

**LAPORAN PRAKTIKUM**  
**Algoritma Pemrograman**

**MODUL 11**  
**TIPE DATA DAN VARIABEL**



**Disusun oleh:**

**DIVA ZAHRAH NABILA**

**109082500112**

**S1IF-13-04**

**PROGRAM STUDI S1 INFORMATIKA**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2025**

## 1. Guided 1

### Source code

```
package main

import "fmt"

func main() {
    var jam12, jam24 int
    var label string
    fmt.Scan(&jam24)
    switch {
    case jam24 == 0:
        jam12 = 12
        label = "AM"
    case jam24 < 12:
        jam12 = jam24
        label = "AM"
    case jam24 == 12:
        jam12 = 12
        label = "PM"
    case jam24 > 12:
        jam12 = jam24 - 12
        label = "PM"
    }
    fmt.Println(jam12, label)
}
```

### Screenshot program

The screenshot shows a Go development environment with the following interface elements:

- File Bar:** File, Edit, Selection, View, Go, ...
- Toolbar:** Back, Forward, Search icon, MODUL 11
- Explorer:** Shows files in the MODUL 11 directory:
  - o 1jam.go (selected)
  - o 1jam.go 2
  - o 2tanaman.go
  - o guided1.go
- Code Editor:** Displays the Go source code for the `1jam.go` file.
- Terminal:** Shows the command line output of running the program.

Code in `1jam.go`:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var jam12, jam24 int
7     var label string
8     fmt.Scan(&jam24)
9     switch {
10    case jam24 == 0:
11        jam12 = 12
12        label = "AM"
13    case jam24 < 12:
14        jam12 = jam24
15        label = "AM"
16    case jam24 == 12:
17        jam12 = 12
18        label = "PM"
19    case jam24 > 12:
20        jam12 = jam24 - 12
21        label = "PM"
22    }
23    fmt.Println(jam12, label)
24 }
```

Terminal Output:

```
PS D:\LAPRAK\MODUL 11> go run "d:\LAPRAK\MODUL 11\1jam.go"
13
1 PM
PS D:\LAPRAK\MODUL 11> go run "d:\LAPRAK\MODUL 11\1jam.go"
0
12 AM
PS D:\LAPRAK\MODUL 11>
```

## Deskripsi program

Program ini berfungsi untuk mengubah format waktu dari 24 jam menjadi 12 jam. Pengguna memasukkan sebuah angka jam dalam format 24 jam (0–23), lalu program mengonversinya menjadi jam 12 beserta penanda waktu AM atau PM.

1. `var jam12, jam24 int, var label string` digunakan untuk menyimpan nilai dalam mengkonversi AM dan PM. Sedangkan `fmt.Scan(&jam24)` untuk membaca inputan dari pengguna.
2. `switch {`  
`case jam24 == 0:`  
    `jam12 = 12`  
    `label = "AM"` artinya jika pengguna memasukan angka 0, maka akan dikonversi menjadi 12 dan diberi label AM.  
`case jam24 < 12:`  
    `jam12 = jam24`  
    `label = "AM"` artinya jika pengguna memasukan angka 1-11 maka jam tetap dan diberi label AM.  
`case jam24 == 12:`  
    `jam12 = 12`  
    `label = "PM"` artinya jika pengguna memasukan angka 12 maka jam tetap 12 dan diberi label PM.  
`case jam24 > 12:`  
    `jam12 = jam24 - 12`  
    `label = "PM"` artinya jika pengguna memasukan angka lebih dari 12, maka angka itu dikurang 12. Hasil itulah yang digunakan sebagai jam, dan diberi label PM.
3. `fmt.Println(jam12, label)` untuk menampilkan hasil konversi jam dan labelnya.

## 2. Guided 2

### Source code

```
package main
import "fmt"
func main() {
    var nama_tanaman string
    fmt.Scan(&nama_tanaman)
    switch nama_tanaman {
    case "nepenthes", "drosera":
        fmt.Println("Termasuk Tanaman Karnivora.")
        fmt.Println("Asli Indonesia.")
    case "venus", "sarracenia":
        fmt.Println("Termasuk Tanaman Karnivora.")
        fmt.Println("Tidak Asli Indonesia.")
    default:
        fmt.Println("Tidak termasuk Tanaman
Karnivora.")
    }
}
```

## Screenshoot program

```
File Edit Selection View Go ... < > MODUL 11
EXPLORER ... Welcome 1jam.go 2 2tanaman.go 1 guided1.go 1
MODUL 11
1jam.go 2
2tanaman.go 1
guided1.go 1
1 package main
2
3 import "fmt"
4
5 func main() {
6     var nama_tanaman string
7     fmt.Scan(&nama_tanaman)
8     switch nama_tanaman {
9         case "nepenthes", "drosera":
10             fmt.Println("Termasuk Tanaman Karnivora.")
11             fmt.Println("Asli Indonesia.")
12         case "venus", "sarracenia":
13             fmt.Println("Termasuk Tanaman Karnivora.")
14             fmt.Println("Tidak Asli Indonesia.")
15     default:
16         fmt.Println("Tidak termasuk Tanaman Karnivora.")
17     }
18 }
19

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\LAPRAK\MODUL 11> go run "d:\LAPRAK\MODUL 11\2tanaman.go"
nepenthes
Termasuk Tanaman Karnivora.
Asli Indonesia.
PS D:\LAPRAK\MODUL 11> go run "d:\LAPRAK\MODUL 11\2tanaman.go"
venus
Termasuk Tanaman Karnivora.
Tidak Asli Indonesia.
PS D:\LAPRAK\MODUL 11> go run "d:\LAPRAK\MODUL 11\2tanaman.go"
kareddk
Tidak termasuk Tanaman Karnivora.
PS D:\LAPRAK\MODUL 11>
```

## Deskripsi program

Program ini digunakan untuk mengidentifikasi apakah suatu tanaman termasuk tanaman karnivora serta menentukan apakah tanaman tersebut berasal dari Indonesia atau bukan. Pengguna memasukkan nama tanaman, lalu program mengecek nama tersebut menggunakan struktur switch case untuk menampilkan informasi yang sesuai.

1. `var nama_tanaman string` untuk menampung inputan pengguna.
2. `fmt.Scan(&nama_tanaman)` untuk membaca inputan dari pengguna.
3. `switch nama_tanaman {`  
 `case "nepenthes", "drosera":`  
 `fmt.Println("Termasuk Tanaman Karnivora.")`  
 `fmt.Println("Asli Indonesia.")` artinya jika pengguna memasukan nepenthes atau drosera maka program akan menjelaskan bahwa itu merupakan tanaman karnivora dan asli Indonesia.  
 `case "venus", "sarracenia":`  
 `fmt.Println("Termasuk Tanaman Karnivora.")`  
 `fmt.Println("Tidak Asli Indonesia.")` artinya jika pengguna memasukan venus atau sarracenia maka program akan menjelaskan bahwa itu merupakan tanaman karnivora namun bukan asli Indonesia.  
 `default:`  
 `fmt.Println("Tidak termasuk Tanaman Karnivora.")` jika pengguna memasukan tanaman selain yang ada di daftar maka program akan menjelaskan bahwa tanaman itu bukan tanaman karnivora.

### 3. Guided 3

#### Source code

```
package main

import "fmt"

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 (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("Jenis kendaraan atau durasi parkir tidak valid")
    }
    fmt.Printf("Tarif Parkir: Rp %d\n", tarif)
}
```

## Screenshoot program

```
File Edit Selection View Go ... < > MODUL 11
EXPLORER ... 3parkir.go > ...
MODUL 11
< 1jam.go 2
< 2tanaman.go
< 3parkir.go 1
3parkir.go 1
3 import "fmt"
4
5 func main() {
6     var kendaraan string
7     var durasi int
8     var tarif int
9     fmt.Print("Masukkan jenis kendaraan (Motor/Mobil/Truk): ")
10    fmt.Scan(&kendaraan)
11    fmt.Print("Masukkan durasi parkir (dalam jam): ")
12    fmt.Scan(&durasi)
13    switch {
14        case kendaraan == "Motor" && durasi >= 1 && durasi <= 2:
15            tarif = 7000
16        case kendaraan == "Motor" && durasi > 2:
17            tarif = 9000
18        case kendaraan == "Mobil" && durasi >= 1 && durasi <= 2:
19            tarif = 15000
20        case kendaraan == "Mobil" && durasi > 2:
21            tarif = 20000
22        case kendaraan == "Truk" && durasi >= 1 && durasi <= 2:
23            tarif = 25000
24        case kendaraan == "Truk" && durasi > 2:
25            tarif = 35000
26        default:
27            fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")
28    }
29    fmt.Printf("Tarif Parkir: Rp %d\n", tarif)
30 }
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 2
Tarif Parkir: Rp 7000
PS D:\APRAK\MODUL 11> go run "d:\APRAK\MODUL 11\3parkir.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Mobil
Masukkan durasi parkir (dalam jam): 4
Tarif Parkir: Rp 20000
PS D:\APRAK\MODUL 11> go run "d:\APRAK\MODUL 11\3parkir.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Sepeda
Masukkan durasi parkir (dalam jam): 2
Jenis kendaraan atau durasi parkir tidak valid
> OUTLINE
> TIMELINE
> GO
> PACKAGE OUTLINE
4 ▲ 0
Ln 31, Col 1 Tab 5b
```

## Deskripsi program

Program ini digunakan untuk menghitung tarif parkir berdasarkan jenis kendaraan dan durasi parkir. Pengguna memasukkan jenis kendaraan (Motor, Mobil, atau Truk) dan lama parkir dalam jam. Program kemudian menentukan tarif yang sesuai menggunakan *switch case* dengan kondisi, lalu menampilkan total tarif parkir.

1. **var kendaraan string, var durasi int** merupakan variabel yang diinputkan pengguna, sedangkan **var tarif int** untuk menyimpan hasil perhitungan.
2. **fmt.Print("Masukkan jenis kendaraan (Motor/Mobil/Truk): ")**  
**fmt.Scan(&kendaraan)**  
**fmt.Print("Masukkan durasi parkir (dalam jam): ")**  
**fmt.Scan(&durasi)** untuk menyimpan inputan pengguna.
3. **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("Jenis kendaraan atau durasi parkir tidak valid")**
4. **fmt.Printf("Tarif Parkir: Rp %d\n", tarif)** selain perintah diatas akan menampilkan tidak valid.

Perintah diatas digunakan untuk menentukan tarif sesuai dengan daftar kendaraan yang ada.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {
    var ph float64
    fmt.Scan(&ph)
    switch {
    case ph < 0 || ph > 14:
        fmt.Println("Input tidak valid, rentang pH 0 - 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 code editor interface with the following details:

- File Explorer:** Shows files in the 'MODUL 11' folder: 1jam.go (3), 2tanaman.go (1), 3parkir.go (1), and soal1.go (1). The 'soal1.go' file is currently selected.
- Code Editor:** Displays the Go code for 'soal1.go'. The code defines a package 'main' with a function 'main()' that reads a float64 value 'ph' from standard input. It then uses a switch statement to check if 'ph' is within the range [0, 14]. If it's not, it prints an invalid message. If it is, and falls between 6.5 and 8.6, it prints 'Air layak minum'. Otherwise, it prints 'Air tidak layak minum'.
- Terminal:** Shows the command 'go run "d:\LAPRAK\MODUL 11\soal1.go"' being run twice. The first run shows the output 'Air layak minum'. The second run shows the output 'Air tidak layak minum'.

#### Deskripsi program

1. **var ph float64** untuk menyimpan nilai ph.
2. **fmt.Scan(&ph)** menyimpan nilai inputan dari pengguna.
3. **case ph < 0 || ph > 14:**  
    **fmt.Println("Input tidak valid, rentang pH 0 - 14")** Jika nilai pH kurang dari 0 atau lebih dari 14, program menampilkan bahwa input tidak valid karena pH hanya berada dalam rentang 0–14.
4. **case ph >= 6.5 && ph <= 8.6:**  
    **fmt.Println("Air layak minum")** Jika nilai pH berada pada rentang 6.5 – 8.6, maka air dianggap layak untuk diminum.
5. **default:**  
    **fmt.Println("Air tidak layak minum")** selain perintah diatas akan menampilkan tidak layak diminum, contoh <6.5 dan >8.6

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {
    var kendaraan string
    var jam int
    fmt.Scan(&kendaraan, &jam)
    if jam < 1 {
        jam = 1
    }
    tarif := 0
    switch kendaraan {
    case "Motor":
        tarif = 2000
    case "Mobil":
        tarif = 5000
    case "Truk":
        tarif = 8000
    default:
        fmt.Println("Jenis kendaraan tidak valid!")
        return
    }
    total := tarif * jam
    fmt.Print("Rp", total)
}
```

### Screenshot program

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

- EXPLORER:** Shows files in the 'MODUL 11' folder: 1jam.go (4), 2tanaman.go (1), 3parkir.go (1), soal1.go (1), and soal2.go (1).
- CODE EDITOR:** Displays the source code for 'soal2.go'. The code is identical to the one shown in the previous text block.
- TERMINAL:** Shows the command 'go run "d:\LAPRAK\MODUL 11\soal2.go"' being run twice, once for 'Motor' and once for 'Mobil', with the expected output 'Rp6000' and 'Rp5000' respectively.
- AI ASSISTANT:** A sidebar on the right is titled 'Build with Agent'. It includes a message 'AI responses may be inaccurate', a button 'Generate Agent Instructions to onboard your codebase.', and a text input field 'Describe what to build next' with placeholder text 'soal2.go +'.

### **Deskripsi program**

1. **var kendaraan string** dan **var jam int** untuk menyimpan nilai yang akan dimasukan pengguna.
2. **fmt.Scan(&kendaraan, &jam)** pengguna memasukan kendaraan dan jam.
3. **if jam < 1 {jam = 1}** artinya jika pengguna memasukan angka kurang dari satu, akan tetap dihitung satu.
4. **switch kendaraan {**  
**case "Motor":**  
    **tarif = 2000**  
**case "Mobil":**  
    **tarif = 5000**  
**case "Truk":**  
    **tarif = 8000**  
code diatas digunakan untuk memberikan nilai tarif sesuai dengan kendaraan.
5. **default:**  
    **fmt.Println("Jenis kendaraan tidak valid!")**  
    **return** artinya jika pengguna memasukan kendaraan selain diatas akan menampilkan tidak valid.
6. **total := tarif \* jam** digunakan untuk menghitung total biaya parkir, yaitu tarif sesuai kendaraan dikali jam.

### 3. Tugas 3

#### Source code

```
package main

import "fmt"

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

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

    case n%5 == 0 && n != 5:
        fmt.Println("Kategori: Bilangan Kelipatan 5")
        fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", n, n*n)

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

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

#### Screenshot program

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

- File Menu:** File, Edit, Selection, View, Go, ...
- Toolbar:** Includes icons for back, forward, search, and other common operations.
- Editor Area:** Displays the content of the file `soal3.go`. The code is identical to the one provided in the text block above.
- Output Area:** Shows the terminal output of running the program with different inputs:
  - Input: 11, Output: Kategori: Bilangan Ganjil
  - Input: 25, Output: Kategori: Bilangan Kelipatan 5
  - Input: 8, Output: Kategori: Bilangan Genap
  - Input: 20, Output: Kategori: Bilangan Kelipatan 10
- Bottom Status Bar:** Shows the current file is `soal3.go`.

## Deskripsi program

1. `var n int` untuk menyimpan nilai inputan. `fmt.Scan(&n)` pengguna memasukan nilai.
2. `switch {`  
`case n%10 == 0:`  
    `fmt.Println("Kategori: Bilangan Kelipatan 10")`  
    `fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", n,`  
    `n/10) Jika angka habis dibagi 10, program mengategorikannya sebagai bilangan kelipatan 10. Lalu angka yang dimasukan akan dibagi 10.`  
  
`case n%5 == 0 && n != 5:`  
    `fmt.Println("Kategori: Bilangan Kelipatan 5")`  
    `fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", n, n*n) Jika`  
    `angka habis dibagi 5 tetapi bukan angka 5, maka kategori adalah bilangan kelipatan 5.`  
    `Lalu angka yang dimasukan akan dikuadratkan.`  
  
`case n%2 != 0:`  
    `fmt.Println("Kategori: Bilangan Ganjil")`  
    `fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya`  
`%d + %d =`  
    `%d\n", n, n+1, n+(n+1)) Jika angka tidak habis dibagi 2, maka termasuk`  
    `bilangan ganjil. Lalu angka yang dimasukan akan ditambah dengan angka setelahnya.`  
  
`case n%2 == 0:`  
    `fmt.Println("Kategori: Bilangan Genap")`  
    `fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d`  
    `* %d = %d\n", n, n+1, n*(n+1)) Jika angka habis dibagi 2, berarti angka`  
    `adalah bilangan genap. Lalu angka yang dimasukan akan dikali angka setelahnya.`