

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II
REVIEW STRUKTUR KONTROL**



Disusun Oleh :

Afra Lintang Maharani / 2311102258

S1 – IF – 11 - 05

Dosen Pengampu :

Arif Amrulloh, S.Kom, M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Struktur kontrol adalah komponen yang sangat penting dalam mendetermin kejalannya pemrograman. Bahasa Go, atau Golang, memiliki beberapa struktur kontrol elementer penting dari sesuatu bagi seorang developer untuk mengontrol aliran logika di dalam programnya. Berikut adalah ulasan mengenai beberapa struktur kontrol yang umum digunakan dalam bahasa Go,

- Percabangan (Conditional Statements)
 - if Statement: Dapat dijalankan untuk mengggunkunci satu atau beberapa blok kode tergantung pada nilai boolean pada ekspresi.
 - if-else Statement: bertujuan untuk memberikan blok alternative bila keadaan di dalam if yang tidak tercapai.
 - if-else if-else Statement: Menghadirkan beberapa kondisi dengan tingkat kompleksitas.
- Perulangan (Loops),
For Loop: Yang hanya ada di Go dan dapat digunakan sebagai loop biasa, while loop dan loop forever.
- Switch,
Switch Statement: Membantu menyediakan cara yang lebih bersih dalam menulis sejumlah besar pernyataan if-else dalam jumlah bertingkat. Switch pada Go juga dapat diisi dengan berbagai macam seperti string dan integer.
- Select,
Select Statement: Terpakai dalam beberapa proses channel operation, mengizinkan goroutine untuk menunggu beberapa komunikasi channel.

II. GUIDED

1. Soal Studi Case

Sourcecode

```
package main

import "fmt"

func main() {
    nama := "Afra Lintang Maharani"
    fmt.Print(nama)
}
```

Screenshoot Output

```
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\G1_M2.go"
Afra Lintang Maharani
PS D:\Dokument\Semester 2\.vscode>
```

Deskripsi Program

Program diatas adalah contoh program yang berguna untuk menampilkan nama seorang mahasiswa

2. Soal Studi Case

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

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			
Percobaan 1:	merah	kuning	hijau	ungu
Percobaan 2:	merah	kuning	hijau	ungu
Percobaan 3:	merah	kuning	hijau	ungu
Percobaan 4:	ungu	kuning	hijau	merah
Percobaan 5:	merah	kuning	hijau	ungu
BERHASIL:	false			

Sourcecode

```
package main

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

func main() {
    //urutan warna yang benar
    correctOrder := []string{"merah", "kuning", "hijau",
"ungu"}

    //membaca input untuk 5 percobaan
    reader := bufio.NewReader(os.Stdin)
    success := true

    for i := 1; i <= 5; i++ {
        fmt.Printf("Percobaan %d: ", i)

        //membaca input dari pengguna
        input, _ := reader.ReadString('\n')
        input = strings.TrimSpace(input)

        //memisahkan input berdasarkan spasi
        colors := strings.Split(input, " ")

        //mengecek apakah urutan warna sesuai
        for j := 0; j < 4; j++ {
            if colors[j] != correctOrder[j] {
                success = false
                break
            }
        }
    }
}
```

```

        //jika ada percobaan yang tidak sesuai, keluar dari
loop
    if !success {
        break
    }
}

//menampilkan hasil
if success {
    fmt.Println("BERHASIL: true")
} else {
    fmt.Println("BERHASIL: false")
}
}

```

Screenshoot Output

```

PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\G1_M2.go"
Afra Lintang Maharani
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\guided1_Modul2.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

```

Deskripsi Program

Program diatas adalah contoh program yang digunakan untuk mencocokkan inputan user dengan data yang ada pada array. Sebuah array di deklarasikan untuk kemudian digunakan untuk sebuah database dengan nama correctOrder. Kemudian bufio.NewReader digunakan untuk membaca inputan panjang dari user, kemudian untuk membaca inputan yang memiliki spasi program ini menggunakan strings.Split. Dalam program ini user akan diminta untuk menginput warna, kemudian program akan mulai mencocokkan apakah warna yang telah di input oleh user ini sesuai dengan data warna yang ada pada array, jika sesuai maka output akan menghasilkan true, namun jika salah output program adalah false.

3. Soal Studi Case

Sourcecode

```

package main

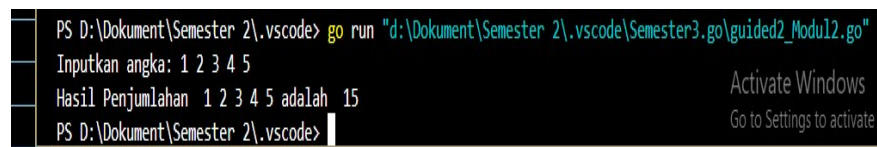
```

```
import (
    "fmt"
)

func main() {
    var a, b, c, d, e int
    var hasil int
    fmt.Print("Inputkan angka: ")
    fmt.Scanln(&a, &b, &c, &d, &e)

    hasil = a + b + c + d + e
    fmt.Println("Hasil Penjumlahan ", a, b, c, d, e, "adalah", hasil)
}
```

Screenshoot Output



```
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\guided2_Modul2.go"
Inputkan angka: 1 2 3 4 5
Hasil Penjumlahan 1 2 3 4 5 adalah 15
PS D:\Dokument\Semester 2\.vscode>
```

Deskripsi Program

Program diatas adalah contoh program yang melakukan penjumlahan 5 angka yang akan di inputkan oleh user, variabel yang ada di dalam program seperti a,b,c,d,e dan hasil memiliki tipe data yang sama. Pada program diatas terdapat fungsi yaitu $hasil = a + b + c + d + e$, pada program ini user akan diminta untuk menginputkan 5 angka secara acak, kemudian program akan menjumlahkan angka-angka yang telah di inputkan user lalu mengeluarkan output berupa hasil penjumlahan tersebut.

4. Soal Studi Case

Diberikan sebuah nilai akhir mata kullah (NAM) [0..100] dan standar penilaian nilai mata kullah (NMK) sebagai berikut:

NAM	NMK
$NAM > 80$	A
$72.5 < NAM \leq 80$	AB
$65 < NAM \leq 72.5$	B
$57.5 < NAM \leq 65$	BC
$50 < NAM \leq 57.5$	C
$40 < NAM \leq 50$	D
$NAM \leq 40$	E

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

Sourcecode

```
package main

import "fmt"

func main() {
    var nam float32
    var nmk string

    // Meminta input nilai
    fmt.Print("Masukkan nilai : ")
    fmt.Scan(&nam)

    // Logika penentuan nilai huruf berdasarkan nilai
    numerik
    if nam > 80 {
        nmk = "A"
    } else if nam > 72.5 {
        nmk = "B"
    } else if nam > 65 {
        nmk = "C"
    } else if nam > 50 {
        nmk = "D"
    } else if nam > 40 {
        nmk = "E"
    } else {
        nmk = "F"
    }

    // Menampilkan hasil
    fmt.Printf("Nilai Indeks untuk nilai %.2f adalah %s\n",
    nam, nmk)
```

```
} }
```

Screenshoot Output

```
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\guided3_Modul2.go"
Masukkan nilai : 97
Nilai Indeks untuk nilai 97.00 adalah A
PS D:\Dokument\Semester 2\.vscode> 
```

Deskripsi Program

Program diatas adalah contoh program yang berguna untuk memilah nilai akhir dengan kategori indeks nilai. Dalam program, kita menggunakan 2 variabel yaitu `nm` `float32` dan juga `nmk` `string`, dalam program diatas kita juga banyak menggunakan `if else` (percabangan) karena kita memerlukan banyak percabangan untuk memisahkan antara nilai akhir yang akan di inputkan user dan kategori indeks nya. Untuk program diatas sendiri, nantinya user akan diminta untuk menginputkan nilai yang ingin di inputkan, lalu program akan mencocokkan nilai yang diinputkan oleh user dengan data kategori yang ada, sehingga akhirnya program akan mengeluarkan output berupa indeks kategori dari nilai yang sudah di inputkan oleh user.

III. UNGUIDED

1. Soal Studi Case

Suatu pita (string) berisi kumpulan nama-nama bunga yang dipisahkan oleh spasi '- ', 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.
- Modifikasi program sebelumnya, proses input akan berhenti apabila user mengetikkan 'SELESAI'. Kemudian tampilkan isi pita beserta banyaknya bunga yang ada di dalam pita.

(Petunjuk: gunakan operasi penggabungan string dengan operator "+").

Sourcecode

```
package main

import (
    "fmt"
    "strings"
)

func main() {
    var (
        n      = 5
        bunga  string
        Pita   []string
        banyak int
    )
    //fmt.Print("N : ")
    //fmt.Scanln(&n)
    for i := 1; i <= n; i++ {
        fmt.Printf("Bunga %d: ", i)
        fmt.Scanln(&bunga)
        if bunga == "Selesai" || bunga == "selesai" {
            break
        }
        Pita = append(Pita, bunga)
        banyak++
    }
    temp := strings.Join(Pita, " - ")
    fmt.Println("Pita : " + temp + " - ")
}
```

```
fmt.Print("Bunga : ", banyak)
}
```

Screenshoot Output

a. Program a

```
N : 3
Bunga 1: Kertas
Bunga 2: Mawar
Bunga 3: Tulip
Pita : Kertas - Mawar - Tulip -
```

b. Program b

```
Bunga 1: Kertas
Bunga 2: Mawar
Bunga 3: Tulip
Bunga 4: Selesai
Pita : Kertas - Mawar - Tulip -
Bunga : 3
```

Deskripsi Program

Program diatas adalah program yang digunakan untuk menginput bunga untuk kemudian diletakkan kedalam sebuah pita, dalam program diatas sendiri terdapat beberapa variabel seperti n (nilai n yang awalnya merupakan inputan user di setting menjadi 5, nilai n ini mengiterasikan jumlah maksimum input bunga), kemudian ada Bunga (Bunga adalah variabel sementara yang digunakan untuk menyimpan inputan bunga dari user), lalu ada Pita (pita adalah sebuah array / slice, dengan tipe string yang digunakan untuk menyimpan daftar bunga yang sudah diinput), lalu ada banyak (variabel 'banyak' ini berguna untuk menghitung jumlah bunga yang dimasukkan ke dalam array Pita tadi).

Dalam program diatas, user akan diminta untuk menginputkan nama bunga, kemudian data ini akan diterima oleh array Pita kemudian akan dihitung menggunakan variabel 'banyak', dalam program ada percabangan dengan kondisi apabila user menginputkan kata 'selesai' atau 'SELESAI' maka program akan berhenti dan menampilkan output bunga yang sebelumnya telah di input user, namun apabila user menginputkan kata selain selesai atau SELESAI, maka program akan terus meminta user untuk meninputkan nama bunga.

2. 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.

- a. 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.
- b. 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"

func main() {
    var (
        a, b float32
    )
    for {
        fmt.Print("Masukan berat belanjaan di kedua kantong : ")

        fmt.Scanln(&a, &b)
        // if a == 9 || b == 9 {
        //     break
        // } else if a-b >= 9 {
        //     break
        // }
        if a < b {
            a, b = b, a
        }

        if a-b <= 9 {
            fmt.Println("Sepeda motor pak Andi akan oleng : false")
        } else if a+b >= 150 || a < 0 || b < 0 {
            break
        } else if a-b >= 9 {
            fmt.Println("Sepeda motor pak Andi akan oleng : true")
        }
    }
    fmt.Print("Proses Selesai.")
}
```

Screenshoot Output

a. Program a

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

b. Program b

```
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\Ungui2_M2.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
Proses Selesai.
```

Deskripsi Program

Program diatas adalah contoh program yang berguna untuk menghitung selisih berat pada 2 kantong belanjaan. Pada program diatas terdapat variabel a dan b dengan tipe data float32 dan juga looping infinite (looping yang tidak terbatas, hal ini terjadi apabila kondisi berat a dan b > 150kg atau a,b = negatif).

Pada program diatas juga terdapat algoritma IF, dimana jika a-b <= 9 maka output yang akan ditampilkan program adalah "Sepeda motor pak andi akan oleng = false, kondisi kedua dimana a+b >= 150 || a < 0 || b > 0 maka program akan berhenti jika kondisi kedua ini terpenuhi dan mengakhiri infinite looping diatas, adapun kondisi ketiga yaitu a-b >= 9 akan mengeluarkan output "Sepeda motor pak andi akan oleng : true".

Program akan meminta user untuk menginputkan nilai berat kantong belanjaan kiri (a) dan kanan(b), kemudian program akan mengecek inputan dari user, dan inputan tersebut akan melalui pengecekan terhadap 3 kondisi yang ada, setelah itu jika inputan user memenuhi kondisi kedua maka program akan berhenti.

3. Soal Studi Case

Buatlah sebuah Program dengan inputan sebuah bilangan K, kemudian menghitung nilai f(K) sesuai dengan persamaan fungsi berikut

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

Sourcecode

```
package main
```

```

import "fmt"

func main() {
    var K float64

    fmt.Print("Nilai : ")
    fmt.Scanln(&K)

    atas := ((4*K+2)*(4*K+2))
    bawah := ((4*K+1)*(4*K+3))

    f := atas / bawah

    fmt.Printf("Nilai f(k) = %.10f", f)
    //%.10f digunakan untuk membatasi panjang output hasil
}

```

Screenshoot Output

```

Proses Selesai.
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\Ungui3_M2.go"
Nilai : 100
Nilai f(k) = 1.0000061880
PS D:\Dokument\Semester 2\.vscode>

```

Deskripsi Program

Program diatas adalah contoh program yang digunakan untuk menghitung nilai fungsi berdasarkan inputan dari user. Pada program diatas user akan diminta untuk menginputkan nilai K, lalu nilai K ini akan di eksekusi oleh program sehingga program dapat menampilkan output hasil.

4. Soal Studi Case

Buatlah sebuah Program yang dimana program digunakan untuk menghitung berat (dalam gram), dan total berat dalam kg dan sisanya dalam gram. Biaya pengiriman memiliki beberapa kondisi,

- Rp. 10.000,- per kg.
- Jika > 500 gram, maka biaya kirim Rp. 5,- per gram
- Jika < 500 gram, maka biaya kirim Rp. 15,- per gram

- d. Sisa berat < 1000 gram akan digratiskan, apabila total berat > 10kg

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var (
        totalBerat int
        sisaHarga   int
        hargaPerKg = 10000
        totalBiaya int
    )

    fmt.Print("Masukkan berat persel (gram): ")
    fmt.Scanln(&totalBerat)

    kg := totalBerat / 1000
    gram := totalBerat % 1000
    fmt.Printf("Detail berat: %d kg + %d gram\n", kg, gram)

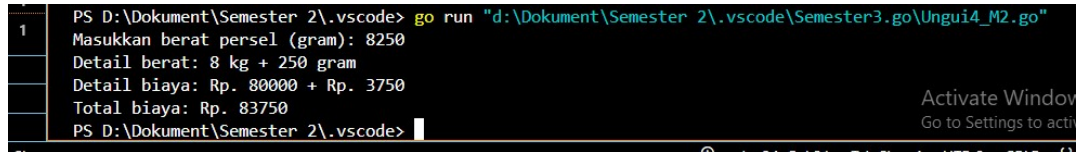
    biayaKg := kg * hargaPerKg

    if gram >= 500 {
        sisaHarga = gram * 5
    } else {
        sisaHarga = gram * 15
    }

    if totalBerat > 10000 && gram <= 1000 {
        sisaHarga = 0
    }

    fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayaKg,
sisaHarga)
    totalBiaya = biayaKg + sisaHarga
    fmt.Printf("Total biaya: Rp. %d\n", totalBiaya)
}
```

Screenshoot Output



```
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\Ungui4_M2.go"
1  Masukkan berat persel (gram): 8250
    Detail berat: 8 kg + 250 gram
    Detail biaya: Rp. 80000 + Rp. 3750
    Total biaya: Rp. 83750
    PS D:\Dokument\Semester 2\.vscode>
```

Deskripsi Program

Program diatas adalah contoh program yang berguna untuk mengetahui berat gram menjadi Kg, lalu program juga akan menentukan harga per kg dan gram nya. Kemudian program akan memberi diskon apabila terdapat pembelian lebih dari 10Kg.

5. Soal Studi Case

Buatlah sebuah Program yang menginputkan bilangan bulat b memiliki faktor bilangan $f > 0$

- Inputan bilangan bulan b dan $b > 1$, Program akan menampilkan faktor dari b
- Lanjutkan program a setelah menerima masukkan bilangan bulat $b > 0$. program akan menampilkan faktor dari b, dan kemudian program akan menentukan apakah b bilangan prima.

Sourcecode

```
package main

import "fmt"

func main() {

    var nilai int
    var faktor []int

    fmt.Print("Bilangan: ")
    fmt.Scanln(&nilai)
    if nilai > 0 {
        for i := 1; i <= nilai; i++ {
            if nilai%i == 0 {
                faktor = append(faktor, i)
                //append input nilai array
            }
        }
    }
}
```

```

    }
}
fmt.Print("Faktor: ")
for j := 0; j < len(faktor); j++ {

    fmt.Print(faktor[j], " ")
}
fmt.Print("\n")
if len(faktor) == 2 {
    //len adalah panjang array

    fmt.Print("Prima: true")
} else {
    fmt.Print("Prima: false")
}

}
}

```

Screenshoot Output

a. Program a

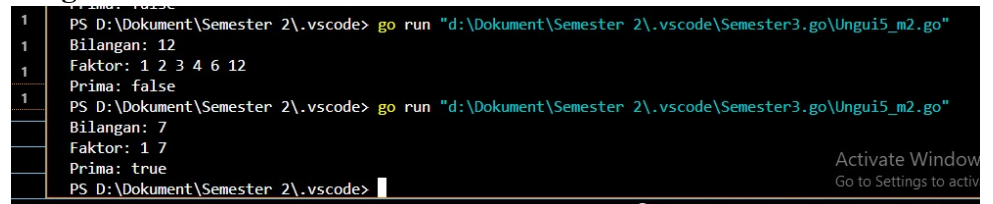


```

PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\Ungui5_m2.go"
Bilangan: 12
Faktor: 1 2 3 4 6 12

```

b. Program b



```

1 PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\Ungui5_m2.go"
1 Bilangan: 12
1 Faktor: 1 2 3 4 6 12
1 Prima: false
PS D:\Dokument\Semester 2\.vscode> go run "d:\Dokument\Semester 2\.vscode\Semester3.go\Ungui5_m2.go"
Bilangan: 7
Faktor: 1 7
Prima: true
PS D:\Dokument\Semester 2\.vscode>

```

Deskripsi Program

Program diatas adalah contoh program yang berguna untuk mengetahui nilai faktor dan apakah bilangan tersebut masuk ke dalam bilangan prima.

IV. DAFTAR PUSTAKA

- * [Golang Control Structures - W3Schools]
(https://www.w3schools.com/go/go_conditions.php)
- * [Effective Go: Control Structures]
(https://golang.org/doc/effective_go.html#control-structures)