

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II**

**Review Struktur Kontrol**



**Disusun Oleh :**

**Yoga Hogantara / 2311102153**

**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

Golang atau Go merupakan Bahasa pemrograman yang dikembangkan oleh Google yang dirancang untuk membangun aplikasi yang handal, berkesala dan berkinerja tinggi. Go sendiri sering digunakan untuk membangun aplikasi web, dan infrastruktur cloud.

Struktur Program Go mempunyai 2 komponen utama yaitu :

- Package main : yang berfungsi sebagai penanda file yang berisi program utama
- Func main() yang berisi kode utama dari program Go.

Contoh sebuah program

```
package main
import "fmt"

func main() {

var name string

fmt.Print("Masukkan nama Anda: ")

fmt.Scanln(&name)

fmt.Println("Halo," , name)

}
```

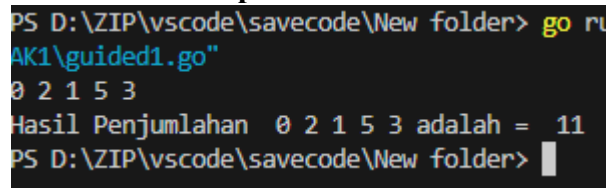
## II. GUIDED

### 1. Sourcecode

```
package main
import "fmt"
func main(){
var a, b, c, d, e int
var hasil int
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:\ZIP\vscode\savecode\New folder> go run
AK1\guided1.go
0 2 1 5 3
Hasil Penjumlahan 0 2 1 5 3 adalah = 11
PS D:\ZIP\vscode\savecode\New folder>
```

#### Deskripsi Program

Program tersebut dapat menjumlahkan sebuah inputan yang di input oleh user dengan mendeklarasikan variable a,b,c,d,e dengan tipe data integer yang di isi oleh user sehingga mengoutputkan hasil penjumlahan dari inputan yang di isi oleh user

### 2. 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, " ")

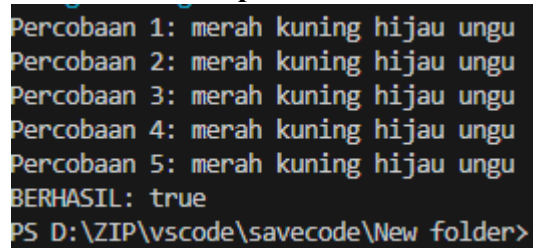
        // Mengecek apakah urutan warna sesuai
        for j := 0; j < 4; j++ {
            if colors[j] != correctOrder[j] {
                success = false
                break
            }
        }

        // 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



```

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
PS D:\ZIP\vscode\savecode\New folder>

```

### Deskripsi Program

Program tersebut mengecek user apakah inputan warna di masukan secara urut atau tidak dalam 5 kali percobaan. Dengan menggunakan bufio.NewReader dan strings.split untuk membaca input dari user sebanyak 5 kali perulangan dan memisah dan mengecek urutan warna yang di inputkan. Jika urutan warna benar maka akan mencetak BERHASIL: true, jika tidak maka mencetak BERHASIL: false.

### 3. Source code

```
package main

import "fmt"

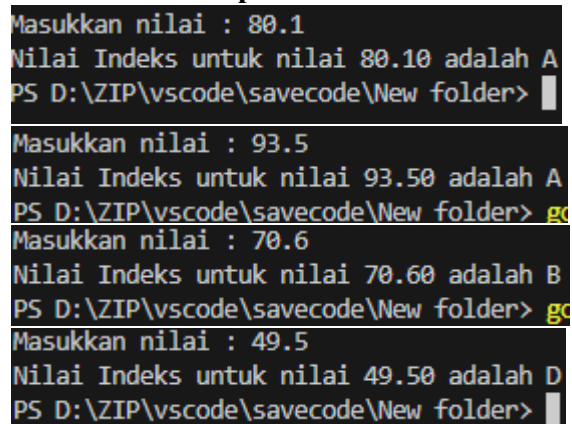
func main() {
    var nam float32
    var nmk string

    // 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 = "B"
    } else if nam > 57.5 {
        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



```
Masukkan nilai : 80.1
Nilai Indeks untuk nilai 80.10 adalah A
PS D:\ZIP\vscode\savecode\New folder>

Masukkan nilai : 93.5
Nilai Indeks untuk nilai 93.50 adalah A
PS D:\ZIP\vscode\savecode\New folder> gc

Masukkan nilai : 70.6
Nilai Indeks untuk nilai 70.60 adalah B
PS D:\ZIP\vscode\savecode\New folder> gc

Masukkan nilai : 49.5
Nilai Indeks untuk nilai 49.50 adalah D
PS D:\ZIP\vscode\savecode\New folder>
```

**Deskripsi Program**

Program tersebut berfungsi untuk menentukan nilai indeks dari inputan yang di isi oleh user dan sudah ditetapkan dalam program. Dengan menggunakan struktur percabanga untuk beberapa kondisi agar output an mengeluarkan hasil yang sesuai dengan yang sudah ditentukan.

### III. UNGUIDED

1.

#### Sourcecode

before

```
package main

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

func main() {
    var N int
    fmt.Print("N: ")
    fmt.Scan(&N)

    pita_153 := ""
    hitung_153 := 0

    reader := bufio.NewReader(os.Stdin)

    for y := 1; y <= N; y++ {
        fmt.Printf("bunga%d: ", y)
        input, _ := reader.ReadString('\n')
        input = strings.TrimSpace(input)

        // Tambahkan nama bunga ke pita
        if pita_153 == "" {
            pita_153 = input
        } else {
            pita_153 += " - " + input
        }
        hitung_153++
    }

    fmt.Printf("\nPita: " + pita_153)
}
```

after

```
package main

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

func main() {
    var N int
    fmt.Print("N: ")
    fmt.Scan(&N)

    pita_153 := ""
    hitung_153 := 0

    reader := bufio.NewReader(os.Stdin)

    for y := 1; y <= N; y++ {
        fmt.Printf("bunga%d: ", y)
        input, _ := reader.ReadString('\n')
        input = strings.TrimSpace(input)

        if strings.ToLower(input) == "SELESAI" {
            break
        }

        if pita_153 == "" {
            pita_153 = input
        } else {
            pita_153 += " - " + input
        }
        hitung_153++
    }

    fmt.Printf("\nPita: " + pita_153)
    fmt.Printf("\nbunga : %d\n", hitung_153)
}
```



## Screenshoot Output

### Before

```
N: 3
bunga1: kertas
bunga2: mawar
bunga3: tulip

Pita: kertas - mawar - tulip
```

### After

```
N: 3
bunga1: kertas
bunga2: mawar
bunga3: tulip

Pita: kertas - mawar - tulip
bunga : 3
|
```

## Deskripsi Program

Program tersebut memiliki perbedaan program awal belum di modifikasi atau di update. Pada program yang sudah di modifikasi, apabila user menginputkan SELESAI maka, output menampilkan isi pita beserta banyaknya bunga yang ada dalam pita. Program diatas menggunakan perulangan jika  $y \leq n$  maka user menginputkan nama sebanyak n.

## 2. Source code

### before

```
package main

import (
    "fmt"
)

func main() {
    for {
        var kantong1_153, kantong2_153 float64
        fmt.Print("Masukan berat belanjaan di kedua kantong: ")
        fmt.Scan(&kantong1_153, &kantong2_153)
        if kantong1_153 >= 9 || kantong2_153 >= 9 {
            fmt.Println("Proses selesai.")
            break
        }
    }
}
```

```
}  
}  
}
```

#### After

```
package main  
  
import (  
    "fmt"  
    "math"  
)  
  
func main() {  
    for {  
  
        var kantong1_153,kantong2_153 float64  
        fmt.Print("Masukkan berat belanjaan di kedua kantong: ")  
        fmt.Scan(&kantong1_153,&kantong2_153)  
  
        if kantong1_153 < 0 || kantong2_153 < 0 {  
            fmt.Println("Proses selesai.")  
            break  
        }  
  
        totalBerat := kantong1_153 + kantong2_153  
  
        if totalBerat > 150 {  
            fmt.Println("Proses selesai.")  
            break  
        }  
  
        selisih2 := math.Abs(kantong1_153 - kantong2_153)  
  
        olengco := selisih2 >= 9  
        fmt.Printf("Sepeda motor Pak Andi akan oleng: %t\n",  
olengco)  
    }  
}
```

## Screenshoot Output

before

```
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.
PS D:\ZIP\vscode\savecode\New folder> go run "d:\7
```

After

```
Masukkan berat belanjaan di kedua kantong: 5 10
Sepeda motor Pak Andi akan oleng: false
Masukkan berat belanjaan di kedua kantong: 55.6 70.2
Sepeda motor Pak Andi akan oleng: true
Masukkan berat belanjaan di kedua kantong: 72.3 66.9
Sepeda motor Pak Andi akan oleng: false
Masukkan berat belanjaan di kedua kantong: 59.5 98.7
Proses selesai.
PS D:\ZIP\vscode\savecode\New folder> █
```

## Deskripsi Program

Pada program pertama yang belum di modifikasi memungkinkan user untuk menginputkan berat belanjaan di kedua kantong, jika berat salah satu kantong memiliki berat lebih dari 9kg atau maka proses program akan berhenti berjalan, namun pada program kedua mengecek apakah kedua kantong belanjaan membuat pak andi oleng atau tidak. Dengan menggunakan perulangan yang didalamnya ada percabangan , sehingga program berjalan sampai kondisi terpenuhi.

### 3. Source code

```
package main

import (
    "fmt"
)

func f(k int) float64 {

    atas:= (4*float64(k) + 2) * (4*float64(k) + 2)
    bawah := (4*float64(k) + 1) * (4*float64(k) + 3)
    return atas / bawah
}

func main() {

    var k int
```

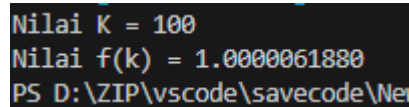
```

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

    hasil := f(k)
    fmt.Printf("Nilai f(k) = %.10f\n", hasil)
}

```

### Screenshot Output



```

Nilai K = 100
Nilai f(k) = 1.0000061880
PS D:\ZIP\vscode\savecode\Ne

```

### Deskripsi Program

Program di atas yang memuat persamaan  $f(k)$  yang nantinya user menginputkan nilai  $k$  sehingga program akan menghitung hasil dari nilai persamaan yang sudah ada.

#### 4. Source code

```

package main

import (
    "fmt"
)

func hitungbiaya(berat int) int {

    const biayaperkg = 10000
    const biayalebih = 5
    const biyakurang = 15

    kg := berat / 1000
    sisa153 := berat % 1000

    biaya := kg * biayaperkg

    if sisa153 >= 500 {
        biaya += sisa153 * biayalebih
    } else {

```

```

        biaya += sisa153 * biayakurang
    }

    if berat >= 10000 {
        biaya = kg * biayaperkg
    }

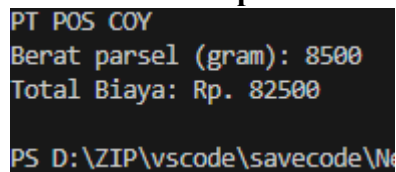
    return biaya
}

func main() {
    var berat int
    fmt.Printf("PT POS COY\n")
    fmt.Printf("Berat parsel (gram): ")
    fmt.Scan(&berat)
    biaya := hitungbiaya(berat)
    fmt.Printf("Total Biaya: Rp. %d\n\n", biaya)

}

```

### Screenshot Output



```

PT POS COY
Berat parsel (gram): 8500
Total Biaya: Rp. 82500
PS D:\ZIP\vscode\savecode\Ne

```

### Deskripsi Program

Program diatas berfungsi untuk menghitung total biaya dari pengiriman parsel berdasar berat. Jika pengiriman kurang dari 500 gram maka terkena tambahan biaya Rp 5 pergram namun jika kurang 500 gram maka tambahan biaya dikenakan Rp 15 pergram. Sisa berat kurang dari 1kg digratiskan biayanya jika total berat lebih 10kg. Dengan menggunakan percabangan pada program yang berfungsi untuk beberapa kondisi.

#### 5. Source code

before

```

package main

import "fmt"

func main() {
    var bilangan153 int
    fmt.Print("Bilangan: ")

```

```

    fmt.Scan(&bilangan153)
    if bilangan153 <= 1 {
        fmt.Println("LEBIH DARI 1 WOY.")
        return
    }
    fmt.Print("Faktor: ")
    for i := 1; i <= bilangan153; i++ {
        if bilangan153%i == 0 {
            fmt.Print(i, " ")
        }
    }
    fmt.Println()
}

```

After

```

package main

import "fmt"

func main() {
    var bilangan153 int
    fmt.Print("Bilangan: ")
    fmt.Scan(&bilangan153)

    if bilangan153 <= 1 {
        fmt.Println("LEBIH DARI 1 WOY.")
        return
    }

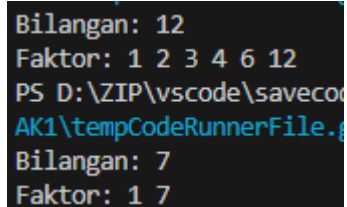
    fmt.Print("Faktor: ")
    var faktor int = 0
    for i := 1; i <= bilangan153; i++ {
        if bilangan153%i == 0 {
            fmt.Print(i, " ")
            faktor++
        }
    }
    fmt.Println()

    if faktor == 2 {
        fmt.Println("Prima: true")
    }
}

```

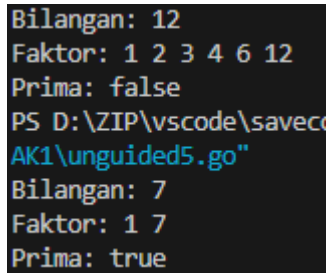
```
    } else {  
        fmt.Println("Prima: false")  
    }  
}
```

#### Screenshoot



```
Bilangan: 12  
Faktor: 1 2 3 4 6 12  
PS D:\ZIP\vscode\savecode  
AK1\tempCodeRunnerFile.g  
Bilangan: 7  
Faktor: 1 7
```

#### After



```
Bilangan: 12  
Faktor: 1 2 3 4 6 12  
Prima: false  
PS D:\ZIP\vscode\savecode  
AK1\unguided5.go"  
Bilangan: 7  
Faktor: 1 7  
Prima: true
```

#### Deskripsi program

Program diatas berfungsi untuk mencari faktor dari bilangan yang di inputkan oleh user. Pada program sebelum dimodifikasi hanya mencari faktornya saja namun pada pada program yang sudah di modifikasi ia mencari bilangan faktor dan prima saja. Dengan menggunakan perulangan untuk memeriksa semua bilangan 1 sampai bilangan yang di inputkan oleh user, dan menggunakan percabangan untuk kondisi apakah termasuk bilangan prima.