

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 11

SWITCH - CASE



Disusun oleh:

ELSA DWI RIZQIYANTI

109082500090

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 periode string

    fmt.Print("Masukkan jam dalam format 24 jam (0-23): ")
    fmt.Scan(&jam24)

    if jam24 < 0 || jam24 > 23 {
        fmt.Println("Input tidak valid! Masukkan angka 0-23.")
        return
    }

    switch {
    case jam24 == 0:
        jam12 = 12
        periode = "AM"
    case jam24 == 12:
        jam12 = 12
    }
```

```

periode = "PM"

case jam24 > 12:

    jam12 = jam24 - 12

    periode = "PM"

default:

    jam12 = jam24

    periode = "AM"

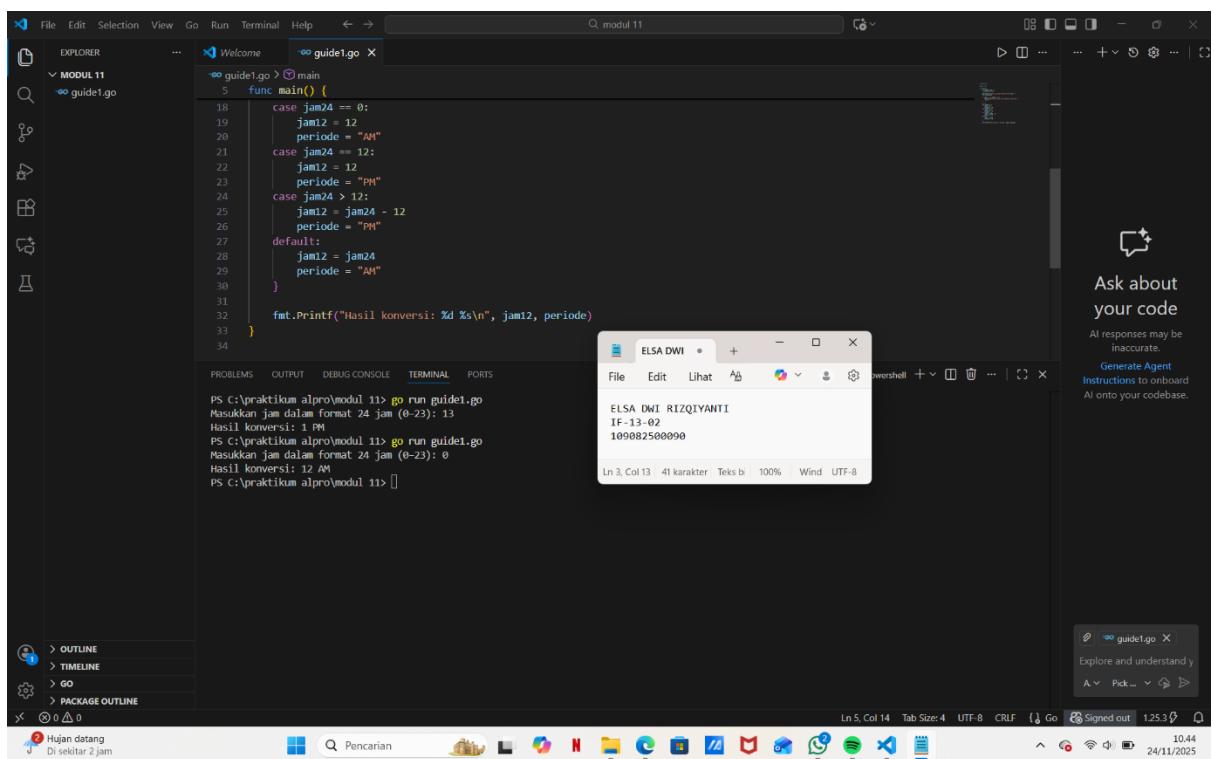
}

fmt.Printf("Hasil konversi: %d %s\n", jam12, periode)

}

```

Screenshot program



Deskripsi program

Deklarasi Variabel:

```
var jam24, jam12 int
```

```
var periode string
```

jam24: menyimpan input jam dalam format 24 jam.

jam12: menyimpan hasil konversi ke format 12 jam.

Periode: menyimpan "AM" atau "PM".

Input dari Pengguna:

```
fmt.Println("Masukkan jam dalam format 24 jam (0–23): ")
```

```
fmt.Scan(&jam24)
```

Program meminta pengguna memasukkan jam antara 0 sampai 23.

Validasi Input:

```
if jam24 < 0 || jam24 > 23 {  
    fmt.Println("Input tidak valid! Masukkan angka 0–23.")  
    return  
}
```

Jika pengguna memasukkan angka di luar rentang 0–23, program akan menolak dan berhenti (return).

Proses Konversi dengan switch tanpa ekspresi

```
switch {  
case jam24 == 0:  
    jam12 = 12  
    periode = "AM"  
case jam24 == 12:  
    jam12 = 12  
    periode = "PM"  
case jam24 > 12:  
    jam12 = jam24 - 12  
    periode = "PM"  
default:  
    jam12 = jam24  
    periode = "AM"  
}
```

Penjelasan setiap case:

Case 1 — jam24 == 0

Contoh: 00

Di format 12 jam menjadi: 12 AM

Case 2 — jam24 == 12

Contoh: 12

Di format 12 jam menjadi: 12 PM

Case 3 — jam24 > 12

Contoh: 13, 14, 15 ... 23

Konversi: jam24 - 12

Misal 15: 15 – 12 = 3 PM

Default — jam24 di antara 1 sampai 11

Konversi langsung tanpa perubahan dan periode = AM.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var tanaman string

    fmt.Scan(&tanaman)

    switch tanaman {
        case "nepenthes":
            fmt.Println("termasuk tanaman Karnivora Asli
Indonesia")
            fmt.Println("asli indonesia")
        case "venus":
            fmt.Println("termasuk tanaman Karnivora Asli
Indonesia")
            fmt.Println("bukan asli indonesia")
        default:
    }
}
```

```

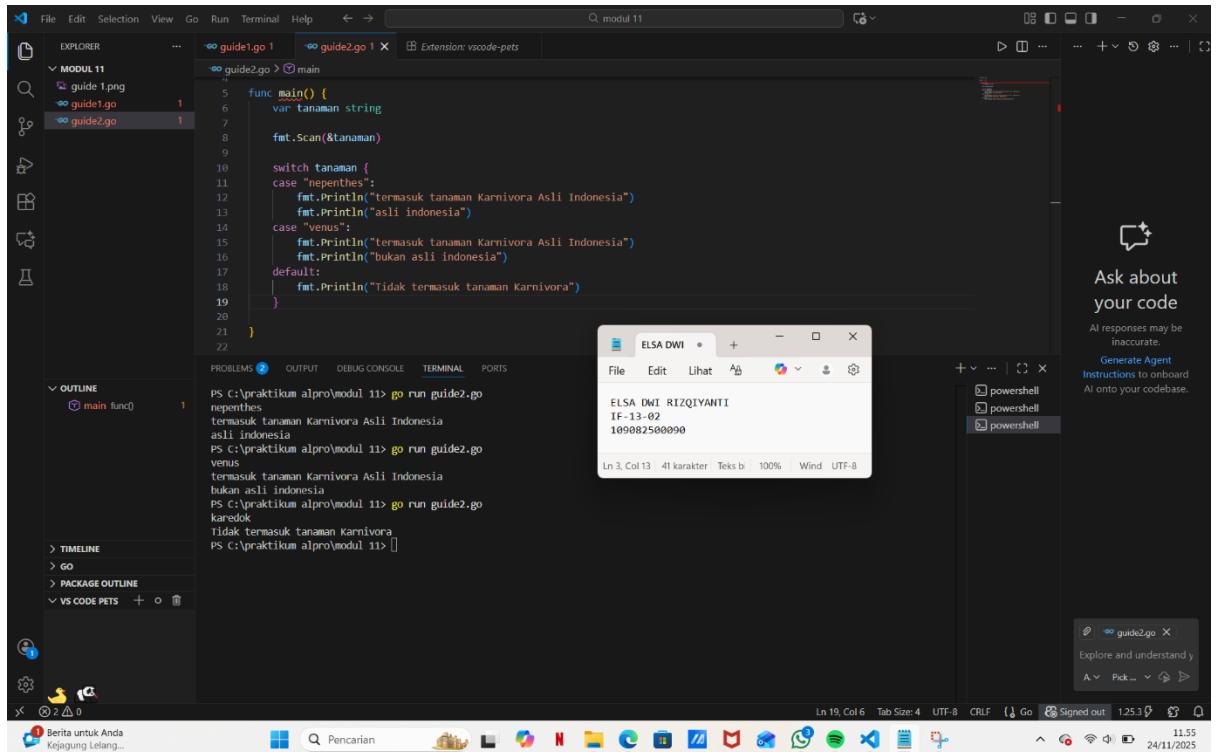
        fmt.Println("Tidak termasuk tanaman Karnivora")

    }

}

```

Screenshot program



Deskripsi program

Pengelompokan dengan switch case:

```

switch tanaman {

case "nepenthes":

    fmt.Println("termasuk tanaman Karnivora Asli Indonesia")

    fmt.Println("asli indonesia")



case "venus":



    fmt.Println("termasuk tanaman Karnivora Asli Indonesia")

    fmt.Println("bukan asli indonesia")
}

```

default:

```
    fmt.Println("Tidak termasuk tanaman Karnivora")  
}
```

Penjelasan setiap case:

Case 1 — tanaman = "nepenthes"

Nepenthes adalah tumbuhan kantong semar.

Termasuk tanaman karnivora.

Dan merupakan tanaman asli Indonesia.

Case 2 — tanaman = "venus":

Venus flytrap adalah tanaman karnivora dari Amerika Utara.

Jadi termasuk tanaman karnivora, tetapi bukan asli Indonesia.

Default — semua input selain "nepenthes" atau "venus":

Jika pengguna mengetik selain dua itu program akan menampilkan
Tidak termasuk tanaman karnivora.

3. Guided 3

Source Code

```
package main  
  
import "fmt"  
  
func main() {  
    var jenis string  
    var durasi, tarif int
```

```
fmt.Println("jenis kendaraan (motor/mobil/truk): ")

fmt.Scanln(&jenis)

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

fmt.Scanln(&durasi)

switch jenis {

case "motor":

    if durasi >= 1 && durasi <= 2 {

        tarif = 7000

    } else if durasi > 2 {

        tarif = 9000

    }

}

case "mobil":

    if durasi >= 1 && durasi <= 2 {

        tarif = 15000

    } else if durasi > 2 {

        tarif = 20000

    }

}

case "truk":

    if durasi >= 1 && durasi <= 2 {

        tarif = 25000

    } else if durasi > 2 {

        tarif = 30000

    }

}
```

```

    }

    default:

        fmt.Println("Jenis kendaraan atau durasi parkir
tidak valid")

        return

    }

}

fmt.Println("Tarif Parkir: Rp", tarif)

}

```

Screenshot program

The screenshot shows a Windows desktop environment. On the left is the Visual Studio Code interface, displaying a Go file named 'guide3.go' with code related to parking rates. On the right is a terminal window titled 'ELSA DWI RIZQIYANTI' showing command-line output for running the program. The taskbar at the bottom includes icons for various applications like File Explorer, Edge browser, and Task View.

```

File Edit Selection View Go Run Terminal Help ↻ 🔍 modul 11
MODUL_11
guide1.png
guide2.png
guide1.go 5
guide2.go 1
guide3.go 1
LAPRAK MODUL_11_S...
soal1.png
soal2.png
soal1.go 1
soal2.go 1
soal3.go 1
OUTLINE
PROBLEMS 0 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\praktikum alpro\modul_11> go run guide3.go
jenis kendaraan (motor/mobil/truk): motor
durasi parkir (dalam jam): 2
Tarif Parkir: Rp 2000
PS C:\praktikum alpro\modul_11> go run guide3.go
jenis kendaraan (motor/mobil/truk): mobil
durasi parkir (dalam jam): 4
Tarif Parkir: Rp 20000
PS C:\praktikum alpro\modul_11> go run guide3.go
jenis kendaraan (motor/mobil/truk): motor
durasi parkir (dalam jam): 3
Tarif Parkir: Rp 25000
PS C:\praktikum alpro\modul_11> go run guide3.go
jenis kendaraan (motor/mobil/truk): truk
durasi parkir (dalam jam): 1
tarif Parkir: Rp 25000
PS C:\praktikum alpro\modul_11> go run guide3.go
jenis kendaraan (motor/mobil/truk): sepeda
durasi parkir (dalam jam): 2
tarif Parkir: Rp 25000
PS C:\praktikum alpro\modul_11>

```

Deskripsi program

Case 1 — Jika kendaraan adalah motor

case "motor":

```

if durasi >= 1 && durasi <= 2 {

    tarif = 7000

} else if durasi > 2 {

```

```
    tarif = 9000  
}
```

Tarif:

1–2 jam: Rp 7.000

lebih dari 2 jam: Rp 9.000

Case 2 — Jika kendaraan adalah mobil

case "mobil":

```
    if durasi >= 1 && durasi <= 2 {  
        tarif = 15000  
    } else if durasi > 2 {  
        tarif = 20000  
    }
```

Tarif:

1–2 jam → Rp 15.000

lebih dari 2 jam → Rp 20.000

Case 3 — Jika kendaraan adalah truk

case "truk":

```
    if durasi >= 1 && durasi <= 2 {  
        tarif = 25000  
    } else if durasi > 2 {  
        tarif = 30000  
    }
```

Tarif:

1–2 jam → Rp 25.000

lebih dari 2 jam → Rp 30.000

Default — Input tidak valid (Jika pengguna memasukkan jenis kendaraan selain tiga pilihan, program langsung berhenti.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var ph float64

    fmt.Print("Masukkan nilai 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

Deskripsi program

Evaluasi Setiap Kondisi:

Case 1 – Validasi pH

```
case ph < 0 || ph > 14:
```

```
fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
```

Jika nilai pH yang dimasukkan di luar rentang 0 sampai 14, maka input dianggap tidak valid, karena skala pH hanya berada pada range itu.

Program akan menampilkan pesan kesalahan.

Case 2 – Air layak minum

case ph >= 6.5 && ph <= 8.6:

```
fmt.Println("Air layak minum")
```

Jika pH berada pada rentang aman menurut standar kualitas air minum (6.5–8.6), maka air dianggap layak minum.

Default (Air tidak layak minum)

default:

```
fmt.Println("Air tidak layak minum")
```

Untuk semua nilai pH di luar rentang 6.5–8.6 (tetapi masih valid 0–14), air dianggap tidak layak minum, misalnya:

pH terlalu asam (< 6.5)

pH terlalu basa (> 8.6)

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var jenis string
    var jam int

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

    fmt.Scan(&jenis)

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

    fmt.Scan(&jam)

    if jam < 1 {
        jam = 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 * jam

fmt.Printf("Total biaya parkir: Rp %d\n", total)
}

```

Screenshoot program

The screenshot shows a Windows desktop environment with a Visual Studio Code (VS Code) window open. The code editor contains a Go file named `soal2.go` with the following content:

```

func main() {
    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 * jam
    fmt.Printf("Total biaya parkir: Rp %d\n", total)
}

```

A terminal window titled "ELSA DWI" is visible, showing the command `go run soal2.go` being run and the resulting output:

```

PS C:\praktikum alpro\modul 11> go run soal2.go
Masukkan jenis kendaraan (motor/mobil/truk): motor 3 jam
Masukkan durasi parkir (jam): Total biaya parkir: Rp 6000
PS C:\praktikum alpro\modul 11> go run soal2.go
Masukkan jenis kendaraan (motor/mobil/truk): mobil 1 jam
Masukkan durasi parkir (jam): Total biaya parkir: Rp 5000
PS C:\praktikum alpro\modul 11> go run soal2.go
Masukkan jenis kendaraan (motor/mobil/truk): truk 5 jam
Masukkan durasi parkir (jam): Total biaya parkir: Rp 40000
PS C:\praktikum alpro\modul 11> []

```

Deskripsi program

Validasi lama parkir:

```
if jam < 1 {  
    jam = 1  
}
```

Aturan: durasi minimal adalah 1 jam.

Jika user memasukkan 0 atau angka negatif, program otomatis mengubahnya menjadi 1 jam.

Menentukan tarif berdasarkan jenis kendaraan:

```
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  
}
```

Penjelasan setiap case:

motor → tarif 2000 per jam

mobil → tarif 5000 per jam

truk → tarif 8000 per jam

Jika input tidak cocok dengan tiga kategori tersebut (misal: "bus", "sepeda", dll.), program menampilkan:

Jenis kendaraan tidak valid.

Menghitung total biaya parkir:

```
total := tarif * jam
```

Total dihitung dengan mengalikan:

tarif per jam × lama parkir

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Print("Masukkan bilangan: ")
    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:
        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 Genap")
        fmt.Printf("Hasil perkalian dengan bilangan
berikutnya %d * %d = %d\n", n, n+1, n*(n+1))

    default:
        fmt.Println("Kategori: Bilangan Ganjil")
        fmt.Printf("Hasil penjumlahan dengan bilangan
berikutnya %d + %d = %d\n", n, n+1, n+(n+1))
    }
}
```

Screenshot program

The screenshot shows a Windows desktop environment. In the center is a Microsoft Visual Studio Code (VS Code) window. The code editor displays a Go program named 'soal3.go' with the following content:

```
func main() {
    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:
            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 Genap")
            fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", n, n+1, n*(n+1))
        default:
            fmt.Println("Kategori: Bilangan Ganjil")
            fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", n, n+1, n+(n+1))
    }
}
```

The terminal window to the right of VS Code shows the output of running the program:

```
PS C:\praktikum alpro\modul 11> go run soal3.go
Masukkan bilangan: 5
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 5 ^ 2 = 25
PS C:\praktikum alpro\modul 11> go run soal3.go
Masukkan bilangan: 8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS C:\praktikum alpro\modul 11> go run soal3.go
Masukkan bilangan: 25
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 25 ^ 2 = 625
PS C:\praktikum alpro\modul 11> go run soal3.go
Masukkan bilangan: 20
Kategori: Bilangan Kelipatan 10
Hasil pembagian antara 20 / 10 = 2
PS C:\praktikum alpro\modul 11> []
```

The taskbar at the bottom of the screen shows various pinned icons, including a browser, file explorer, and system tray icons.

Deskripsi program

Evaluasi Setiap Kondisi:

Case 1 — Bilangan Kelipatan 10

case n%10 == 0:

Syarat: bilangan habis dibagi 10.

Misalnya: 10, 20, 30, 40, ...

Program menampilkan:

Kategori: Bilangan Kelipatan 10

Lalu melakukan perhitungan:

```
fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", n, n/10)
```

Contoh:

$20 / 10 = 2$

Case 2 — Bilangan Kelipatan 5

case n%5 == 0:

Syarat: bilangan habis dibagi 5 tetapi *tidak termasuk kelipatan 10* (karena kelipatan 10 sudah ditangani di case pertama).

Contoh: 5, 15, 25, 35, ...

Program menampilkan:

Kategori: Bilangan Kelipatan 5

Lalu menghitung kuadrat:

```
fmt.Printf("Hasil kuadrat dari %d ^ 2 = %d\n", n, n*n)
```

Contoh:

$$15^2 = 225$$

Case 3 — Bilangan Genap

case $n \% 2 == 0$:

Syarat: bilangan genap tetapi bukan kelipatan 5 atau 10.

Contoh: 2, 4, 6, 8, 14, 16, 18, ...

Program menampilkan:

Kategori: Bilangan Genap

Lalu menghitung perkalian dengan bilangan berikutnya:

```
fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", n, n+1, n*(n+1))
```

Contoh:

$$6 \times 7 = 42$$

Default — Bilangan Ganjil

Jika tidak memenuhi semua kondisi di atas, berarti bilangan ganjil.

Contoh: 1, 3, 7, 9, 11, ...

Program menampilkan:

Kategori: Bilangan Ganjil

Lalu menghitung penjumlahan dengan bilangan berikutnya:

```
fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", n, n+1, n+(n+1))
```

Contoh:

$$7 + 8 = 15$$