

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II
REVIEW STRUKTUR KONTROL**



Disusun Oleh :

Muhammad Ihab Aufa Rafi / 2311102226

S1IF-11-06

Dosen Pengampu :

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

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Bahasa Go, atau dikenal juga sebagai Golang, merupakan bahasa pemrograman yang dikembangkan oleh Google dengan tujuan untuk menyederhanakan pengembangan aplikasi yang efisien, cepat dan juga diandalkan. Go memiliki keunggulan dalam hal efisiensi memori dan kinerja, serta dukungan paralelisme yang kuat. Dibandingkan dengan bahasa pemrograman lain seperti Python atau Java, Go menawarkan eksekusi yang lebih cepat karena Go merupakan bahasa yang dikompilasi langsung ke kode mesin. Go juga mendukung concurrency secara native, memungkinkan pembuatan aplikasi yang memanfaatkan prosesor multi-core secara optimal.

Beberapa bahasa pemrograman dirancang untuk diimplementasikan sebagai interpreter dan lainnya sebagai kompilator. Interpreter akan membaca setiap baris instruksi dan kemudian langsung mengeksekusinya, dengan hanya sedikit pemeriksaan apakah penulisan keseluruhan program sudah benar atau belum. Kompilator akan memeriksa keseluruhan program sumber dan kemudian mengubahnya menjadi program eksekutabel, sehingga konsistensi penulisan 100 (seperti penggunaan tipe data) sudah diperiksa sebelum eksekusi. Selain itu karena program dibuat menjadi eksekutabel lebih dahulu, proses optimasi dapat dilakukan sehingga program menjadi sangat efisien.

Go diimplementasikan sebagai kompilator Berikut adalah contoh sesi yang biasa dilakukan saat mengkompilasi dan mengeksekusi program dalam bahasa Go:

- Panggil shell atau terminal (program/utiliti cmd.exe di Windows).
- Masuk ke dalam (cd) folder program (normalnya ada di C:\Users\go\src\ atau yang sejenis).
- Kemudian panggil perintah go build atau go build file.go untuk mengkompilasi file.go.
- Jika gagal, akan muncul pesan eror yang sesuai, pelajari dengan baik pesan tersebut, perbaiki teks program sumber, kemudian ulangi proses build-nya.
- Jika berhasil maka pada folder tersebut akan dibuat program dengan nama yang sama dan diakhiri dengan.exe (untuk Windows).

- Panggil program eksekutabel tersebut dari terminal yang sama. Jangan memanggil program tersebut dengan mengklik eksekutabel tersebut dari folder karena program kalian hanya berbasis teks, bukan/belum dirancang dengan tampilan Windows.

II. GUIDED

1. Soal Studi Case

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?

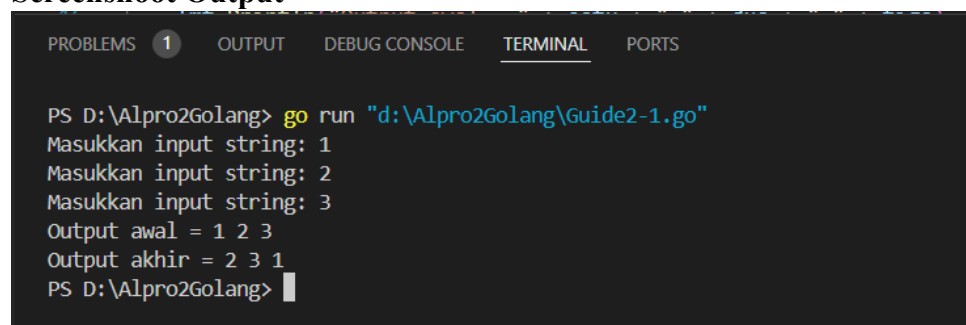
Sourcecode

```
package main

import "fmt"

func main(){
    var(
        satu, dua, tiga string
        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)
}
```

Screenshoot Output



```
PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Guide2-1.go"
Masukkan input string: 1
Masukkan input string: 2
Masukkan input string: 3
Output awal = 1 2 3
Output akhir = 2 3 1
PS D:\Alpro2Golang>
```

Deskripsi Program

Program ini meminta user untuk menginputkan 3(tiga) data string secara berurutan yang disimpan ke dalam variabel **satu**, **dua**, dan **tiga**. Setelah itu program akan mencetak output dari ketiga data string tersebut. Berikutnya, program menukar posisi data pertama ke posisi data ketiga menggunakan variabel **temp**.

2. Soal Studi Case

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**).

Sourcecode

```
package main

import "fmt"
func isKabisat(year int) bool {
    if year%400 == 0 {
        return true
    }
    if year%4 == 0 && year%100 != 0 {
        return true
    }

    return false
}

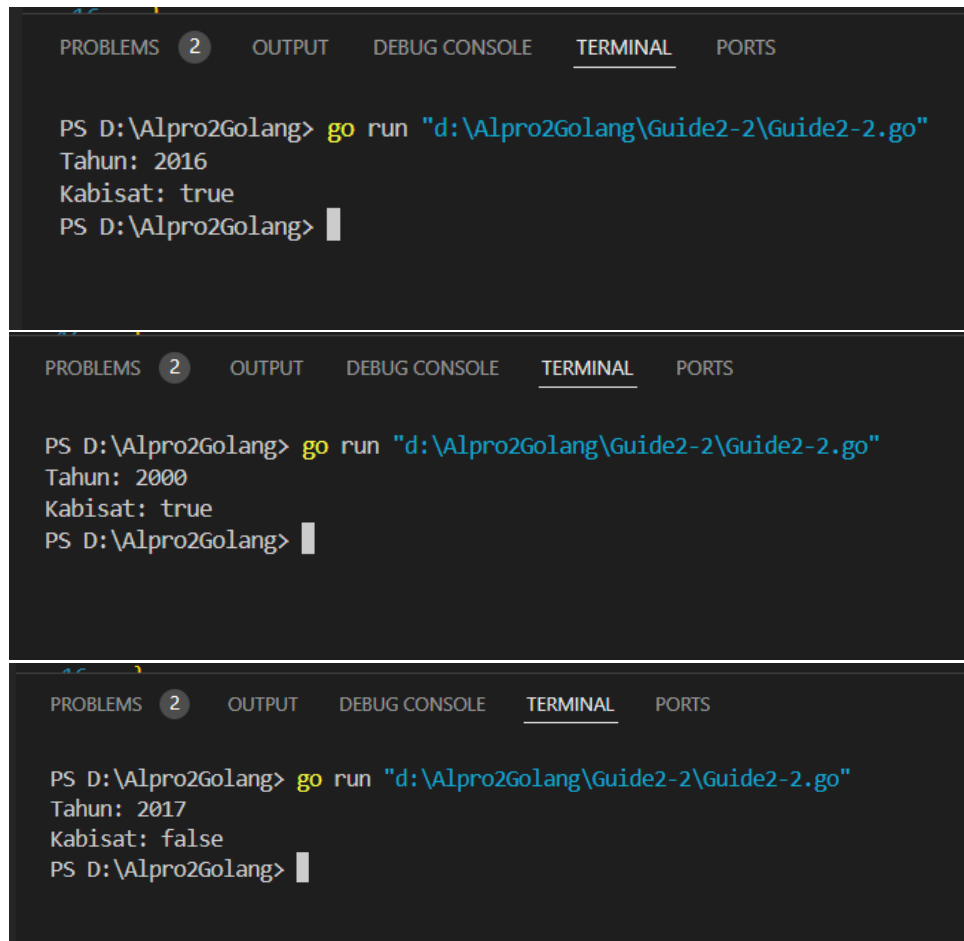
func main() {
    var tahun int

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

    kabisat := isKabisat(tahun)

    fmt.Printf("Kabisat: %t\n", kabisat)
}
```

Screenshoot Output



The image shows three sequential screenshots of a terminal window in an IDE, demonstrating the execution of a Go program. Each screenshot has a tab bar at the top with 'PROBLEMS' (containing a '2'), 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (selected), and 'PORTS'. The terminal shows the command `go run "d:\Alpro2Golang\Guide2-2\Guide2-2.go"` being executed. The program outputs the year, whether it is a leap year (`Kabisat: true` or `false`), and returns to the prompt.

```
PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Guide2-2\Guide2-2.go"
Tahun: 2016
Kabisat: true
PS D:\Alpro2Golang>
```

```
PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Guide2-2\Guide2-2.go"
Tahun: 2000
Kabisat: true
PS D:\Alpro2Golang>
```

```
PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Guide2-2\Guide2-2.go"
Tahun: 2017
Kabisat: false
PS D:\Alpro2Golang>
```

Deskripsi Program

Program ini merupakan program sederhana yang ditulis dalam bahasa Go (Golang) untuk menentukan apakah sebuah tahun merupakan tahun kabisat atau tidak. Fungsi `isKabisat` menerima parameter `year` bertipe `int` dan mengembalikan nilai `bool`. Fungsi ini menggunakan aturan untuk menentukan apakah sebuah tahun adalah tahun kabisat:

- Jika tahun habis dibagi 400, maka itu adalah tahun kabisat.
- Jika tahun habis dibagi 4 tetapi tidak habis dibagi 100, maka itu adalah tahun kabisat.
- Jika tidak memenuhi kedua kondisi di atas, maka itu bukan tahun kabisat.

3. Soal Studi Case

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 $luasbola = 4\pi r^2$ ($\pi \approx 3.1415926535$).

Sourcecode

```
package main

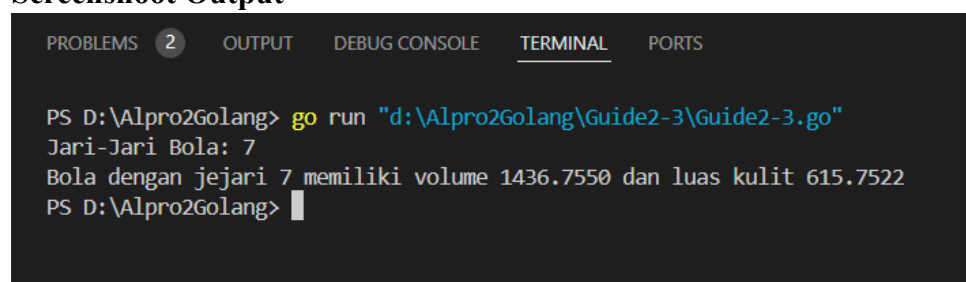
import (
    "fmt"
    "math"
)

func main() {
    var jejariBola float64
    fmt.Print("Jari-Jari Bola: ")
    fmt.Scanln(&jejariBola)

    volumeBola := (4.0/3.0) * math.Pi * math.Pow(jejariBola,
3)
    luasBola := 4 * math.Pi * math.Pow(jejariBola, 2)

    fmt.Printf("Bola dengan jejari %.0f memiliki volume %.4f
dan luas kulit %.4f\n", jejariBola, volumeBola, luasBola)
}
```

Screenshoot Output



```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Guide2-3\Guide2-3.go"
Jari-Jari Bola: 7
Bola dengan jejari 7 memiliki volume 1436.7550 dan luas kulit 615.7522
PS D:\Alpro2Golang>
```

Deskripsi Program

Program di atas adalah sebuah program sederhana yang ditulis dalam bahasa Go untuk menghitung volume dan luas kulit bola berdasarkan jari-jari bola yang diberikan oleh user. Program menampilkan hasilnya ke console dengan format:

- Jari-jari bola dengan nilai yang diinput oleh user.

- Volume bola dengan 4 digit desimal.
- Luas kulit bola dengan 4 digit desimal.

III. UNGUIDED

1. Soal Studi Case

Dibaca nilai temperatur dalam derajat Celsius. Nyatakan temperatur tersebut dalam Fahrenheit. Lanjutkan program di atas, sehingga temperatur dinyatakan juga dalam derajat Reamur dan Kelvin.

Sourcecode

```
package main

import "fmt"

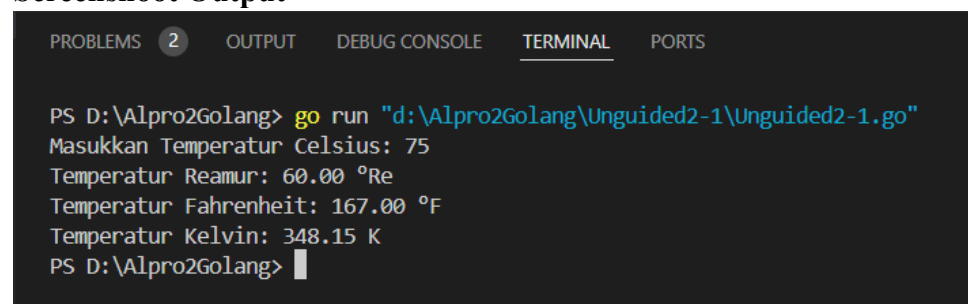
func main() {

    var C_226 float64
    fmt.Print("Masukkan Temperatur Celsius: ")
    fmt.Scanln(&C_226)

    F_226 := (C_226 * 9/5) + 32
    R_226 := C_226 * 4/5
    K_226 := C_226 + 273.15

    fmt.Printf("Temperatur Reamur: %.2f °Re\n", R_226)
    fmt.Printf("Temperatur Fahrenheit: %.2f °F\n", F_226)
    fmt.Printf("Temperatur Kelvin: %.2f K\n", K_226)
}
```

Screenshoot Output



The screenshot shows a terminal window with the following content:

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Unguided2-1\Unguided2-1.go"
Masukkan Temperatur Celsius: 75
Temperatur Reamur: 60.00 °Re
Temperatur Fahrenheit: 167.00 °F
Temperatur Kelvin: 348.15 K
PS D:\Alpro2Golang> |
```

Deskripsi Program

Program di atas merupakan program untuk mengonversi suhu dari Celsius ke beberapa skala suhu lainnya, yaitu Reamur, Fahrenheit, dan Kelvin berdasarkan rumus-rumus yang telah diketahui sebelumnya.

2. Soal Studi Case

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. Masukan data integer mempunyai nilai antara 32 s.d. 127. Masukan 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 setelah karakter yang diinputkan (tanpa dipisahkan oleh spasi).

Sourcecode

```
package main

import "fmt"

func main() {

    var angka1_226, angka2_226, angka3_226, angka4_226,
    angka5_226 int
    fmt.Println("Masukkan 5 angka integer (32 - 127)
    dipisahkan spasi:")
    fmt.Scan(&angka1_226, &angka2_226, &angka3_226,
    &angka4_226, &angka5_226)

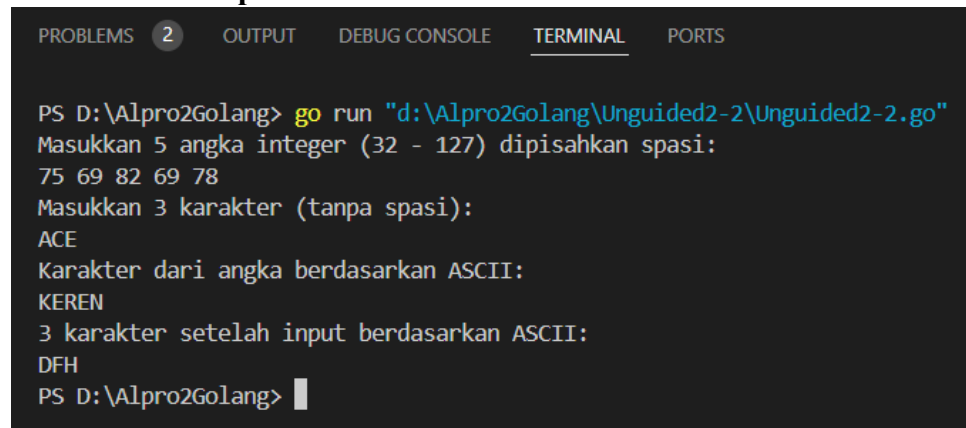
    var char_226 string
    fmt.Println("Masukkan 3 karakter (tanpa spasi):")
    fmt.Scan(&char_226)

    fmt.Println("Karakter dari angka berdasarkan ASCII:")
    fmt.Printf("%c%c%c%c%c\n", angka1_226, angka2_226,
    angka3_226, angka4_226, angka5_226)

    fmt.Println("3 karakter setelah input berdasarkan
    ASCII:")
    for _, char_226 := range char_226 {
        fmt.Printf("%c", char_226+3)
    }
}
```

```
}  
    fmt.Println()  
}
```

Screenshoot Output



The screenshot shows a terminal window with the following content:

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS  
  
PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Unguided2-2\Unguided2-2.go"  
Masukkan 5 angka integer (32 - 127) dipisahkan spasi:  
75 69 82 69 78  
Masukkan 3 karakter (tanpa spasi):  
ACE  
Karakter dari angka berdasarkan ASCII:  
KEREN  
3 karakter setelah input berdasarkan ASCII:  
DFH  
PS D:\Alpro2Golang> |
```

Deskripsi Program

Program ini adalah program sederhana dalam bahasa Go yang meminta user untuk memasukkan beberapa angka dan karakter, kemudian mengubahnya menjadi karakter berdasarkan nilai ASCII dan juga memodifikasi karakter input dengan menambahkannya tiga posisi ke depan dalam tabel ASCII. Pertama, program meminta user untuk memasukkan lima angka integer. Angka-angka ini akan disimpan dalam lima variabel berbeda, yaitu `angka1_226`, `angka2_226`, `angka3_226`, `angka4_226`, dan `angka5_226`. Setelah itu, program meminta user memasukkan tiga karakter tanpa spasi, yang disimpan dalam variabel `char_226`. Setelah menerima input, program akan mengonversi setiap angka yang dimasukkan ke karakter ASCII yang sesuai. Ini dilakukan dengan menggunakan format `%c` dalam `fmt.Printf`, yang menampilkan karakter berdasarkan nilai ASCII yang diberikan. Misalnya, jika angka yang dimasukkan adalah 65, maka karakter yang dihasilkan adalah 'A', karena 65 adalah kode ASCII untuk huruf 'A'.

3. Soal Studi Case

Dibaca 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 warna sesuai sesuai dengan d informasi yang diberikan pada paragraf sebelumnya, dan **false** jika urutan warnanya salah.

Sourcecode

```
package main

import (
    "fmt"
    "strings"
)

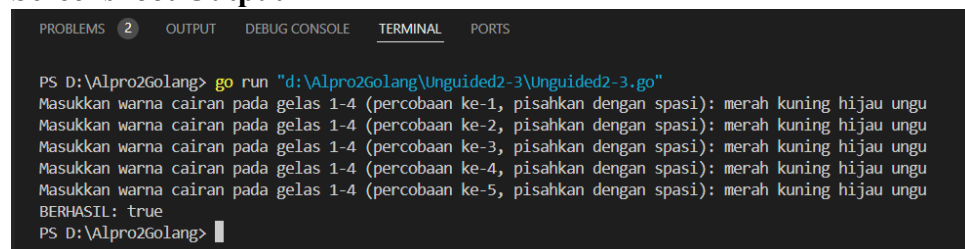
func main() {
    var warna1_226, warna2_226, warna3_226, warna4_226
    string
    var berhasil bool = true

    for i := 1; i <= 5; i++ {
        fmt.Printf("Masukkan warna cairan pada gelas 1-4 (percobaan ke-%d, pisahkan dengan spasi): ", i)
        fmt.Scanln(&warna1_226, &warna2_226, &warna3_226, &warna4_226)

        if !strings.EqualFold(warna1_226, "merah") ||
            !strings.EqualFold(warna2_226, "kuning") ||
            !strings.EqualFold(warna3_226, "hijau") ||
            !strings.EqualFold(warna4_226, "ungu") {
            berhasil = false
            break
        }
    }

    fmt.Println("BERHASIL:", berhasil)
}
```

Screenshoot Output



```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Unguided2-3\Unguided2-3.go"
Masukkan warna cairan pada gelas 1-4 (percobaan ke-1, pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna cairan pada gelas 1-4 (percobaan ke-2, pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna cairan pada gelas 1-4 (percobaan ke-3, pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna cairan pada gelas 1-4 (percobaan ke-4, pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna cairan pada gelas 1-4 (percobaan ke-5, pisahkan dengan spasi): merah kuning hijau ungu
BERHASIL: true
PS D:\Alpro2Golang> |
```

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Unguided2-3\Unguided2-3.go"
Masukkan warna cairan pada gelas 1-4 (percobaan ke-1, pisahkan dengan spasi): merah hijau kuning ungu
BERHASIL: false
PS D:\Alpro2Golang> |
```

Deskripsi Program

Program di atas merupakan sebuah aplikasi sederhana dalam bahasa Go yang meminta user untuk memasukkan urutan warna cairan di empat gelas sebanyak lima kali, kemudian memeriksa apakah urutan warna yang dimasukkan selalu sesuai dengan urutan yang diinginkan. Urutan warna yang benar adalah "merah", "kuning", "hijau", dan "ungu". Jika pada salah satu percobaan user memasukkan warna yang tidak sesuai, program akan menghentikan proses dan menganggap percobaan gagal.

4. Soal Studi Case

Suatu pita (string) berisi kumpulan nama-nama bunga yang dipisahkan oleh spasi dan '-', contoh pita diilustrasikan seperti berikut ini. Pita: mawar - melati - tulip - teratai - kamboja - anggrek.

Buatlah sebuah program yang menerima input sebuah bilangan bulat positif (dan tidak nol) N, kemudian program akan meminta input berupa nama bunga secara berulang sebanyak N kali dan nama tersebut disimpan ke dalam pita.

(Petunjuk: gunakan operasi penggabungan string dengan operator "+").
Tampilkan isi pita setelah proses input selesai.

Modifikasi program sebelumnya, proses input akan berhenti apabila user mengetikkan 'SELESAI'. Kemudian tampilkan isi pita beserta banyaknya bunga yang ada di dalam pita.

Sourcecode

```
package main

import (
    "fmt"
    "strings"
)

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

```

hitungBunga_226 := 0

for {
    var bunga_226 string
    fmt.Printf("Bunga %d: ", hitungBunga_226+1)
    fmt.Scan(&bunga_226)

    if strings.ToUpper(bunga_226) == "SELESAI" {
        break
    }

    if pita == "" {
        pita = bunga_226
    } else {
        pita += " - " + bunga_226
    }

    hitungBunga_226++
}

fmt.Println("Pita:", pita)
fmt.Println("Bunga:", hitungBunga_226)
}

```

Screenshoot Output

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Unguided2-4\Unguided2-4.go"
Bunga 1: Krisan
Bunga 2: Melati
Bunga 3: Kamboja
Bunga 4: Terompet
Bunga 5: SELESAI
Pita: Krisan - Melati - Kamboja - Terompet
Bunga: 4
PS D:\Alpro2Golang>

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Unguided2-4\Unguided2-4.go"
Bunga 1: SELESAI
Pita:
Bunga: 0
PS D:\Alpro2Golang>

```

Deskripsi Program

Program di atas merupakan program yang meminta user untuk memasukkan nama-nama bunga satu per satu. User akan terus memasukkan nama bunga sampai mereka mengetikkan kata "SELESAI", yang mengakhiri input. Setelah itu, program menampilkan daftar bunga yang sudah dimasukkan serta jumlah bunga yang tercatat. program mendeklarasikan dua variabel, `pita` dan `hitungBunga_226`. Variabel `pita` berfungsi untuk menyimpan daftar bunga yang dimasukkan, dan `hitungBunga_226` digunakan untuk menghitung jumlah bunga yang sudah dimasukkan.

5. Soal Studi Case

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

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

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.

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var beratKiri_226, beratKanan_226 float64

    for {
        fmt.Print("Masukan berat belanjaan di kedua kantong: ")
        fmt.Scan(&beratKiri_226, &beratKanan_226)

        totalBerat := beratKiri_226 + beratKanan_226
```

```

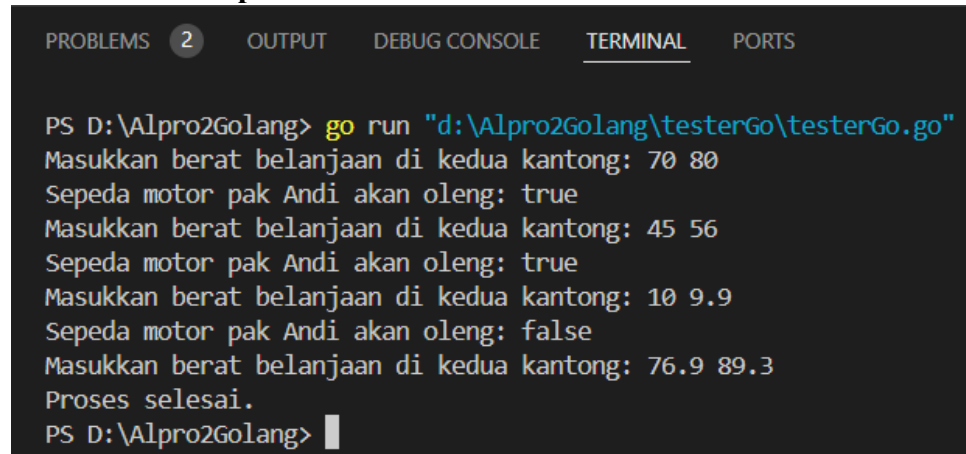
        if totalBerat > 150 {
            fmt.Println("Proses selesai.")
            break
        }

        selisih := math.Abs(beratKiri_226 - beratKanan_226)
        oleng := selisih >= 9

        fmt.Printf("Sepeda motor pak Andi akan oleng: %t\n",
oleng)
    }
}

```

Screenshoot Output



```

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\testerGo\testerGo.go"
Masukkan berat belanjaan di kedua kantong: 70 80
Sepeda motor pak Andi akan oleng: true
Masukkan berat belanjaan di kedua kantong: 45 56
Sepeda motor pak Andi akan oleng: true
Masukkan berat belanjaan di kedua kantong: 10 9.9
Sepeda motor pak Andi akan oleng: false
Masukkan berat belanjaan di kedua kantong: 76.9 89.3
Proses selesai.
PS D:\Alpro2Golang>

```

Deskripsi Program

Program ini merupakan program sederhana yang digunakan untuk mengevaluasi keseimbangan beban di dua kantong belanja yang dibawa oleh Pak Andi pada sepeda motornya. Program akan terus meminta input dari pengguna mengenai berat belanjaan di dua kantong, kemudian menentukan apakah sepeda motor Pak Andi akan oleng (tidak seimbang) berdasarkan perbedaan berat antara kedua kantong. Program ini juga menghentikan proses ketika total berat dari kedua kantong melebihi 150 kg.

6. Soal Studi Case

Diberikan sebuah persamaan berikut ini.

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

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

$\sqrt{2}$ merupakan bilangan irasional. Meskipun demikian, nilai tersebut dapat dihampiri 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.

Sourcecode

```
package main

import (
    "fmt"
)

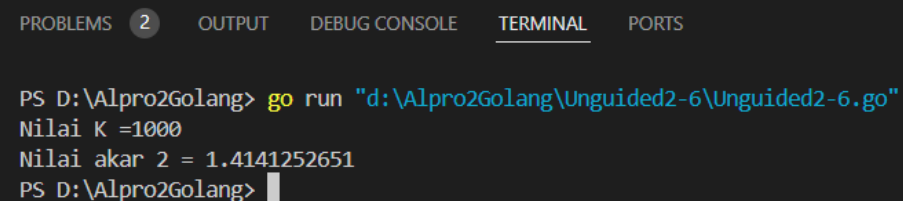
func main() {
    var K_226 int
    var sqrt2 float64 = 1.0

    fmt.Print("Nilai K = ")
    fmt.Scan(&K_226)

    for i := 0; i <= K_226; i++ {
        atas := (4*float64(i) + 2) * (4*float64(i) + 2)
        bawah := (4*float64(i) + 1) * (4*float64(i) + 3)
        sqrt2 *= atas / bawah
    }

    fmt.Printf("Nilai akar 2 = %.10f\n", sqrt2)
}
```

Screenshoot Output



The screenshot shows a terminal window with the following content:

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Alpro2Golang> go run "d:\Alpro2Golang\Unguided2-6\Unguided2-6.go"
Nilai K =1000
Nilai akar 2 = 1.4141252651
PS D:\Alpro2Golang> 
```

Deskripsi Program

Program ini merupakan sebuah aplikasi dalam bahasa Go yang digunakan untuk menghitung nilai perkiraan akar kuadrat dari 2 menggunakan deret tak hingga berdasarkan algoritma Wallis product.

Program dimulai dengan mendeklarasikan dua variabel, yaitu **K_226** dan **sqrt2**. Variabel **K_226** adalah integer yang akan diinput oleh pengguna, yang menentukan berapa banyak iterasi yang akan dilakukan untuk menghitung akar kuadrat. Variabel **sqrt2** dideklarasikan sebagai float64 dan diinisialisasi dengan nilai 1.0. Nilai ini akan digunakan untuk menyimpan hasil akhir dari perhitungan.

Setelah deklarasi variabel, program meminta pengguna untuk memasukkan nilai **K_226** dengan pesan "Nilai K = ". Nilai yang dimasukkan oleh pengguna ini akan menjadi batas iterasi dalam perhitungan, yaitu sebanyak **K_226** + 1 kali. Program kemudian memasuki loop for yang akan berjalan dari 0 hingga **K_226**.