

**LAPORAN PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN 1**  
**MODUL 3**  
**“TIPE DATA & VARIABEL”**



**DISUSUN OLEH:**  
**RAIHAN ADI ARBA**  
**103112400071**  
**S1 IF-12-01**  
**DOSEN:**  
**Yohani Setiya Rafika Nur, M. Kom.**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2024/2025**

## DASAR TEORI

I/O (Input/Output) merupakan konsep penting dalam pemrograman yang mengatur bagaimana data masuk dan keluar dari program. Dalam Golang, I/O dilakukan melalui paket seperti `fmt` untuk operasi dasar seperti membaca input dari pengguna dan menampilkan output ke layar.

Tipe data dalam Golang menentukan jenis data yang dapat disimpan dan diolah oleh program. Seperti kebanyakan bahasa pemrograman, Golang memiliki dua kategori utama tipe data:

1. **Tipe Data Primitif:** Ini adalah tipe dasar yang langsung disediakan oleh bahasa tanpa perlu definisi tambahan. Golang menyediakan tipe data primitif seperti:
  - **Numerik:** Termasuk integer (`int`, `int8`, `int16`, `int32`, `int64`), unsigned integer (`uint`, `uint8`, `uint16`, `uint32`, `uint64`), dan floating point (`float32`, `float64`).
  - **Boolean:** Merepresentasikan nilai kebenaran (`true` atau `false`).
  - **String:** Untuk menyimpan teks, yang diwakili oleh kumpulan karakter.
2. **Tipe Data Non-Primitif atau Tipe Data Referensi:** Tipe data ini lebih kompleks dan dibuat sesuai dengan kebutuhan pengembang. Beberapa tipe data non-primitif di Golang adalah:
  - **Array:** Kumpulan data tipe sama yang disimpan dalam urutan tertentu.
  - **Slice:** Versi lebih fleksibel dari array yang ukurannya bisa berubah.
  - **Map:** Koleksi pasangan kunci-nilai dinamis.
  - **Struct:** Kumpulan berbagai tipe data yang didefinisikan sebagai satu unit.

**Variabel** adalah entitas yang digunakan untuk menyimpan data dalam program. Di Golang, variabel dideklarasikan dengan menyebutkan tipe data, diikuti dengan nama variabel. Contoh deklarasi variabel di Golang adalah `var angka int`, yang menunjukkan bahwa variabel `angka` akan menyimpan data bertipe integer.

**Konstanta** di Golang adalah variabel yang nilainya tetap yang tidak berubah selama waktu eksekusi program. Konstanta dideklarasikan menggunakan kata kunci `const`, dan nilai dari konstanta ini harus ditetapkan pada saat deklarasi. Penggunaan konstanta di Golang mirip dengan penggunaannya di bahasa pemrograman lain, membantu dalam menjaga integritas dan keamanan nilai-nilai yang tidak boleh berubah selama eksekusi program. Contoh deklarasi konstanta adalah `const Pi = 3.14159`.

Dengan memahami I/O, tipe data, variabel, dan konstanta, pemrogram dapat mengorganisir dan mengoperasikan data secara efektif dalam pengembangan perangkat lunak menggunakan Golang, memungkinkan penanganan data yang lebih efisien dan program yang lebih rapi dan mudah dipelihara.

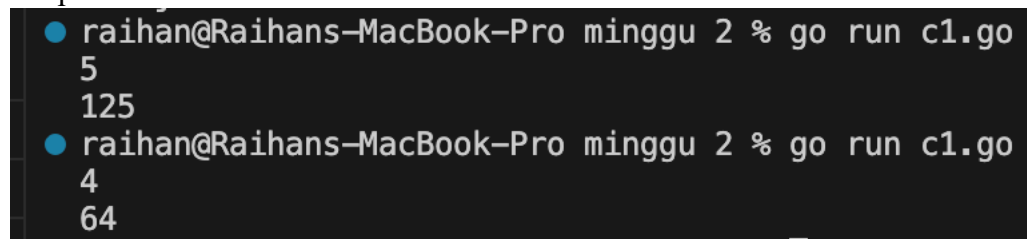
## A. GUIDED

1. Buatlah program dalam Bahasa Go untuk menghitung volume kubus berdasarkan panjang sisinya.

Source code :

```
package main
import "fmt"
func main() {
    var sisi, volume float64
    fmt.Scan(&sisi)
    volume = sisi * sisi * sisi
    fmt.Println(volume)
}
```

Output :



```
raihan@Raihans-MacBook-Pro minggu 2 % go run c1.go
5
125
raihan@Raihans-MacBook-Pro minggu 2 % go run c1.go
4
64
```

Deskripsi :

Program ini meminta input berupa panjang sisi kubus, kemudian menghitung volume kubus berdasarkan rumus  $\text{volume} = \text{sisi}^3$ . Setelah perhitungan, program menampilkan hasil volume kubus tersebut. Ini menggambarkan penggunaan input, output, dan operasi aritmatika sederhana pada variabel dalam Go.

2. Buatlah program dalam Bahasa Go yang digunakan untuk menghitung luas segitiga, apabila diketahui panjang alas dan tinggi dari segitiga.

Source code :

```
package main

import "fmt"

func main() {
    var alas, tinggi, luas float64
    fmt.Scan(&alas, &tinggi)
    luas = 0.5 * alas * tinggi
    fmt.Println(luas)
}
```

Output :

```

● raihan@Raihans-MacBook-Pro minggu 2 % go run c2.go
8 5
20
● raihan@Raihans-MacBook-Pro minggu 2 % go run c2.go
6 7
21
● raihan@Raihans-MacBook-Pro minggu 2 % go run c2.go
12 15
90

```

Deskripsi :

Program ini dirancang untuk menghitung luas segitiga. Pengguna diminta memasukkan dua nilai, yaitu panjang alas dan tinggi segitiga. Program kemudian menghitung luas segitiga menggunakan rumus  $\text{luas} = 0.5 * \text{alas} * \text{tinggi}$  dan menampilkan hasilnya. Penggunaan fungsi `Scan` memungkinkan program untuk menerima input nilai alas dan tinggi secara langsung dari pengguna melalui command line, dan `Println` digunakan untuk menampilkan hasil perhitungan.

3. Sebuah program digunakan untuk menghitung konversi mata uang dari IDR ke Dolar US dengan Kurs 15,000 IDR / USD.

Source code:

```

package main

import "fmt"

func main() {
    var rupiah, dolar int
    fmt.Scan(&rupiah)
    dolar = rupiah / 15000
    fmt.Println(dolar)
}

```

Output :

```

● raihan@Raihans-MacBook-Pro minggu 2 % go run c3.go
15000
1
● raihan@Raihans-MacBook-Pro minggu 2 % go run c3.go
75000
5
● raihan@Raihans-MacBook-Pro minggu 2 % go run c3.go
65000
4

```

Deskripsi :

Program ini berfungsi untuk mengkonversi mata uang Rupiah ke Dolar Amerika. Pengguna diminta untuk memasukkan jumlah uang dalam Rupiah, dan program kemudian menghitung jumlah yang setara dalam Dolar Amerika dengan kurs konversi

1 Dolar = 15,000 Rupiah. Program menggunakan operasi pembagian untuk mendapatkan nilai Dolar dari Rupiah yang diinput dan menampilkan hasilnya.

## B. UNGUIDED

### 1. Latihan 1

Source Code:

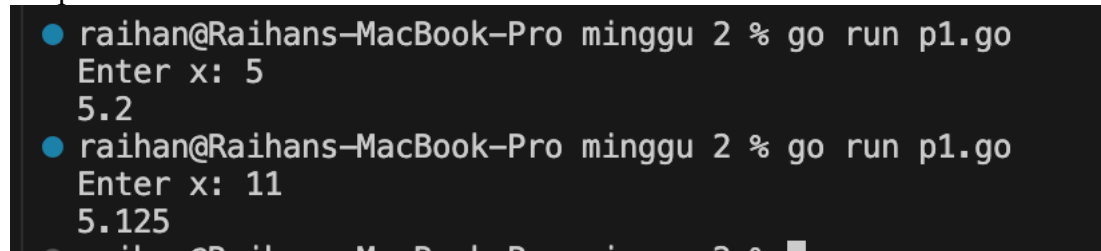
```
package main

import (
    "fmt"
)

func main() {
    var x float64
    fmt.Print("Enter x: ")
    fmt.Scan(&x)

    fx := 2/(x+5) + 5
    fmt.Println(fx)
}
```

Output :



```
● raihan@Raihans-MacBook-Pro minggu 2 % go run p1.go
Enter x: 5
5.2
● raihan@Raihans-MacBook-Pro minggu 2 % go run p1.go
Enter x: 11
5.125
```

Penjelasan Program:

Program untuk menghitung nilai fungsi matematika  $fx := 2/(x+5) + 5$  Awalnya, program meminta pengguna untuk memasukkan nilai x melalui prompt "Enter x: ". Setelah nilai x dimasukkan, program menghitung dan menampilkan hasil fungsi tersebut. Operasi ini melibatkan pembagian dan penambahan untuk mendapatkan  $f(x)$ , dan hasilnya langsung ditampilkan ke pengguna. Program ini menggabungkan konsep input dan output, serta manipulasi aritmetika dasar, yang memberikan pengalaman interaktif dalam mengaplikasikan matematika dalam pemrograman.

### 2. Latihan 2

Source Code:

```
package main

import "fmt"

func main() {
```

```

var volume, luas, r float64
fmt.Print("Jari Jari = ")
fmt.Scan(&r)
luas = 4.0 * 3.1415926535 * r * r
volume = (4.0 / 3.0) * 3.1415926535 * r * r * r
}

```

Output:

```

● raihan@Raihans-MacBook-Pro minggu 2 % go run p2.go
Jari Jari = 5
Bola dengan jejari 5 memiliki volume 523.5988 dan luas kulit 314.1593

```

Deskripsi Program:

Program ini dibuat untuk menghitung luas permukaan dan volume bola berdasarkan radius yang dimasukkan pengguna. Dalam implementasinya di Go, program terlebih dahulu meminta pengguna memasukkan nilai jari-jari (radius) bola melalui prompt "Jari Jari = ". Setelah nilai jari-jari diterima, program melakukan perhitungan menggunakan rumus matematika untuk luas permukaan bola,  $S = 4 \times \pi \times r^2$ , dan volume bola,  $V = \frac{4}{3} \times \pi \times r^3$ , dengan menggunakan nilai  $\pi = 3.1415926535$ .

### 3. Latihan 3

Source Code:

```

package main

import "fmt"

func main() {
    var tahun int
    var ckabisat bool

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

    if tahun%100 == 0 {
        ckabisat = tahun%400 == 0
    } else {
        ckabisat = tahun%4 == 0
    }

    fmt.Printf("Kabisat: %t\n", ckabisat)
}

```

Output:

```

• raihan@Raihans-MacBook-Pro minggu 2 % go run p3.go
Tahun: 2016
Kabisat: true
• raihan@Raihans-MacBook-Pro minggu 2 % go run p3.go
Tahun: 2029
Kabisat: false
• raihan@Raihans-MacBook-Pro minggu 2 % go run p3.go
Tahun: 2020
Kabisat: true
• raihan@Raihans-MacBook-Pro minggu 2 % go run p3.go
Tahun: 1995
Kabisat: false

```

#### Deskripsi Program:

Program ini bertujuan untuk menentukan apakah suatu tahun adalah tahun kabisat. Pengguna diminta untuk memasukkan tahun melalui prompt "Tahun: ". Setelah tahun dimasukkan, program akan melakukan pengecekan:

- Jika tahun tersebut habis dibagi 100, maka harus juga habis dibagi 400 untuk diklasifikasikan sebagai tahun kabisat.
- Jika tahun tersebut tidak habis dibagi 100, maka cukup dengan habis dibagi 4 untuk diklasifikasikan sebagai tahun kabisat.

Hasil dari pengecekan ini kemudian ditampilkan dalam format "Kabisat: true" atau "Kabisat: false" tergantung dari hasil pengecekan.

#### 4. Latihan 4

Source Code:

```

package main

import "fmt"

type Hitung struct {
    tcelcius, dfahrenheit, dreamur, dkelvin, hkelvin, hreamur, hcelsius, hfahrenheit
float64
}

func main() {
    var t Hitung
    fmt.Print("Temperatur Celsius: ")
    fmt.Scanln(&t.tcelcius)

    // Celsius -> Fahrenheit
    t.hfahrenheit = (t.tcelcius * 9 / 5) + 32

    // Celsius -> Reamur
    t.hreamur = t.tcelcius * 4 / 5

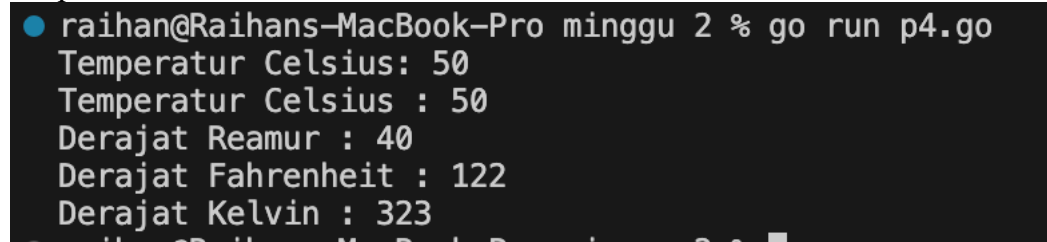
    // Celsius -> Kelvin
    t.hkelvin = t.tcelcius + 273.15

    fmt.Printf("Temperatur Celsius : %.0f\n", t.tcelcius)
    fmt.Printf("Derajat Reamur : %.0f\n", t.hreamur)
}

```

```
    fmt.Printf("Derajat Fahrenheit : %.0f\n", t.hfahrenheit)
    fmt.Printf("Derajat Kelvin : %.0f\n", t.hkelvin)
}
```

Output:

A terminal window screenshot showing the execution of a Go program. The prompt is 'raihaan@Raihans-MacBook-Pro minggu 2 %' and the command is 'go run p4.go'. The output consists of five lines: 'Temperatur Celsius: 50', 'Temperatur Celsius : 50', 'Derajat Reamur : 40', 'Derajat Fahrenheit : 122', and 'Derajat Kelvin : 323'.

```
raihaan@Raihans-MacBook-Pro minggu 2 % go run p4.go
Temperatur Celsius: 50
Temperatur Celsius : 50
Derajat Reamur : 40
Derajat Fahrenheit : 122
Derajat Kelvin : 323
```

Deskripsi Program:

Program ini mengkonversi suhu dari Celsius ke unit suhu lainnya, yaitu Fahrenheit, Reamur, dan Kelvin. Pengguna diminta untuk memasukkan suhu dalam Celsius melalui prompt "Temperatur Celsius: ". Setelah input diterima, program melakukan tiga konversi yaitu:

- Celsius ke Fahrenheit:
- Celsius ke Reamur:
- Celsius ke Kelvin:

Hasil dari ketiga konversi ini ditampilkan dalam format yang mudah dibaca, dengan satu desimal untuk presisi. Program ini menggunakan struktur data Hitung untuk menyimpan suhu awal dan hasil konversi, memastikan kode lebih rapi dan terorganisir.

## DAFTAR PUSTAKA

Prayogo, N. A. (2021). *Dasar Pemrograman Go. Ebook*



