```

Deskripsi Program

Program ini digunakan untuk mengonversi suhu dari derajat Celsius ke beberapa satuan suhu lainnya, yaitu Reamur, Fahrenheit, dan Kelvin. Setelah pengguna memasukkan suhu dalam derajat Celsius, program menghitung nilai suhu dalam satuan Reamur, Fahrenheit, dan Kelvin menggunakan rumus konversi yang sesuai: suhu Reamur dihitung dengan mengalikan suhu Celsius dengan $\frac{4}{5}$, suhu Fahrenheit dihitung dengan rumus $\frac{9}{5} \times \text{Celsius} + 32$, dan suhu Kelvin dihitung dengan menambahkan 273,15 ke suhu Celsius. Hasil konversi ini kemudian ditampilkan dengan format dua angka desimal. Misalnya, jika pengguna memasukkan suhu 25 derajat Celsius, program akan menampilkan hasil konversi suhu tersebut dalam Reamur, Fahrenheit, dan Kelvin.

2a. 2. Soal Studi Case

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

Catatan: Gunakan `fmt.Scanf("%c", &var)` untuk pembacaan satu karakter dan `fmt.Printf("%c", var)` untuk penulisan satu karakter.

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var a, b, c, d, e int
    var char1, char2, char3 rune

    fmt.Println("Masukkan 5 buah integer (spasi dipisahkan) (Antara 32 s.d. 127): ")
    fmt.Scanf("%d %d %d %d %d", &a, &b, &c, &d, &e)

    fmt.Println("Masukkan 3 buah karakter (tanpa spasi): ")
    fmt.Scanf("%c%c%c", &char1, &char2, &char3)

    fmt.Printf("%c%c%c%c%c\n", a, b, c, d, e)

    fmt.Printf("%c%c%c\n", char1, char2, char3)
}
```



Screenshoot Output

```
PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run "d:\Kuliah\9
Masukkan 5 buah integer (spasi dipisahkan)(Antara 32 s.d. 127):
35 44 64 51 111
Masukkan 3 buah karakter (tanpa spasi):
WUI
#,@3o
WU
```

Deskripsi Program

Program ini meminta pengguna untuk memasukkan lima buah integer dan tiga karakter, kemudian menampilkan karakter-karakter yang sesuai dengan kode ASCII dari integer yang dimasukkan serta karakter yang diinput. Pertama, pengguna diminta untuk memasukkan lima integer (antara 32 hingga 127) yang dipisahkan oleh spasi, dan program menggunakan `fmt.Scanf` untuk membaca input ini ke dalam variabel `a`, `b`, `c`, `d`, dan `e`. Selanjutnya, pengguna diminta memasukkan tiga karakter tanpa spasi, yang disimpan dalam variabel `char1`, `char2`, dan `char3`. Program kemudian menampilkan karakter yang sesuai dengan nilai integer tersebut menggunakan `fmt.Printf` untuk mengonversi integer menjadi karakter ASCII, serta menampilkan tiga karakter yang telah diinput oleh pengguna.

2b. 1. Soal Studi Case

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				

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var warna [5][4]string
    urutanBenar := [4]string{"merah", "kuning",
    "hijau", "ungu"}
    var berhasil bool = true

    for i := range warna {
        fmt.Printf("Masukkan warna untuk percobaan
        %d (pisahkan dengan spasi): ", i+1)
        fmt.Scan(&warna[i][0], &warna[i][1],
        &warna[i][2], &warna[i][3])
    }

    for i, percobaan := range warna {
```

```

        fmt.Printf("Percobaan %d: %s %s %s %s\n",
i+1, percobaan[0], percobaan[1], percobaan[2],
percobaan[3])
        if percobaan != urutanBenar {
            berhasil = false
        }
    }

    fmt.Printf("Berhasil: %t\n", berhasil)
}

```

Screenshoot Output

```

PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run "d:\Kuliah\SEM 3\parktikum al
Masukkan warna untuk percobaan 1 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 2 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 3 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 4 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 5 (pisahkan dengan spasi): merah kuning hijau ungu
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

Masukkan warna untuk percobaan 1 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 2 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 3 (pisahkan dengan spasi): merah kuning hijau biru
Masukkan warna untuk percobaan 4 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 5 (pisahkan dengan spasi): merah kuning hijau ungu
Percobaan 1: merah kuning hijau ungu
Percobaan 2: merah kuning hijau ungu
Percobaan 3: merah kuning hijau biru
Percobaan 4: merah kuning hijau ungu
Percobaan 5: merah kuning hijau ungu
Berhasil: false

```

Deskripsi Program

Program ini meminta pengguna untuk memasukkan urutan warna dalam beberapa percobaan dan membandingkannya dengan urutan warna yang benar. Pertama, program mendeklarasikan array dua dimensi warna untuk menyimpan input warna dari pengguna, serta array urutanBenar yang berisi warna yang dianggap benar, yaitu "merah", "kuning", "hijau", dan "ungu". Variabel boolean berhasil diinisialisasi dengan nilai true. Selanjutnya, program meminta pengguna memasukkan empat warna untuk masing-masing dari lima percobaan dan menyimpannya dalam array warna. Setelah mendapatkan input, program mencetak hasil dari setiap percobaan dan membandingkannya dengan urutan warna yang

benar. Jika ada percobaan yang tidak sesuai, variabel berhasil diubah menjadi false. Akhirnya, program mencetak apakah semua percobaan berhasil (true) atau tidak (false) berdasarkan kecocokan dengan urutan yang benar. Sebagai contoh, jika pengguna memasukkan warna yang benar di semua percobaan, output akhir akan menunjukkan Berhasil: true, sementara jika ada yang salah, output akan menunjukkan Berhasil: false.

2b. 2. Soal Studi Case

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

in 23 | Modul Praktikum Algoritma dan Pemrograman 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.

(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 : 0
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	

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    ribbon := ""
```

```

    flowerCount := 0

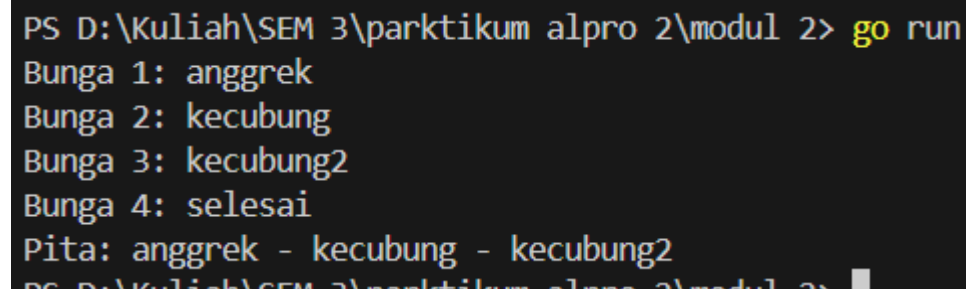
    for {
        flowerCount++
        var flower string
        fmt.Printf("Bunga %d: ", flowerCount)
        fmt.Scan(&flower)

        if flower == "selesai" {
            break
        }

        if flowerCount == 1 {
            ribbon = flower
        } else {
            ribbon = ribbon + " - " + flower
        }
    }
    fmt.Println("Pita:", ribbon)
}

```

Screenshoot Output



```

PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run
Bunga 1: anggrek
Bunga 2: kecubung
Bunga 3: kecubung2
Bunga 4: selesai
Pita: anggrek - kecubung - kecubung2
PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2>

```

Deskripsi Program

Program ini digunakan untuk mengumpulkan nama bunga dari pengguna dan menyusunnya menjadi satu string yang dipisahkan oleh tanda " - ". Program dimulai dengan mendeklarasikan variabel ribbon untuk menyimpan nama bunga dan flowerCount untuk menghitung jumlah bunga yang dimasukkan. Dalam loop tak terbatas, pengguna diminta memasukkan nama bunga; jika memasukkan "selesai", loop berhenti. Nama bunga pertama disimpan langsung di ribbon, sedangkan bunga berikutnya ditambahkan dengan pemisah " - ". Setelah pengguna selesai, program mencetak daftar bunga yang telah dimasukkan.

2b. 3. Soal Studi Case

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: 5.5 1.0
Masukan berat belanjaan di kedua kantong: 7.1 8.5
Masukan berat belanjaan di kedua kantong: 2 6
Masukan berat belanjaan di kedua kantong: 9 5.8
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.
```

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var beratKiri, beratKanan float64
    totalBerat := 0.0
    prosesSelesai := false

    for {
        fmt.Print("Masukkan berat belanjaan di
kedua kantong: ")
        fmt.Scanln(&beratKiri, &beratKanan)
```

```

        fmt.Printf("Berat di kedua kantong adalah:
%.1f kg dan %.1f kg\n", beratKiri, beratKanan)

        totalBerat += beratKiri + beratKanan

        if beratKiri >= 9 || beratKanan >= 9 {
            prosesSelesai = true
        }

        selisih := math.Abs(beratKiri - beratKanan)

        if selisih >= 9 {
            fmt.Println("Sepeda motor Pak Andi akan
oleng: true")
        } else {
            fmt.Println("Sepeda motor Pak Andi akan
oleng: false")
        }
        if totalBerat > 150 {
            prosesSelesai = true
        }
        if prosesSelesai {
            fmt.Println("Proses selesai.")
            break
        }
    }
}

```

Screenshoot Output

```

PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run
Masukkan berat belanjaan di kedua kantong: 2.9 5.3
Berat di kedua kantong adalah: 2.9 kg dan 5.3 kg
Sepeda motor Pak Andi akan oleng: false
Masukkan berat belanjaan di kedua kantong: 55.8 88.3
Berat di kedua kantong adalah: 55.8 kg dan 88.3 kg
Sepeda motor Pak Andi akan oleng: true

```

Deskripsi Program

Program ini memantau berat belanjaan di dua kantong dan menentukan apakah sepeda motor akan oleng. Pengguna diminta untuk memasukkan berat di kedua kantong secara berulang. Setelah input, program mencetak berat tersebut dan menambahkan total berat ke variabel totalBerat. Jika salah satu berat mencapai atau

melebihi 9 kg, atau total berat melebihi 150 kg, proses akan dihentikan. Program juga memeriksa selisih antara berat kedua kantong; jika selisihnya 9 kg atau lebih, program mencetak bahwa sepeda motor Pak Andi akan oleng. Proses akan berlanjut hingga salah satu kondisi terpenuhi, setelah itu program mencetak "Proses selesai."

4o mini

2b. 4. Soal Studi Case

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 dihampiri dengan rumus berikut:

ian 25 | Modul Praktikum Algoritma dan Pemrograman 2

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

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 = <u>10</u> Nilai akar 2 = 1.4062050441
2	Nilai K = <u>100</u> Nilai akar 2 = 1.4133387072
3	Nilai K = <u>1000</u> Nilai akar 2 = 1.4141252651

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func fK(K float64) float64 {
    numerator := (4*K + 2) * (4*K + 2)
    denominator := (4*K + 1) * (4*K + 3)
```

```

        return numerator / denominator
    }

    func sqrt2_approx(K int) float64 {
        result := 1.0
        for i := 0; i <= K; i++ {
            numerator := (4*float64(i) + 2) *
(4*float64(i) + 2)
            denominator := (4*float64(i) + 1) *
(4*float64(i) + 3)
            result *= numerator / denominator
        }
        return result
    }

    func main() {
        var K float64
        fmt.Print("Masukan nilai K: ")
        fmt.Scan(&K)

        resultF := fK(K)
        fmt.Printf("Nilai f(K) = %.10f\n", resultF)

        Kint := int(math.Round(K))
        resultSqrt2 := sqrt2_approx(Kint)
        fmt.Printf("Nilai akar 2 = %.10f\n",
resultSqrt2)
    }

```

Screenshoot Output

```

PS D:\Kuliah\SEM 3\parktikum alpro 2\modul 2> go run
Masukan nilai K: 100
Nilai f(K) = 1.0000061880
Nilai akar 2 = 1.4133387072

```

Deskripsi Program

Program ini menghitung nilai fungsi $f(K)$ dan memberikan perkiraan akar dari 2 menggunakan metode yang melibatkan faktor K . Pertama, terdapat fungsi $fK(K \text{ float64})$ yang menghitung nilai $f(K)$ dengan rumus

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

Selanjutnya, fungsi $\text{sqrt2_approx}(K \text{ int})$ digunakan untuk memperkirakan akar dari 2 dengan menghitung produk dari serangkaian pecahan yang dibentuk berdasarkan rumus yang sama, diulang dari 0 hingga K . Dalam

fungsi `main()`, program meminta pengguna memasukkan nilai K , kemudian menghitung dan mencetak hasil $f(K)$. Setelah itu, nilai K dibulatkan menjadi integer dan digunakan untuk memperkirakan akar 2 dengan memanggil `sqrt2_approx`, yang hasilnya juga dicetak. Sebagai contoh, jika pengguna memasukkan nilai $K = 2$, program akan menampilkan nilai dari $f(2)$ dan perkiraan akar 2 berdasarkan metode yang