

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



Oleh:

**PRATAMA BINTANG DANISWARA**

103112400051

12 IF 01

**S1 TEKNIK INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2025**

## I. DASAR TEORI

Go (Golang) adalah bahasa pemrograman yang dikembangkan oleh Google untuk efisiensi, kecepatan, dan kemudahan dalam pemrograman sistem. Go memiliki sintaks sederhana tetapi kuat, cocok untuk berbagai keperluan, mulai dari aplikasi backend hingga pemrograman sistem.

## II. GUIDED

Source Code + Screenshot hasil program beserta penjelasan

Contoh 1

```
package main

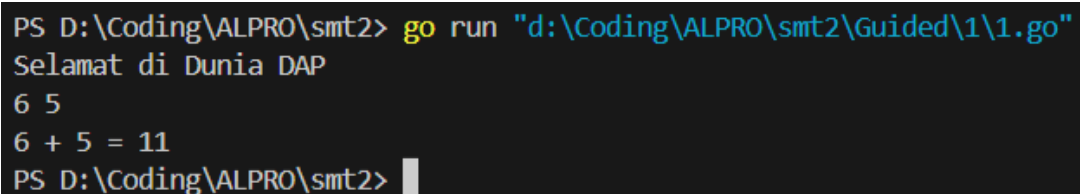
import "fmt"

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

    fmt.Println(greetings)
    fmt.Scan(&a, &b)

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

Screenshots Output



```
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\Guided\1\1.go"
Selamat di Dunia DAP
6 5
6 + 5 = 11
PS D:\Coding\ALPRO\smt2>
```

Deskripsi: Program ini memulai dengan menampilkan pesan sapaan "SELAMAT DATANG DI DUNIA DAP", Program ini menunjukkan dasar penggunaan bahasa Go, termasuk deklarasi variabel, input pengguna, dan output format. Program pertama-tama mencetak pesan sambutan, lalu menerima dua angka sebagai masukan dari pengguna, dan akhirnya menampilkan hasil penjumlahan kedua angka tersebut dengan format yang rapi.

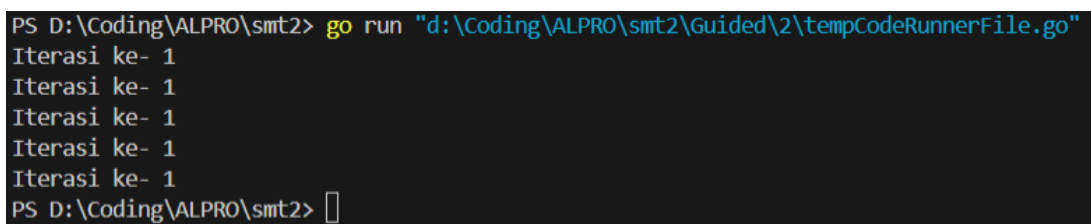
## Contoh 2

```
package main

import "fmt"

func main() {
    for i := 1; i <= 5; i++ {
        fmt.Println("Iterasi ke-", i)
    }
}
```

## Screenshots Output



```
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\Guided\2\tempCodeRunnerFile.go"
Iterasi ke- 1
Iterasi ke- 1
Iterasi ke- 1
Iterasi ke- 1
Iterasi ke- 1
PS D:\Coding\ALPRO\smt2> █
```

Deskripsi: Program ini menerapkan **struktur perulangan for** dalam bahasa Go untuk mencetak teks secara berulang. Perulangan berjalan dari  $i = 1$  hingga  $i = 5$ , dengan setiap iterasi menaikkan nilai  $i$  sebesar satu.

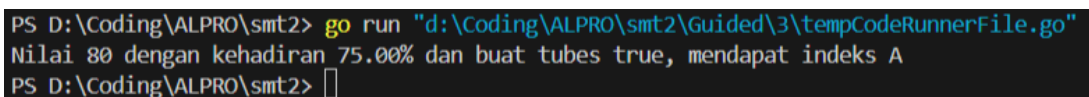
### Contoh 3

```
package main

import "fmt"

func main() {
    nilai := 80
    pctHadir := 0.75
    adaTubes := true
    var indeks string
    if nilai > 75 && adaTubes {
        indeks = "A"
    } else if nilai > 65 {
        indeks = "B"
    } else if nilai > 50 && pctHadir > 0.7 {
        indeks = "C"
    } else {
        indeks = "F"
    }
    fmt.Printf("Nilai %d dengan kehadiran %.2f%% dan buat tubes %t, mendapat indeks %s\n", nilai, pctHadir*100, adaTubes, indeks)
}
```

### Screenshots Output



```
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\Guided\3\tempCodeRunnerFile.go"
Nilai 80 dengan kehadiran 75.00% dan buat tubes true, mendapat indeks A
PS D:\Coding\ALPRO\smt2> 
```

Deskripsi: Program mencetak nilai akhir, persentase kehadiran, status pengerjaan tugas besar (true/false), serta grade yang diperoleh.

### III. UNGUIDED

Source Code + Screenshot hasil program beserta penjelasan

#### Soal 1

```
// PRATAMA BINTANG DANISWARA
// 103112400051

package main

import "fmt"

func main() {
    var tahun int

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

    switch {
    case tahun%400 == 0:
        fmt.Print("Kabisat: true")

    case tahun%100 == 0:
        fmt.Print("Kabisat: false")

    case tahun%4 == 0:
        fmt.Print("Kabisat: true")

    default:
        fmt.Print("Kabisat: false")
    }
}
```

Screenshots Output

```

PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\2a no 2\tempCodeRunnerFile.go"
Tahun: 2016
Kabisat: true
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\2a no 2\tempCodeRunnerFile.go"
Tahun: 2020
Kabisat: true
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\2a no 2\tempCodeRunnerFile.go"
Tahun: 2018
Kabisat: false
PS D:\Coding\ALPRO\smt2>

```

Deskripsi: Program ini menentukan tahun kabisat menggunakan switch-case di Go. Tahun yang habis dibagi 400 dianggap kabisat, sedangkan yang habis dibagi 100 bukan kabisat, tetapi jika hanya habis dibagi 4 tetap kabisat. Urutan pengecekan kurang tepat, sehingga bisa menghasilkan output salah. Penggunaan switch menunjukkan cara efisien menangani kondisi numerik, meskipun perlu perbaikan dalam logika.

## Soal 2

```

// PRATAMA BINTANG DANISWARA
// 103112400051
package main

import "fmt"

func main() {
    var K int
    fmt.Print("Masukan nilai K: ")
    fmt.Scan(&K)
    atas := float64((4*K + 2) * (4*K + 2))
    bawah := float64((4*K + 1) * (4*K + 3))
    hasil := float64((atas / bawah))
    fmt.Printf("Nilai fk(k) = %.10f\n", hasil)
}

```

Screenshots Output

```
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\2b no 4\2b no 4.go"
Masukan nilai K: 100
Nilai f(k) = 1.0000061880
PS D:\Coding\ALPRO\smt2> █
```

Deskripsi: Program menghitung rasio fungsi dengan input K, lalu menampilkan hasilnya dengan presisi 10 desimal.

### Soal 3

```
// PRATAMA BINTANG DANISWARA
// 103112400051
package main

import "fmt"

func main() {
    var beratgr, sisa int
    fmt.Print("Berat parcel (gram): ")
    fmt.Scanln(&beratgr)

    parcel := beratgr / 1000
    berat := beratgr % 1000
    fmt.Printf("Detail berat: %d kg + %d
gr\n", parcel, berat)

    biaya := parcel * 10000
    if berat < 500 {
        sisa = berat * 15
    } else if berat >= 500 {
        sisa = berat * 5
    }

    fmt.Printf("Detail biaya: Rp. %d + Rp.
%d \n", biaya, sisa)
    fmt.Printf("Total biaya: Rp. %d",
biaya+sisa)
}
```

Screenshots Output

```
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\2c no 1\22.go"
Berat parcel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\2c no 1\22.go"
Berat parcel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
PS D:\Coding\ALPRO\smt2> go run "d:\Coding\ALPRO\smt2\2c no 1\22.go"
Berat parcel (gram): 11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 3750
Total biaya: Rp. 113750
PS D:\Coding\ALPRO\smt2> █
```

Deskripsi: Program menghitung biaya pengiriman berdasarkan berat paket dalam gram, dengan tarif dasar per kilogram dan tambahan biaya jika berat lebih dari 500 gram.

#### IV. KESIMPULAN

Secara keseluruhan, berbagai program yang telah dianalisis menunjukkan bagaimana Go memanfaatkan struktur kendali seperti if-else, switch, dan for untuk mengelola alur program. Selain itu, program-program ini juga mengajarkan cara menangani operasi matematika, mengelola variabel, serta menerima dan menampilkan input/output. Pemahaman terhadap konsep-konsep ini memungkinkan kita untuk menyusun program yang lebih dinamis dan efisien dalam menyelesaikan berbagai permasalahan komputasi.



## **V.     REFERENSI**

MODUL 2. REVIEW STRUKTUR KONTROL, Praktikum Algoritma dan Pemrograman 2