

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 2**

**RIVIEW STRUKTUR KONTROL**



**Disusun Oleh : Brian Farrel Evandhika 2311102037**

**IF 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

## 2.1 Struktur Program Go

Go, atau Golang, adalah bahasa pemrograman yang dirancang untuk kinerja tinggi dan pemrosesan secara paralel (concurrency). Setiap program Go memiliki elemen-elemen dasar seperti paket, fungsi utama, dan pernyataan impor.

**Paket (Package):** Semua program Go dimulai dengan mendefinisikan paket. Paket `main` khusus digunakan untuk program yang dapat dijalankan secara langsung.

**Pernyataan Impor (Import Statement):** Digunakan untuk memasukkan pustaka atau modul eksternal yang diperlukan, seperti pustaka `fmt` untuk melakukan operasi input dan output.

**Fungsi Utama (Main Function):** Fungsi `main` adalah titik awal eksekusi dari setiap program Go yang dapat dijalankan.

Contoh sederhana program Go:

```
package main
import "fmt"

func main() {
    fmt.Println("Hello, World!")
}
```

Di luar elemen-elemen inti ini, program Go juga dapat mengandung variabel, konstanta, dan fungsi lain yang mendukung fungsi `main`.

## 2.2 Tipe Data dan Instruksi Dasar

Go memiliki beberapa kategori tipe data, seperti tipe data dasar, tipe komposit, dan tipe referensi.

**Tipe Data Dasar:** Ini termasuk tipe-tipe seperti `int`, `float64`, `bool`, dan `string`. Tipe numerik juga memiliki beberapa variasi seperti `int8`, `int16`, `int32`, dan `int64` untuk bilangan bulat, serta `float32` dan `float64` untuk bilangan pecahan.

Tipe Komposit: Tipe data ini mencakup array, slice, dan map, yang memungkinkan penyimpanan beberapa nilai dalam satu variabel.

Instruksi Dasar:

Deklarasi Variabel: Bisa dilakukan menggunakan kata kunci var atau melalui inferensi tipe secara otomatis.

```
var x int = 10
y := 20.5 // tipe otomatis ditentukan
```

Operasi Dasar: Meliputi operasi aritmatika seperti (+, -, \*, /), operasi perbandingan (==, !=, <, >), dan operasi logika (&&, ||).

Pernyataan Input/Output: Pustaka fmt menyediakan fungsi untuk menangani input dan output.

```
fmt.Println("Nilai x:", x)
fmt.Scanln(&x)
```

## 2.3 Struktur Kontrol Perulangan

Go menggunakan satu bentuk utama perulangan, yaitu for. Struktur ini sangat fleksibel dan dapat menggantikan bentuk perulangan lainnya, seperti while dan do-while di bahasa pemrograman lain.

Perulangan for Klasik: Terdiri dari inisialisasi variabel, kondisi, dan perubahan variabel dalam satu baris.

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

Perulangan Berdasarkan Kondisi: Setara dengan perulangan while pada bahasa lain.

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

Perulangan Tak Terbatas: Bisa dijalankan tanpa batas waktu dan diakhiri menggunakan perintah seperti break.

```
for {
    fmt.Println("Perulangan tanpa henti")
    break // diperlukan untuk menghentikan
}
```

Dengan hanya menggunakan struktur for, Go mampu menangani berbagai jenis perulangan, menjadikannya bahasa yang efisien dan sederhana untuk pengelolaan alur kontrol perulangan.

## II. GUIDED

### Soal Studi Case 2.1

Struktur Program Go

Sourcecode

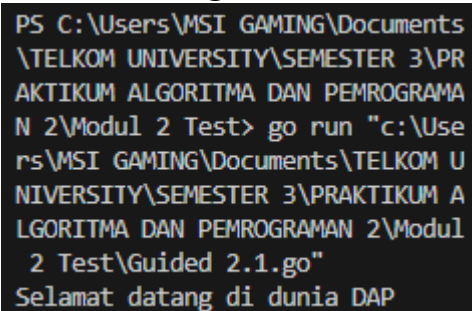
```
package main

import "fmt"

func main() {
    var greetings = "Selamat datang di dunia DAP"
    var a, b int

    fmt.Println(greetings)
    fmt.Scanln(&a, &b)
    fmt.Printf("%v + %v = %v\n", a, b, a+b)
}
```

Screenshot Program



```
PS C:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PR  
AKTIKUM ALGORITMA DAN PEMROGRAMA  
N 2\Modul 2 Test> go run "c:\Use  
rs\MSI GAMING\Documents\TELKOM U  
NIVERSITY\SEMESTER 3\PRAKTIKUM A  
LGORITMA DAN PEMROGRAMAN 2\Modul  
2 Test\Guided 2.1.go"  
Selamat datang di dunia DAP
```

Deskripsi Program

Program Go ini menyapa pengguna dengan pesan "Selamat datang di dunia DAP". Kemudian, program meminta pengguna untuk memasukkan dua angka. Setelah angka-angka tersebut dimasukkan, program menampilkan hasil penjumlahan dari kedua angka tersebut dalam format:

a+b=hasil.

## Soal Studi Case 2.2

Tipe Data dan Instruksi Dasar

### Sourcecode

```
package main

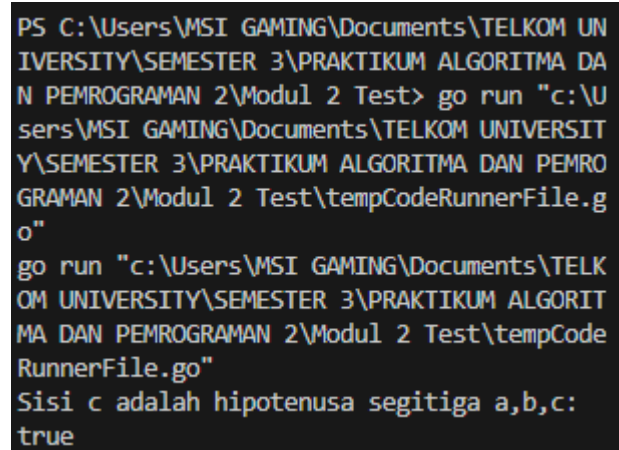
import "fmt"

func main() {
    var a, b, c float64
    var hipotenusa bool

    fmt.Scanln(&a, &b, &c)
    hipotenusa = (c*c) == (a*a + b*b)

    fmt.Println("Sisi c adalah hipotenusa segitiga a,b,c: ",
hipotenusa)
}
```

### Screenshot Program



```
PS C:\Users\MSI GAMING\Documents\TELKOM UN
IVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DA
N PEMROGRAMAN 2\Modul 2 Test> go run "c:\U
sers\MSI GAMING\Documents\TELKOM UNIVERSIT
Y\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMRO
GRAMAN 2\Modul 2 Test\tempCodeRunnerFile.g
o"
go run "c:\Users\MSI GAMING\Documents\TELK
OM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORIT
MA DAN PEMROGRAMAN 2\Modul 2 Test\tempCode
RunnerFile.go"
Sisi c adalah hipotenusa segitiga a,b,c:
true
```

### Deskripsi Program

Program Go ini memeriksa apakah sisi c dari segitiga dengan sisi-sisi a, b, dan c adalah hipotenusa. Pengguna memasukkan nilai untuk a, b, dan c. Program kemudian menghitung apakah

$c^2 = a^2 + b^2$  Jika benar, variabel hipotenusa diatur ke true, menandakan bahwa c adalah hipotenusa. Hasil ini kemudian ditampilkan kepada pengguna.

## Soal Studi Case 2.3

### Kontrol Perulangan

#### Sourcecode

```
package main

import "fmt"

func main() {
    maxF := 100
    f0 := 0
    f1 := 1
    f2 := 1

    fmt.Println("Bilangan pertama:", f1)

    for f2 <= maxF {
        f0 = f1
        f1 = f2
        f2 = f1 + f0
        fmt.Println("Bilangan berikutnya:", f1)
    }
}
```

#### Screenshot Program

```
PS C:\Users\MSI GAMING\Documents\TELKOM UN
IVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DA
N PEMROGRAMAN 2\Modul 2 Test> go run "c:\U
sers\MSI GAMING\Documents\TELKOM UNIVERSIT
Y\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMRO
GRAMAN 2\Modul 2 Test\Guided 2.3.go"
Bilangan pertama: 1
Bilangan berikutnya: 1
Bilangan berikutnya: 2
Bilangan berikutnya: 3
Bilangan berikutnya: 5
Bilangan berikutnya: 8
Bilangan berikutnya: 13
Bilangan berikutnya: 21
Bilangan berikutnya: 34
Bilangan berikutnya: 55
Bilangan berikutnya: 89
PS C:\Users\MSI GAMING\Documents\TELKOM UN
IVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DA
N PEMROGRAMAN 2\Modul 2 Test>
```

## Deskripsi Program

Program Go ini menghasilkan dan menampilkan bilangan Fibonacci hingga nilai maksimum 100. Program dimulai dengan nilai awal untuk tiga variabel: f0 (0), f1 (1), dan f2 (1). Bilangan pertama yang ditampilkan adalah f1. Dalam loop, program menghitung bilangan Fibonacci berikutnya dengan menambahkan dua bilangan sebelumnya (f0 dan f1), menggeser nilai variabel untuk langkah selanjutnya, dan menampilkan bilangan Fibonacci yang dihasilkan. Proses ini berlanjut hingga nilai bilangan Fibonacci melebihi 100.

## Soal Studi Case 2A 1

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

### Sourcecode

```
//2311102037 BRIAN FARREL EVANDHIKA IF 11 06
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)
}
```



## Screenshoot Output

```
PS C:\Users\MSI GAMING\laravel-11> go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul 2\package main.go"
Masukan input string: brian
Masukan input string: farrel
Masukan input string: evandhika
Output awal = brian farrel evandhika
Output akhir = farrel evandhika brian
PS C:\Users\MSI GAMING\laravel-11> █
```

## Deskripsi Program

Program Go ini meminta pengguna untuk memasukkan tiga string, menyimpannya dalam variabel satu, dua, dan tiga. Setelah itu, program menampilkan urutan awal ketiga string tersebut. Kemudian, nilai string dirotasikan: satu menjadi dua, dua menjadi tiga, dan tiga menjadi satu. Urutan baru string ini kemudian ditampilkan.

Soal Studi Case 2A 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).

## Sourcecode

```
//2311102037 BRIAN FARREL EVANDHIKA IF 11 06
package main

import "fmt"

func main() {
    var tahun int
    var kabisat bool

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

    if tahun%400 == 0 {
        kabisat = true
    } else if tahun%100 == 0 {
        kabisat = false
    } else if tahun%4 == 0 {
        kabisat = true
    }
}
```

```

    } else {
        kabisat = false
    }

    fmt.Println("Kabisat:", kabisat)
}

```

### Screenshoot Output

```

-Module PSReadLine'.

PS C:\Users\MSI GAMING\laravel-11> go run "c:
\Users\MSI GAMING\Documents\TELKOM UNIVERSITY
\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAM
AN 2\Modul 2\tempCodeRunnerFile.go"
Tahun: 2014
Kabisat: false
PS C:\Users\MSI GAMING\laravel-11> go run "c:
\Users\MSI GAMING\Documents\TELKOM UNIVERSITY
\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAM
AN 2\Modul 2\tempCodeRunnerFile.go"
Tahun: 2020
Kabisat: true
PS C:\Users\MSI GAMING\laravel-11>

```

### Deskripsi Program

Program Go ini menghitung tahun kabisat dengan menggunakan modulus 4 yang dimana jika kita mengisikan inputan tahun yang jika digunakan modulus 4 memiliki sisa bagi maka output yang keluar akan menjadi false tapi jika sisa bagi modulus 4 sama dengan 0 maka output yang keluar akan true

### Soal Studi Case 2A 3

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

### Sourcecode

```

//2311102037 BRIAN FARREL EVANDHIKA IF 11 06
package main

```

```

import (
    "fmt"
    "math"
)

func main() {
    var jejari float64
    const pi = 3.1415926535

    fmt.Print("Jejari = ")
    fmt.Scanln(&jejari)

    volume := (4.0 / 3.0) * pi * math.Pow(jejari, 3)

    luas := 4 * pi * math.Pow(jejari, 2)

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

```

#### Screenshot Program

```

PS C:\Users\MSI GAMING\laravel-11>
go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRA
TIKUM ALGORITMA DAN PEMROGRAMAN 2\
Modul 2\tempCodeRunnerFile.go"
Jejari = 21
Bola dengan jejari 21 memiliki volu
me 38792.3861 dan luas kulit 5541.7
694
PS C:\Users\MSI GAMING\laravel-11>

```

#### Deskripsi Program

Program Go ini menghitung volume dan luas permukaan bola dari jejari yang dimasukkan pengguna. Setelah menerima nilai jejari, program menggunakan rumus matematika untuk menghitung volume dan luas permukaan bola. Hasil perhitungan ini kemudian ditampilkan kepada p

### III. UNGUIDED

#### Soal Studi Case 2A 4

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

### Source Code

```
//2311102037 BRIAN FARREL EVANDHIKA IF 11 06
package main

import (
    "fmt"
)

func main() {

    var celsius float64

    fmt.Print("Masukkan suhu dalam derajat Celsius: ")
    fmt.Scanf("%f", &celsius)

    reaumur := (4.0 / 5.0) * celsius

    fahrenheit := (celsius * 9.0 / 5.0) + 32.0

    kelvin := celsius + 273.15

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

### Screenshot Program

```
PS C:\Users\MSI GAMING\laravel-11>
go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRA
TIKUM ALGORITMA DAN PEMROGRAMAN 2\
Modul 2 Test\2a 4.go"
Masukkan suhu dalam derajat Celsius
: 30
Temperatur Celsius: 30.00
Derajat Reaumur: 24.00
Derajat Fahrenheit: 86.00
Derajat Kelvin: 303.15
PS C:\Users\MSI GAMING\laravel-11>
```

### Deskripsi Program

Program Go ini mengonversi suhu dari derajat Celsius ke beberapa skala suhu lainnya: Reaumur, Fahrenheit, dan Kelvin. Setelah pengguna memasukkan suhu dalam derajat Celsius, program menghitung suhu dalam derajat Reaumur Celsius Fahrenheit Celsius dan Kelvin Hasil konversi ini kemudian ditampilkan kepada pengguna.

### Soal Studi Case 2A 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).

### Source Code

```
package main

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

func main() {
    var a, b, c, d, e int
    reader := bufio.NewReader(os.Stdin)

    fmt.Println("Masukkan 5 buah angka (nilai antara 32 hingga 127):")
    fmt.Scanf("%d %d %d %d %d", &a, &b, &c, &d, &e)

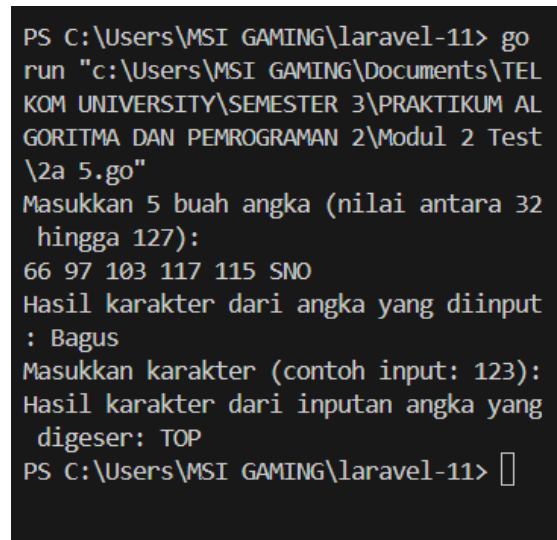
    fmt.Printf("Hasil karakter dari angka yang diinput: %c%c%c%c%c\n", a, b, c, d, e)

    fmt.Println("Masukkan karakter (contoh input: 123):")
    inputStr, _ := reader.ReadString('\n')

    shiftedStr := ""
    for _, char := range inputStr {
        if char != '\n' && char != '\r' {
            shiftedStr += string(char + 1)
        }
    }

    fmt.Printf("Hasil karakter dari inputan angka yang digeser: %s\n", shiftedStr)
}
```

## Screenshot Program



```
PS C:\Users\MSI GAMING\laravel-11> go
run "c:\Users\MSI GAMING\Documents\TEL
KOM UNIVERSITY\SEMESTER 3\PRAKTIKUM AL
GORITMA DAN PEMROGRAMAN 2\Modul 2 Test
\2a 5.go"
Masukkan 5 buah angka (nilai antara 32
hingga 127):
66 97 103 117 115 SNO
Hasil karakter dari angka yang diinput
: Bagus
Masukkan karakter (contoh input: 123):
Hasil karakter dari inputan angka yang
digeser: TOP
PS C:\Users\MSI GAMING\laravel-11> █
```

## Deskripsi Program

Program Go ini meminta pengguna untuk memasukkan lima angka antara 32 dan 127 serta beberapa karakter. Angka-angka tersebut dikonversi menjadi karakter ASCII dan ditampilkan. Selain itu, program membaca inputan karakter sebagai string, lalu menggeser setiap karakter dalam string tersebut satu posisi ke depan dalam kode ASCII, dan hasilnya ditampilkan kepada pengguna.

## Soal Studi Case 2B 1

Siswa kelas IPA disalah 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 berturutan 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.

## Source Code

```
//2311102037 BRIAN FARREL EVANDHIKA IF 11 06
package main

import (
    "fmt"
)

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

    for i := 0; i < 5; i++ {
        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 := 0; i < 5; i++ {
        fmt.Printf("percobaan %d: %s %s %s %s\n", i+1, warna[i][0],
warna[i][1], warna[i][2], warna[i][3])
        if warna[i] != urutanBenar {
            berhasil = false
        }
    }

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



### Screenshot Program

```
PS C:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul 2 Test>
go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul 2 Test\tempCodeRunnerFile.go"
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): ungu merah kuning hijau
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: ungu merah kuning hijau
percobaan 4: merah kuning hijau ungu
percobaan 5: merah kuning hijau ungu
berhasil : false
PS C:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul 2 Test>
```

### Deskripsi Program

Program Go ini meminta pengguna untuk memasukkan urutan warna dalam lima percobaan terpisah, setiap percobaan terdiri dari empat warna.

Program kemudian memeriksa apakah setiap percobaan cocok dengan urutan warna yang benar (merah, kuning, hijau, ungu). Jika semua percobaan cocok dengan urutan benar, variabel berhasil akan tetap true, dan jika ada percobaan yang tidak cocok, berhasil akan diatur ke false. Hasil akhir (berhasil: true atau false) ditampilkan kepada pengguna.

## Soal Studi Case 2B 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.

Tampilkan isi pita setelah proses input selesai. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read)

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 dibawah ini (teks bergaris bawah adalah input/read):

## Source Code

```
//2311102037 BRIAN FARREL EVANDHIKA IF 11 06

package main

import (
    "fmt"
    "strings"
)

func main() {
    var N int
    fmt.Print("Masukkan jumlah bunga (N): ")
    fmt.Scan(&N)
```

```
var pita strings.Builder
var bunga string
count := 0

for i := 1; i <= N; i++ {
    fmt.Printf("Bunga %d: ", i)
    fmt.Scan(&bunga)

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

    if pita.Len() > 0 {
        pita.WriteString(" - ")
    }
    pita.WriteString(bunga)
    count++
}

fmt.Printf("Pita: %s\n", pita.String())
fmt.Printf("Bunga: %d\n", count)
}
```

### Screenshot Program

```
PS C:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul 2 Test>
go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul 2 Test\2b 2.go"
Masukkan jumlah bunga (N): 1
Bunga 1: Tulip
Pita: Tulip
Bunga: 1
PS C:\Users\MSI GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul 2 Test>
```

### Deskripsi Program

Program Go ini meminta pengguna untuk memasukkan sejumlah nama bunga dalam sebuah pita, hingga mencapai jumlah tertentu (N) atau sampai pengguna memasukkan kata "SELESAI". Nama-nama bunga digabungkan dalam sebuah string dengan tanda "-" sebagai pemisah. Program juga menghitung dan menampilkan jumlah bunga yang dimasukkan.

### Soal Studi Case 2B 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):

## Source Code

```
// 2311102037 BRIAN FARREL EVANDHIKA IF 11 06

package main

import (
    "fmt"
)

func main() {
    var berat1, berat2 float64

    for {
        fmt.Print("masukkan berat belanjaan di kedua kantong = ")
        fmt.Scan(&berat1, &berat2)

        if berat1 >= 9 || berat2 >= 9 {
            fmt.Println("sepeda motor pak andi akan oleng : true")
        } else {
            fmt.Println("sepeda motor pak andi akan oleng : false")
        }
    }
}
```

## Screenshot Program

```
PS C:\Users\MSI GAMING\Documents\TELKOM UNIVE
RSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMR
OGRAMAN 2\Modul 2 Test> go run "c:\Users\MSI
GAMING\Documents\TELKOM UNIVERSITY\SEMESTER 3
\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Modul
2 Test\tempCodeRunnerFile.go"
masukkan berat belanjaan di kedua kantong =
2.0 2.5
sepeda motor pak andi akan oleng : false
masukkan berat belanjaan di kedua kantong =
```

### Deskripsi Program

Program Go ini meminta pengguna untuk memasukkan berat belanjaan di dua kantong secara berulang. Jika salah satu kantong memiliki berat 9 kg atau lebih, program akan menampilkan pesan bahwa sepeda motor Pak Andi akan oleng (berat sebelah). Jika tidak, program menyatakan sepeda motor tidak akan oleng. Proses ini berlangsung dalam loop tanpa henti sampaiimbang.

### Soal Studi Case 2B 4

buatlah 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 dibawah ini (teks bergaris bawah adalah input/read):

### Source Code

```
// 2311102037 BRIAN FARREL EVANDHIKA IF 11 06
package main

import (
    "fmt"
    "math"
)

func hitungF(k int) float64 {
    pembilang := math.Pow(float64(4*k+2), 2)
    penyebut := float64((4*k + 1) * (4*k + 3))
```

```
    return pembilang / penyebut
}

func hitungAkarDua(k int) float64 {
    hasil := 1.0
    for i := 0; i <= k; i++ {
        hasil *= hitungF(i)
    }
    return hasil
}

func main() {
    var k int
    fmt.Print("Masukkan nilai K: ")
    fmt.Scan(&k)

    nilaiF := hitungF(k)
    fmt.Printf("Nilai f(%d) = %.10f\n", k, nilaiF)

    akarDua := hitungAkarDua(k)
    fmt.Printf("Nilai akar 2 untuk K = %d: %.10f\n", k, akarDua)
}
```

**Screenshot Program**

```
PS C:\Users\MSI GAMING\Documents\TELKOM UN  
IVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DA  
N PEMROGRAMAN 2\Modul 2 Test> go run "c:\U  
sers\MSI GAMING\Documents\TELKOM UNIVERSIT  
Y\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMRO  
GRAMAN 2\Modul 2 Test\tempCodeRunnerFile.g  
o"
```

Masukkan nilai K: 100

Nilai  $f(100) = 1.0000061880$

Nilai akar 2 untuk K = 100: 1.4133387072

```
PS C:\Users\MSI GAMING\Documents\TELKOM UN  
IVERSITY\SEMESTER 3\PRAKTIKUM ALGORITMA DA  
N PEMROGRAMAN 2\Modul 2 Test> █
```

### Deskripsi Program

Program Go ini menghitung nilai dari sebuah fungsi matematika dan mendekati nilai akar dua dengan nilai K tertentu. Fungsi  $hitungF(k)$  menghitung nilai dengan pembilang dan penyebut. Fungsi  $hitungAkarDua(k)$  mengalikan hasil dari  $hitungF(i)$  dari 0 sampai K untuk mendekati nilai akar dua. Program meminta input nilai K dari pengguna, kemudian menampilkan nilai dan hasil pendekatan nilai akar dua untuk K yang dimasukkan.