

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II  
REVIEW STRUKTUR KONTROL**



**Disusun Oleh :**

**Marsep Trianto Pakondo / 2311102251**

**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

### Struktur Kontrol dalam Golang

Struktur kontrol dalam pemrograman, termasuk Golang, adalah mekanisme yang digunakan untuk mengontrol alur eksekusi suatu program. Dengan struktur kontrol, kita dapat membuat keputusan, mengulang suatu blok kode, atau menjalankan kode berdasarkan kondisi tertentu.

**Golang menyediakan beberapa jenis struktur kontrol yang umum digunakan, yaitu:**

#### 1. Pernyataan Kondisional (Conditional Statements)

- **if:** Digunakan untuk mengeksekusi blok kode jika suatu kondisi bernilai benar.

```
if kondisi {
```

```
    // Kode yang akan dijalankan jika kondisi benar
```

```
}
```

- **if-else:** Digunakan untuk mengeksekusi blok kode yang berbeda berdasarkan nilai kebenaran suatu kondisi.

```
if kondisi {
```

```
    // Kode yang akan dijalankan jika kondisi benar
```

```
} else {
```

```
    // Kode yang akan dijalankan jika kondisi salah
```

```
}
```

- **else if:** Digunakan untuk menambahkan kondisi tambahan setelah if atau else.

```
if kondisi1 {
```

```
    // Kode jika kondisi1 benar
```

```
} else if kondisi2 {
```

```
// Kode jika kondisi1 salah dan kondisi2 benar  
} else {  
  
    // Kode jika kedua kondisi salah  
}
```

## 2. Perulangan (Loops)

- for: Perulangan yang paling umum digunakan dalam Golang. Fleksibel untuk berbagai jenis perulangan.

```
// Perulangan dengan kondisi  
for i := 0; i < 10; i++ {  
  
    fmt.Println(i)  
}  
  
// Perulangan tanpa kondisi (infinite loop)  
for {  
  
    // Kode yang akan terus diulang  
}
```

## II. GUIDED

### Soal Studi Case

XXXXXXXXXXXXXXXXXXXX

#### Sourcecode

```
package main

import "fmt"

func main() {
    var (
        satu, dua, tiga string
        temp string
    )

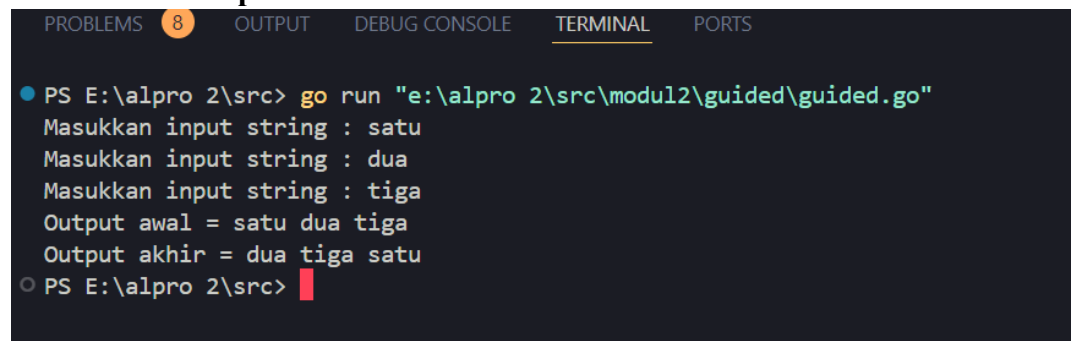
    fmt.Print("Masukkan input string : ")
    fmt.Scanln(&satu)
    fmt.Print("Masukkan input string : ")
    fmt.Scanln(&dua)
    fmt.Print("Masukkan input string : ")
    fmt.Scanln(&tiga)

    fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)

    temp = satu
    satu = dua
    dua = tiga
    tiga = temp

    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
}
```

#### Screenshoot Output



```
PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\guided\guided.go"
Masukkan input string : satu
Masukkan input string : dua
Masukkan input string : tiga
Output awal = satu dua tiga
Output akhir = dua tiga satu
PS E:\alpro 2\src>
```

## Deskripsi Program

*Program diatas digunakan untuk memindahkan suatu nilai dari tempat asalnya ke tempat sebelumnya, ibaratnya seperti turun ke lantai sebelumnya. Dan nilai yang tempatnya diawal akan dipindahkan ke akhir. Pertama buat struktur dasar golang dimana ada package main yang artinya didalam file tersebut adalah file program utama, kemudian import fmt digunakan untuk input/output, lalu buat fungsi utama untuk menjalankan code-code yang akan dibuat. Didalam fungsi utama tersebut awalnya mendeklarasikan sebuah variabel untuk menampung sebuah nilai yang bertipe data string. Kemudian membuat inputan untuk memsukkan nilai kedalam variabel yang sudah dibuat. Setelah itu print untuk sebagai output awal dan belum dipindahkan. Setelah itu pindahkan nilai dari variabel asalnya ke variabel sebelumnya, dan variabel awal pindah ke variabel akhir. Setelah itu print lagi untuk melihat hasil output akhir.*

## Sourcecode

```
package main

import "fmt"

func main() {
    var tahun int

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

    if (tahun%400 == 0) || (tahun%4 == 0 && tahun%100 !=
0) {
        fmt.Println("Kabisat :", true)
    } else {
        fmt.Println("Kabisat :", false)
    }
}
```

## Screenshoot Output

```
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\guided\guided1.go"
Masukkan Tahun : 2016
Kabisat : true
● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\guided\guided1.go"
Masukkan Tahun : 2000
Kabisat : true
● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\guided\guided1.go"
Masukkan Tahun : 2018
Kabisat : false
● PS E:\alpro 2\src> █
```

## Deskripsi Program

*Program diatas digunakan untuk menentukan tahun kabisat. Seperti biasa buat struktur golang seperti guided sebelumnya. Didalam fungsi utama tersebut dideklarasikan dahulu variabel yang bertipedata interger. Kemudian buat input/output unutm memasukkan tahun yang akan diperiksa apakah masuk kedalam tahun kabisat atau tidak. Lalu tahun tersebut akan diperiksa apabila angka dari tahun tersebut habis dibagi 400 sama dengan 0 atau angka dari tahun tersebut habis dibagi 4 sama dengan 0 dan habis dibagi 100 tidak sama dengan 0, maka tahun tersebut adalah tahun kabisat. Jika salah satu dari kedua kondisi tersebut tidak terpenuhi maka tahun tersebut bukan tahun kabisat.*

## Sourcecode

```
package main

import (
    "fmt"
    "math"
)

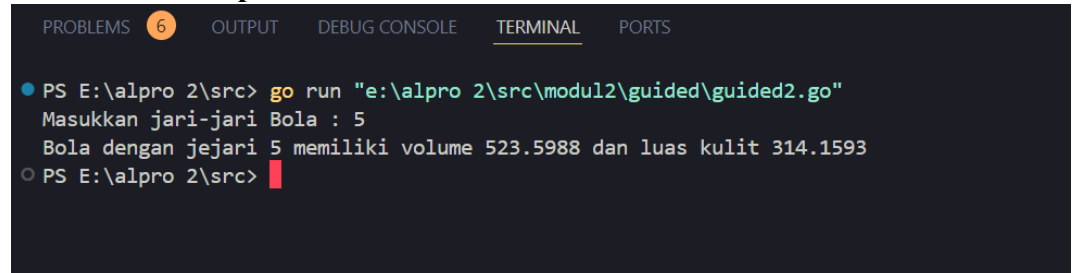
func main() {
    var r float64

    fmt.Print("Masukkan jari-jari Bola : ")
    fmt.Scanln(&r)

    volumebola := (4.0/3.0) * math.Pi * math.Pow(r, 3)
    luasbola := 4.0 * math.Pi * math.Pow(r, 2)

    fmt.Printf("Bola dengan jejari %v memiliki volume %.4f dan luas kulit %.4f", r, volumebola, luasbola)
}
```

## Screenshoot Output



```
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\guided\guided2.go"
Masukkan jari-jari Bola : 5
Bola dengan jejari 5 memiliki volume 523.5988 dan luas kulit 314.1593
PS E:\alpro 2\src>
```

## Deskripsi Program

*Program diatas digunakan untuk mengitung volume dan luas suatu bola. Seperti biasa buat struktur golang, nanum ada tambahan yaitu mengimport math untuk mengambil fungsi – fungsi matematika. Kemudian didalam fungsi utama tersebut dideklarasikan sebuah variabel untuk menampung jari jari yang bertipe float. Lalu buat inputan untuk memasukkan nilai jari-jari. Setelah menampung rumus untuk menentukan volume bola kedalam variabelnya dan juga untuk luas bola. Setelah itu print untuk melihat hasilnya yang akan bernilai desimal.*

### III. UNGUIDED

#### Soal Studi Case

XXXXXXXXXXXXXXXXXXXX

#### Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var celsius float32

    fmt.Print("\nMasukkan Temperatur Celsius : ")
    fmt.Scanln(&celsius)

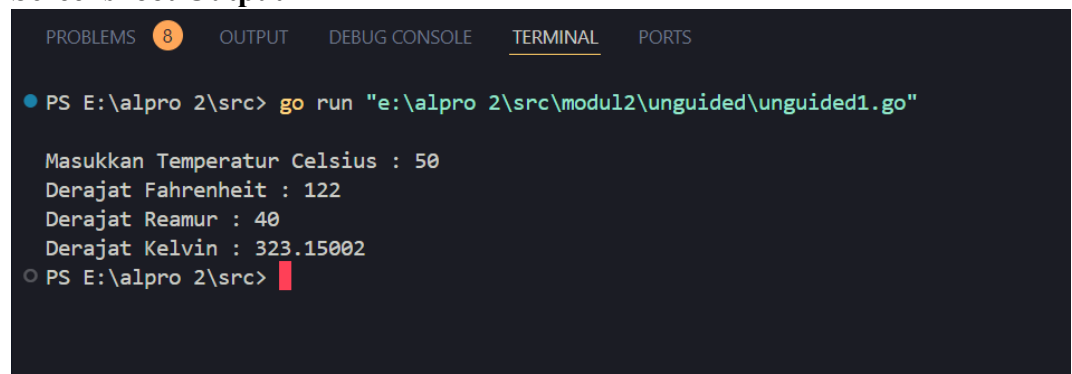
    fahrenheit := 32 + (celsius * 9/5)

    reamur := celsius * 4/5

    kelvin := (fahrenheit + 459.67) * 5/9

    fmt.Printf("Derajat Fahrenheit : %v\nDerajat Reamur : %v\nDerajat Kelvin : %v\n", fahrenheit, reamur, kelvin)
}
```

#### Screenshot Output



```
PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided1.go"

Masukkan Temperatur Celsius : 50
Derajat Fahrenheit : 122
Derajat Reamur : 40
Derajat Kelvin : 323.15002
PS E:\alpro 2\src>
```

#### Deskripsi Program

Program diatas digunakan untuk menentukan konversi dari tempertaur celsius ke beberapa temperatur. Seperti bisa membuat struktur golang yang didalam fungsi utama dideklarasikan sebuah variabel untuk menampung nilai tempertatur celsius yang akan diinputkan oleh user. Setelah buat inputan, inisialisasikan sebuah variabel untuk menampung



*sebuah rumus untuk mengkonversikan temperatur celsius ke temperatur sesuai yang diinginkan (satu variabel satu temperatur). Setelah itu print untuk melihat hasil konversinya.*

### Sourcecode

```
package main

import (
    "fmt"
)

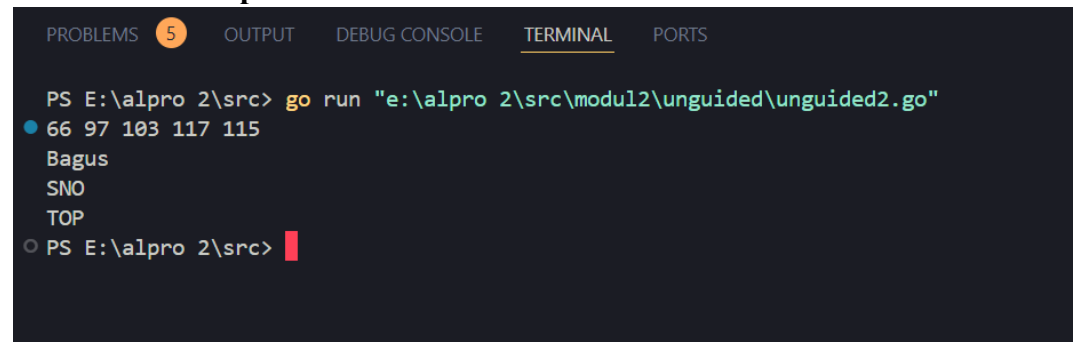
func main() {
    var a, b, c, d, e int
    var x, y, z rune

    fmt.Scanf("%d %d %d %d %d", &a, &b, &c, &d, &e)
    fmt.Printf("%c%c%c%c%c\n", a, b, c, d, e)

    fmt.Scanf("\n")

    fmt.Scanf("%c%c%c", &x, &y, &z)
    fmt.Printf("%c%c%c\n", x+1, y+1, z+1)
}
```

### Screenshoot Output



```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided2.go"
66 97 103 117 115
Bagus
SNO
TOP
PS E:\alpro 2\src>
```

### Deskripsi Program

*Program diatas digunakan untuk membaca lima nilai integer dari input pengguna, mengonversi nilai-nilai tersebut menjadi karakter berdasarkan kode ASCII, lalu dicetak. Setelah itu, program membaca tiga karakter dari input, lalu menambahkan nilai 1 pada setiap karakter (menggeser ke karakter berikutnya dalam urutan ASCII), lalu dicencetak hasilnya.*

## Sourcecode

```
package main

import "fmt"

func main() {
    var a, b, c, d string
    var berhasil bool = true

    for i := 1; i <= 5; i++ {
        fmt.Printf("Percobaan %v : ", i)
        fmt.Scanln(&a, &b, &c, &d)

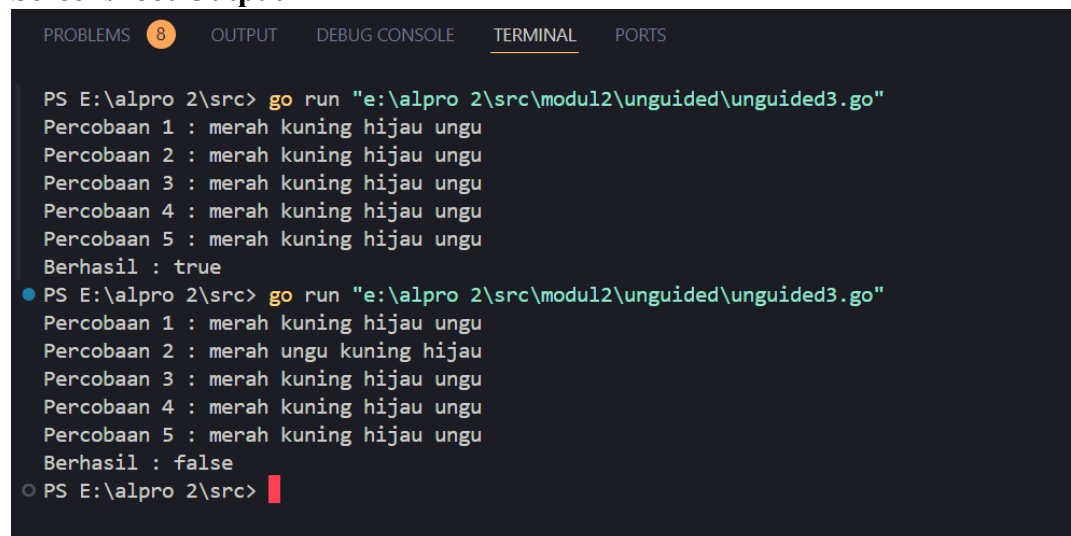
        if a == "merah" && b == "kuning" && c ==
            "hijau" && d == "ungu" {

            } else {
                berhasil = false
            }

        }

    if berhasil {
        fmt.Printf("Berhasil : %v", berhasil)
    } else {
        fmt.Printf("Berhasil : %v", berhasil)
    }
}
```

## Screenshot Output



```
PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided3.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
● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided3.go"
Percobaan 1 : merah kuning hijau ungu
Percobaan 2 : merah ungu kuning hijau
Percobaan 3 : merah kuning hijau ungu
Percobaan 4 : merah kuning hijau ungu
Percobaan 5 : merah kuning hijau ungu
Berhasil : false
○ PS E:\alpro 2\src> █
```

## Deskripsi Program

*Program diatas digunakan untuk menentukan apakah inputan warna dari lima-limanya apakah sudah sesuai dengan lima warna yang sudah di tentukan. Jika percobaan pertama sampai kelima sesuai tanpa ada yang terlewatkan atau salah, maka akan mengeluarkan true. Tetapi jika ada yang salah maka false.*

### Sourcecode

```
package main

import "fmt"

func main() {
    var a, pita string
    i := 0
    for a != "SELESAI" {
        fmt.Printf("\nBunga %v : ", i+1)
        fmt.Scanln(&a)

        if a != "SELESAI" {
            a += " - "
            pita += a
            i++
        }

    }
    fmt.Printf("\nPita   : %v\n", pita)
    fmt.Printf("Bunga  : %v\n", i)
}
```

### Screenshoot Output

```
PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided4.go"

Bunga 1 : Kertas

Bunga 2 : Mawar

Bunga 3 : Tulip

Bunga 4 : SELESAI

Pita : Kertas - Mawar - Tulip -
Bunga : 3
● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided4.go"

Bunga 1 : SELESAI

Pita :
Bunga : 0
○ PS E:\alpro 2\src>
```

## Deskripsi Program

*Program diatas digunakan untuk mengumpulkan nama-nama bunga dari inputan user. Jadi didalam fungsi utama program akan selesai jika user menginputkan nama bunganya SELESAI. Tetapi jika nama lain program akan berjalan terus sambil meminta inputan. Setelah itu akan ditampilkan kumpulan nama-nama bunga yang sudah dimasukkan beserta jumlah inputannya ada berapa.*

## Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var a, b, selisih float32
    var oleng, selesai bool
    var d float32

    for !selesai {
        fmt.Printf("masukkan berat belanjaan di kedua
kantong : ")
        fmt.Scanln(&a, &b)
```

```

        d = a + b

        if d <= 0 || d >= 150 {
            fmt.Println("Proses selesai.")
            selesai = true
        } else {
            selisih = a - b

            if selisih >= 9 || selisih <= -9 {
                oleng = true
            } else {
                oleng = false
            }
            fmt.Printf("Sepedah motor pak Andi akan
oleng : %v\n", oleng)
        }
    }
}

```

## Screenshoot Output

```

PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided5.go"
● masukkan berat belanjaan di kedua kantong : 5 10
Sepedah motor pak Andi akan oleng : false
masukkan berat belanjaan di kedua kantong : 55.6 70.2
Sepedah motor pak Andi akan oleng : true
masukkan berat belanjaan di kedua kantong : 72.3 66.9
Sepedah motor pak Andi akan oleng : false
masukkan berat belanjaan di kedua kantong : 59.5 98.7
Proses selesai.
○ PS E:\alpro 2\src>

```

## Deskripsi Program

*Program diatas digunakan untuk menentukan nilai true jika selisi kedua inputan user lebih atau sama dengan 9. Didalam fungsi utama apa bila kedua inputan user jika ditotalkan mendapatkan berat lebih dari 150 atau beratnya negatif maka program akan langsung selesai. Tapi jika tidak maka program akan lanjut memeriksa apakah selisih kedua inputan lebih atau sama dengan 9 maka akan menghasilkan true atau motor oleng.*

## Sourcecode

```

package main

import (
    "fmt"

```

```

    "math"
)

func main() {
    var k int
    var hasil float64 = 1.0

    fmt.Printf("Nilai K = ")
    fmt.Scanln(&k)

    for i := 0; i < k; i++ {
        pembilang := math.Pow(float64(4*i + 2), 2)
        penyebut := float64((4*i + 1) * (4*i + 3))

        hasil *= pembilang / penyebut
    }

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

```

## Screenshoot Output

```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided6.go"
Nilai K = 10
Nilai akar 2 = 1.4054086752
● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided6.go"
Nilai K = 100
Nilai akar 2 = 1.4133299615
● PS E:\alpro 2\src> go run "e:\alpro 2\src\modul2\unguided\unguided6.go"
Nilai K = 1000
Nilai akar 2 = 1.4141251768
○ PS E:\alpro 2\src>

```

## Deskripsi Program

*Program diatas digunakan untuk menghitung hampiran akar 2 dari inputan user. Program ini mengimport math untuk menghitung pangkat. Rumus yang dibuat akan dihitung berulang kali sebanyak inputan user. Dan untuk nilai variabel rumusnya akan dimasukkan nilai dari hasil ingkremen i, setelah dihitung, hasilnya akan dikali dengan variabel hasil sampai sebanyak nilai inputan. Lalu tinggal di print hasilnya, angka dibelakang koma disetel sebanyak 10.*