

**LAPORAN PRAKTIKUM ALGORITMA**

**DAN PEMROGRAMAN 1**

**MODUL 11**

**SWITCH-CASE**



**Disusun oleh:**

**Mohamad Naufal Mubarok**

**109082500128**

**S1IF-13-02**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## LATIHAN KELAS – GUIDED

### 1. Guided 1

#### Source Code

```
package main

import (
    "fmt"
)

func main() {
    var jam24,jam12 int
    var ampm string

    fmt.Print("Masukkan jam : ")
    fmt.Scan(&jam24)

    switch {
    case jam24 == 0:
        jam12 = 12
        ampm = "AM"
    case jam24 == 12:
        jam12 = 12
        ampm = "PM"
    case jam24 >= 1 && jam24 <= 11:
        jam12 = jam24
        ampm = "AM"
    case jam24 >= 13 && jam24 <= 23:
```

```
jam12 = jam24 - 12

ampm = "PM"

default:

    fmt.Println("invalid, Input harus Jam 0-23")

    return

}

fmt.Printf("%d %s\n", jam12, ampm)

}
```

## Screenshot program

The screenshot shows a Go development environment with the following details:

- File Explorer:** Shows a project structure under "PRAKTIKUM-ALPRO1 - Latihan".
- Code Editor:** Displays a Go file named "switch guided 1.go" containing the following code:

```
switch guided 1 > 1guided1.go > main
4   |
5   )
6
Windsurf: Refactor | Explain | Generate GoDoc | X
7 func main() {
8     var jam24,jam12 int
9     var ampm string
10
11    fmt.Println("Masukkan jam : ")
12    fmt.Scan(&jam24)
13
14    switch {
15        case jam24 == 0:
16            jam12 = 12
17            ampm = "AM"
18        case jam24 == 12:
19            jam12 = 12
20            ampm = "PM"
21        case jam24 >= 1 && jam24 <= 11:
22            jam12 = jam24
23            ampm = "AM"
24        case jam24 >= 13 && jam24 <= 23:
25            jam12 = jam24 - 12
26            ampm = "PM"
27        default:
28            fmt.Println("invalid, Input harus Jam 0-23")
29            return
30    }
31}
```

- Terminal:** Shows the command "go run c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan> 1guided1.go" and its output:

```
PS C:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan> go run "c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan> 1guided1.go"
Masukkan jam : 13
1 PM
PS C:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan> go run "c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan> 1guided1.go"
Masukkan jam : 12 AM
PS C:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan>
```
- Modal Window:** A "switch guided" window displays the user's input and the converted time.

## Deskripsi program

Program tersebut merupakan kode Go yang berfungsi untuk mengonversi format jam berbasis 24 jam menjadi format 12 jam lengkap dengan penanda AM dan PM. Program dimulai dengan meminta pengguna memasukkan sebuah nilai jam, kemudian nilai tersebut diproses menggunakan struktur *switch* untuk menentukan kategori waktunya.

## 2. Guided 2

### Source Code

```
package main

import (
    "fmt"
)

func main() {
    var tanaman string
    fmt.Print("Masukan Tanaman : ")
    fmt.Scan(&tanaman)

    switch tanaman{
        case "nepenthes", "drosea":
            fmt.Println("Termasuk Tanaman Karnivora")
            fmt.Println("asli indonesia")

        case "venus", "sarracenia":
            fmt.Println("Termasuk Tanaman Karnivora")
            fmt.Println("Bukab asli indonesia")

        default :
            fmt.Println("Bukan Tanaman karnivora")
    }
}
```

## Screenshot program

The screenshot shows a Go code editor interface with the following code:

```
package main
import(
    "fmt"
)
func main() {
    var tanaman string
    fmt.Println("Masukan Tanaman : ")
    fmt.Scan(&tanaman)
    switch tanaman{
        case "nepenthes", "drossea":
            fmt.Println("Termasuk Tanaman Karnivora")
            fmt.Println("Asli Indonesia")
        case "venus", "sarracenia":
            fmt.Println("Termasuk Tanaman Karnivora")
            fmt.Println("Bukan Asli Indonesia")
        default :
            fmt.Println("Bukan Tanaman Karnivora")
    }
}
```

The terminal window below shows the output of running the program:

```
Termasuk Tanaman Karnivora
Asli Indonesia
Masukan Tanaman : venus
Termasuk Tanaman Karnivora
Bukan Asli Indonesia
Masukan Tanaman : sarracenia
Bukan Tanaman Karnivora
```

## Deskripsi program

Program tersebut membaca nama tanaman dari pengguna lalu mengelompokkannya menggunakan `switch`. Jika nama yang dimasukkan adalah nepenthes atau drosera, program menampilkan bahwa tanaman itu karnivora dan asli Indonesia. Jika yang dimasukkan venus atau sarracenia, program menyatakan bahwa tanaman tersebut karnivora tetapi bukan asli Indonesia. Jika tidak cocok dengan semua pilihan, program menampilkan bahwa tanaman tersebut bukan tanaman karnivora.

### Guided 3

#### Source Code

```
package main

import "fmt"

func main() {
    var kendaraan string
    var durasi int
    var tarif int

    fmt.Println("Masukkan jenis kendaraan
(Motor/Mobil/Truk): ")

    fmt.Scan(&kendaraan)

    fmt.Println("Masukkan durasi parkir (dalam jam): ")

    fmt.Scan(&durasi)

    switch {
        case kendaraan == "Motor" && durasi >= 1 && durasi <=
2:
            tarif = 7000
        case kendaraan == "Motor" && durasi > 2:
            tarif = 9000
        case kendaraan == "Mobil" && durasi >= 1 && durasi <=
2:
            tarif = 15000
        case kendaraan == "Mobil" && durasi > 2:
            tarif = 20000
        case kendaraan == "Truk" && durasi >= 1 && durasi <=
2:
            tarif = 25000
        case kendaraan == "Truk" && durasi > 2:
            tarif = 35000
    }
}
```

```
default:  
    fmt.Println("Data Tidak valid")  
}  
  
fmt.Printf("Tarif Parkir: Rp %d\n", tarif)  
}
```

## Screenshot program

The screenshot shows a Go code editor interface with the following code:

```
func main() {
    var kendaraan string
    var durasi int
    var tarif int

    fmt.Print("Masukkan jenis kendaraan (Motor/Mobil/Truk): ")
    fmt.Scan(&kendaraan)
    fmt.Print("Masukkan durasi parkir : ")
    fmt.Scan(&durasi)

    switch {
        case kendaraan == "Motor" && durasi >= 1 && durasi <= 2:
            | tarif = 2000
        case kendaraan == "Motor" && durasi > 2:
            | tarif = 5000
        case kendaraan == "Mobil" && durasi >= 1 && durasi <= 2:
            | tarif = 10000
        case kendaraan == "Mobil" && durasi > 2:
            | tarif = 20000
        case kendaraan == "Truk" && durasi >= 1 && durasi <= 2:
            | tarif = 25000
        case kendaraan == "Truk" && durasi > 2:
            | tarif = 35000
        default:
            | fmt.Println("Data Tidak valid")
    }
    fmt.Printf("Tarif Parkir: Rp %d\n", tarif)
}
```

The terminal window shows the execution of the program:

```
PS C:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan> go run "c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan\switch guided3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir : 1
Tarif Parkir: Rp 25000
PS C:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan> go run "c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan\switch guided3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): sepeda
Masukkan durasi parkir : 2
Data Tidak valid
Tarif Parkir: Rp 0
PS C:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan>
```

## Deskripsi program

Program tersebut menghitung tarif parkir berdasarkan jenis kendaraan dan durasi parkir. Pengguna memasukkan jenis kendaraan serta lama parkir, lalu `switch` menentukan tarifnya: motor Rp 2.000 ( $\leq 2$  jam) atau Rp 5.000 ( $> 2$  jam), mobil Rp 5.000 ( $\leq 2$  jam) atau Rp 10.000 ( $> 2$  jam), dan truk Rp 20.000 ( $\leq 2$  jam) atau Rp 35.000 ( $> 2$  jam). Jika jenis kendaraan tidak sesuai, program menampilkan bahwa data tidak valid. Program kemudian mencetak tarif parkir yang harus dibayar.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import (
    "fmt"
)

func main() {
    var pH float64

    fmt.Print("Masukkan kadar pH: ")

    fmt.Scan(&pH)

    switch {
    case pH < 0 || pH > 14:
        fmt.Println("Nilai pH tidak valid. Nilai pH harus
antara 0 dan 14.")

    case pH >= 6.5 && pH <= 8.6:
        fmt.Println("Air layak minum")

    default:
        fmt.Println("Air tidak layak minum")
    }
}
```

## Screenshot program

The screenshot shows a Go code editor interface with the following details:

- File Explorer:** Shows a project structure under "PRAKTIKUM-ALPRO1 - Latihan".
- Code Editor:** Displays a Go file named "case1.go" containing a switch statement. The code is as follows:

```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var pH float64
9     fmt.Println("Masukkan kadar pH: ")
10    fmt.Scan(&pH)
11
12    switch {
13        case pH < 0 || pH > 14:
14            fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
15
16        case pH >= 6.5 && pH <= 8.6:
17            fmt.Println("Air layak minum")
18
19    default:
20        fmt.Println("Air tidak layak minum")
21    }
22 }
```

- Terminal:** Shows the command "go run c:/Users/muh..." followed by the output of the program for different pH inputs.
- Output:** Shows the results of the program execution for pH values 8.6, 9, and 16.
- Code Preview:** A small window shows the code for "case1.go".

## Deskripsi program

Program ini mengevaluasi kelayakan air berdasarkan pH. Jika pH di luar 0–14, program menyatakan nilainya tidak valid. Jika pH berada pada rentang 6,5–8,6, air dinyatakan layak diminum. Selain itu, air dianggap tidak layak diminum. Program menampilkan hasil sesuai nilai pH yang dimasukkan.

## 2. Tugas 2

### Source code

```
package main

import (
    "fmt"
)

func main() {
    var jenis string
    var durasi int

    fmt.Print("Masukkan jenis kendaraan (motor/mobil/truk): ")
    fmt.Scan(&jenis)

    fmt.Print("Masukkan durasi parkir (jam): ")
    fmt.Scan(&durasi)

    if durasi < 1 {
        durasi = 1
    }

    var tarif int

    switch jenis {
    case "motor":
        tarif = 2000
    case "mobil":
        tarif = 5000
    case "truk":
        tarif = 8000
    }
}
```

```

default:

    fmt.Println("Jenis kendaraan tidak valid.")

    return

}

total := tarif * durasi

fmt.Println("Total biaya parkir: Rp", total)

}

```

## Screenshot program

The screenshot shows a Go development environment with the following components:

- EXPLORER**: Shows the project structure under "PRAKTIKUM-ALPRO1 - Latihan". It includes files like case2.go, guided3.go, main.go, main2.go, main3.go, main33.go, main44.go, guided 1.go, and several .go files for various assignments (latihan 1-6, LATIHAN SEMUA).
- CODE EDITOR**: Displays the content of case2.go. The code handles user input for vehicle type and parking duration, then calculates and prints the total parking fee based on rates (motor: Rp 2.000, mobil: Rp 5.000, truk: Rp 7.000). A tooltip "Windsurf Refactor | Explain | Generate GoDoc | X" is visible over the code.
- TERMINAL**: Shows the command "go run c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan\case2.go" being run, followed by the output: "Masukkan jenis kendaraan (motor/mobil/truk): motor", "Masukkan durasi parkir (jam): 3", "Total biaya parkir: Rp 6000".
- OUTPUT**: Shows the same terminal output as the terminal tab.
- DEBUG CONSOLE**: Shows the command "go run c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan\case2.go" and the output: "Masukkan jenis kendaraan (motor/mobil/truk): mobil", "Masukkan durasi parkir (jam): 1", "Total biaya parkir: Rp 5000".
- PORTS**: Shows the command "go run c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan\case2.go" and the output: "Masukkan jenis kendaraan (motor/mobil/truk): truk", "Masukkan durasi parkir (jam): 5".
- QUERY RESULTS**: Shows the command "go run c:\Users\muham\OneDrive\Documents\KULIAH\PRAKTIKUM-ALPRO1 - Latihan\case2.go" and the output: "Nama : Mohamad Naufal Mubarok", "Nim : 109082500128".

## Deskripsi program

Program tersebut menghitung biaya parkir berdasarkan jenis kendaraan dan lama durasi parkir. Pengguna memasukkan jenis kendaraan serta jumlah jam parkir. Jika durasi kurang dari satu jam, program otomatis menetapkannya menjadi satu jam. Setelah itu, *switch* menentukan tarif: motor dikenakan Rp 2.000 per jam, mobil Rp 5.000 per jam, dan truk Rp 7.000 per jam. Terakhir, program mengalikan tarif dengan durasi lalu menampilkan total biaya parkir yang harus dibayar.

## 1. Tugas 3

### Source code

```
package main

import (
    "fmt"
)

func main() {
    var n int
    fmt.Println("Masukkan bilangan: ")
    fmt.Scan(&n)

    switch {
    case n%2 != 0:
        hasil := n + (n + 1)
        fmt.Println("Kategori: Bilangan Ganjil")
        fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", n, n+1, hasil)

    case n%2 == 0 && n%10 != 0 && n%5 != 0:
        hasil := n * (n + 1)
        fmt.Println("Kategori: Bilangan Genap")
        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", n, n+1, hasil)

    case n%5 == 0 && n%10 != 0:
        hasil := n * n
        fmt.Println("Kategori: Bilangan Kelipatan 5")
        fmt.Printf("Hasil kuadrat dari %d = %d\n", n, hasil)

    case n%10 == 0:
        hasil := n / 10
        fmt.Println("Kategori: Bilangan Kelipatan 10")
        fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", n, hasil)

    default:
        fmt.Println("Kategori tidak ditemukan.")
    }
}
```

## Screenshot program

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- File Explorer:** Shows the project structure under "PRAKTIKUM-ALPRO1 - LATIHAN".
- Editor:** Displays a Go file named "case3.go" with code related to number classification. It includes several conditional statements using the modulus operator (%) to determine if numbers are odd, even, or multiples of 5.
- Terminal:** Shows the command "go run" followed by the path to the Go file, indicating the code is ready to be executed.
- Output:** Shows the results of the execution, including the classification of input numbers (5, 8, 25) as odd, even, and multiples of 5 respectively.
- File Browser:** A separate window on the right lists multiple files named "case3.go" from various locations, likely due to a naming conflict.

## Deskripsi program

Program tersebut meminta sebuah bilangan dari pengguna, lalu menentukan apakah bilangan itu ganjil, genap, atau kelipatan lima menggunakan *switch-case*. Setelah itu, program menghitung operasi tertentu: bilangan ganjil dijumlahkan dengan bilangan berikutnya, bilangan genap dikalikan dengan bilangan berikutnya, dan bilangan kelipatan lima dikuadratkan. Hasilnya kemudian ditampilkan ke layar.