

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 11

SWITCH CASE



Disusun oleh:

M MAHDAN ARGYA SYARIF

109082500059

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

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 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 GoLang development environment in a code editor. On the left, the code for a Go program is displayed:

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

On the right, the terminal window shows the execution of the program:

- PS C:\GoLang> go run guided1.go
13
1 PM
- PS C:\GoLang> go run guided1.go
0
12 AM
- PS C:\GoLang> go run guided1.go
12
12 PM

Deskripsi program

Program ini dirancang untuk mengonversi format waktu 24 jam menjadi format internasional 12 jam menggunakan dua tipe data utama, yaitu integer (jam12, jam24) untuk operasi numerik dan string (label) untuk penanda periode waktu. Dengan menerapkan algoritma *switch case*, logika program membagi input ke dalam beberapa kondisi spesifik: jika jam24 bernilai 0 maka dikonversi menjadi 12 AM, jika bernilai 12 menjadi 12 PM, nilai di bawah 12 ditetapkan sebagai AM, sedangkan nilai di atas 12 ditetapkan sebagai PM dengan operasi pengurangan nilai jam sebesar 12. Melalui mekanisme ini, variabel integer berfungsi menangani konversi angka jam, sementara

variabel string memastikan pelabelan AM dan PM diberikan secara akurat sesuai aturan waktu internasional.

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.")
    }
}
```

Screenshot program

The screenshot shows a GoLang development environment. On the left, the code editor displays a Go program named `guided2.go`. The code reads a string input from the user and prints whether it's a carnivorous plant native to Indonesia or not. The code is as follows:

```
1 package main
2 import "fmt"
3 func main() {
4     var nama_tanaman string
5     fmt.Scan(&nama_tanaman)
6
7     switch nama_tanaman {
8     case "nepenthes", "drosera":
9         fmt.Println("Termasuk Tanaman Karnivora.")
10        fmt.Println("Asli Indonesia.")
11    case "venus", "sarracenia":
12        fmt.Println("Termasuk Tanaman Karnivora.")
13        fmt.Println("Tidak Asli Indonesia.")
14    default:
15        fmt.Println("Tidak termasuk Tanaman Karnivora.")
16    }
17 }
```

To the right of the code editor is a terminal window showing the execution of the program. It lists three test cases: running the program with "nepenthes" and "drosera" both result in "Termasuk Tanaman Karnivora." and "Asli Indonesia.", while running it with "venus" results in "Termasuk Tanaman Karnivora." and "Tidak Asli Indonesia.". The terminal also shows the user information: Nama: M Mahdan Argya Syarif, NIM: 109082500059, Kelas: IF-13-07.

Below the terminal, the status bar indicates the current line (Ln 3, Col 16), character count (61 characters), file type (Plain text), and zoom level (100%).

At the bottom of the interface, there are tabs for PROBLEMS (10), OUTPUT, TERMINAL (selected), DEBUG CONSOLE, and PORTS.

Deskripsi program

Program ini dirancang untuk melakukan klasifikasi tanaman dengan fokus spesifik pada spesies karnivora asal Indonesia, di mana sistem akan mengidentifikasi input pengguna dan menempatkannya ke dalam salah satu dari tiga kategori output: tanaman karnivora asli Indonesia, tanaman karnivora non-endemik (luar negeri), atau bukan termasuk tanaman karnivora. Secara teknis, program ini mengandalkan tipe data string sepenuhnya untuk menangani seluruh alur informasi, mulai dari penerimaan nama tanaman yang diinputkan oleh pengguna hingga penyajian pernyataan status klasifikasi tanaman tersebut.

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)
}

```

Screenshot program

The screenshot shows a GoLang IDE interface. On the left, the code editor displays a Go program with syntax highlighting. On the right, a terminal window shows the execution of the program and its output.

```

1 package main
2 import "fmt"
3 func main() {
4     var kendaraan string
5     var durasi int
6     var tarif int
7     fmt.Print("Masukkan jenis kendaraan (Motor/Mobil/Truk): ")
8     fmt.Scan(&kendaraan)
9     fmt.Print("Masukkan durasi parkir (dalam jam): ")
10    fmt.Scan(&durasi)
11
12    switch {
13        case kendaraan == "Motor" && durasi >= 1 && durasi <= 2:
14            tarif = 7000
15        case kendaraan == "Motor" && durasi > 2:
16            tarif = 9000
17        case kendaraan == "Mobil" && durasi >= 1 && durasi <= 2:
18            tarif = 15000
19        case kendaraan == "Mobil" && durasi > 2:
20            tarif = 20000
21        case kendaraan == "Truk" && durasi >= 1 && durasi <= 2:
22            tarif = 25000
23        case kendaraan == "Truk" && durasi > 2:
24            tarif = 35000
25    default:
26        fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")
27    }
28    fmt.Printf("Tarif Parkir: Rp %d\n", tarif)
29 }

```

Terminal Output:

- PS C:\GoLang> go run guided3.go

Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor

Masukkan durasi parkir (dalam jam): 2

Tarif Parkir: Rp 7000
- PS C:\GoLang> go run guided3.go

Masukkan jenis kendaraan (Motor/Mobil/Truk): Mobil

Masukkan durasi parkir (dalam jam): 4

Tarif Parkir: Rp 20000
- PS C:\GoLang> go run guided3.go

Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor

Masukkan durasi parkir (dalam jam): 3

Tarif Parkir: Rp 9000
- PS C:\GoLang> go run guided3.go

Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk

Masukkan durasi parkir (dalam jam): 1

Tarif Parkir: Rp 25000
- PS C:\GoLang> go run guided3.go

Masukkan jenis kendaraan (Motor/Mobil/Truk): Sepeda

Masukkan durasi parkir (dalam jam): 2

Jenis kendaraan atau durasi parkir tidak valid

Tarif Parkir: Rp 0
- PS C:\GoLang> []

Deskripsi program

Program ini dirancang untuk menghitung tarif parkir secara otomatis berdasarkan klasifikasi jenis kendaraan dan durasi waktu parkir menggunakan kombinasi struktur kontrol *switch case* dan *if-else*. Dengan memanfaatkan tipe data string untuk identifikasi kendaraan dan integer untuk pemrosesan waktu serta biaya, algoritma program membagi alur ke dalam tiga kategori kendaraan utama—Motor, Mobil, dan Truk—di mana struktur *switch case* bertugas mengarahkan input pengguna ke kategori yang sesuai atau ke kondisi *default* jika input tidak valid. Di dalam setiap blok *case*, logika *if-else* diterapkan untuk menentukan tarif spesifik: Motor dikenakan Rp7.000 (1-2 jam) atau Rp9.000 (lebih dari 2 jam), Mobil Rp15.000 atau Rp20.000, dan Truk Rp25.000 atau Rp35.000, sehingga biaya akhir kalkulasi selalu akurat sesuai dengan aturan durasi yang telah ditetapkan.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var ph float64

    fmt.Print("Nilai pH: ")

    fmt.Scan(&ph)

    switch {

        case ph < 0 || ph > 14:

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

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

            fmt.Print("Air layak minum")

        default:

            fmt.Print("Air tidak layak minum")

    }

}
```

Screenshot program

The screenshot shows a GoLang IDE interface. On the left, the code editor displays a Go program. On the right, the terminal window shows the execution of the program and its output.

```
1 package main
2 import "fmt"
3
4 func main() {    main redeclared in this block
5     var ph float64
6     fmt.Println("Nilai pH: ")
7     fmt.Scan(&ph)
8
9     switch {
10         case ph < 0 || ph > 14:
11             fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
12         case ph ≥ 6.5 && ph ≤ 8.6:
13             fmt.Println("Air layak minum")
14         default:
15             fmt.Println("Air tidak layak minum")
16     }
17 }
```

Terminal Output:

- PS C:\GoLang> go run tugas1.go
Nilai pH: 8.6
Air layak minum
- PS C:\GoLang> go run tugas1.go
Nilai pH: 9
Air tidak layak minum
- PS C:\GoLang> go run tugas1.go
Nilai pH: 16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.
- PS C:\GoLang>

Deskripsi program

Program ini dirancang untuk mengevaluasi kualitas dan kelayakan konsumsi air berdasarkan indeks derajat keasaman (pH) menggunakan tipe data float64 pada variabel pH guna mengakomodasi nilai desimal. Dengan menerapkan algoritma *switch case*, logika program memetakan input ke dalam beberapa kondisi hierarkis: pertama, mendekripsi anomali input jika nilai berada di luar skala standar 0–14 dengan output "Nilai pH tidak valid"; kedua, mengidentifikasi rentang ideal antara 6.5 hingga 8.6 sebagai kondisi "Air layak minum"; dan ketiga, mengategorikan nilai sisa yang berada di antara skala valid namun di luar rentang aman (bersifat terlalu asam di bawah 6.5 atau terlalu basa di atas 8.6) sebagai status "Air tidak layak minum".

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var nama_kendaraan string

    var jam, tarif, motor, mobil, truk int

    fmt.Scan(&nama_kendaraan, &jam)

    switch nama_kendaraan {

        case "Motor":

            if jam >= 1 {

                tarif = 2000

                motor = jam * tarif

                fmt.Println("Rp.", motor)

            }

        case "Mobil":

            if jam >= 1 {

                tarif = 5000

                mobil = jam * tarif

                fmt.Println("Rp.", mobil)

            }

        case "Truk":

            if jam >= 1 {

                tarif = 8000
```

```

        truk = jam * tarif

        fmt.Print("Rp.", truk)

    }

}

}

```

Screenshot program

The screenshot shows a GoLang IDE interface with the following details:

- Code Editor:** The file `tugas2.go` is open, showing a Go program with a switch statement for vehicle types (Motor, Mobil, Truk) based on time (jam). The code includes imports for `fmt` and a main function.
- Terminal:** The terminal window shows the execution of the program with three different inputs:
 - For Motor: Input "Motor 3 jam", Output "Rp.6000"
 - For Mobil: Input "Mobil 1 jam", Output "Rp.5000"
 - For Truk: Input "Truk 5 jam", Output "Rp.40000"
- Output Panel:** Displays the results of the program execution.
- Status Bar:** Shows the current line (Ln 3, Col 16), character count (61), and file type (Plain text).

```

tugas2.go > main
1 package main
2 import "fmt"
3
4 func main() {    main redeclared in this block
5     var nama_kendaraan string
6     var jam, tarif, motor, mobil, truk int
7     fmt.Scan(&nama_kendaraan, &jam)
8
9     switch nama_kendaraan {
10     case "Motor":
11         if jam ≥ 1 {
12             tarif = 2000
13             motor = jam * tarif
14             fmt.Println("Rp.", motor)
15         }
16     case "Mobil":
17         if jam ≥ 1 {
18             tarif = 5000
19             mobil = jam * tarif
20             fmt.Println("Rp.", mobil)
21         }
22     case "Truk":
23         if jam ≥ 1 {
24             tarif = 8000
25             truk = jam * tarif
26             fmt.Println("Rp.", truk)
27         }
28     }
29 }

```

PROBLEMS 10 OUTPUT TERMINAL DEBUG CONSOLE PORTS

- PS C:\GoLang> go run tugas2.go
Motor 3 jam
Rp.6000
- PS C:\GoLang> go run tugas2.go
Mobil 1 jam
Rp.5000
- PS C:\GoLang> go run tugas2.go
Truk 5 jam
Rp.40000
- PS C:\GoLang>

Deskripsi program

Program ini merupakan pengembangan dari materi *guided* ketiga yang dirancang untuk mengkalkulasi total tarif parkir secara linear berdasarkan durasi penggunaan. Berbeda dengan skema sebelumnya, logika perhitungan di sini menerapkan sistem akumulasi tarif per jam, di mana biaya dasar—Rp2.000 untuk motor, Rp5.000 untuk mobil, dan Rp8.000 untuk truk—akan dikalikan langsung dengan jumlah jam parkir. Untuk mengimplementasikan alur tersebut, program memanfaatkan struktur kontrol *switch case* yang membagi eksekusi ke dalam tiga kategori kendaraan, serta didukung oleh logika *if-then* guna memastikan ketepatan proses aritmatika pada setiap kondisi input.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Scan(&n)

    switch {
    case n % 10 == 0:
        fmt.Println("Kategori: bilangan kelipatan 10")
        bagi := n / 10
        fmt.Println("Hasil pembagian antara", n, "/", "10", "=?", bagi)
    case n % 5 == 0 && n != 5:
        fmt.Println("Kategori: bilangan kelipatan 5")
        pangkat := n * n
        fmt.Println("Hasil kuadrat dari", n, "^2", "=?", pangkat)
    case n % 2 == 1:
        fmt.Println("Kategori: bilangan ganjil")
        tambah := n + 1
        total := n + tambah
        fmt.Println("Hasil penjumlahan dengan bilangan berikutnya", n, "+", tambah, "=?", total)
    case n % 2 == 0:
```

```

        fmt.Println("Kategori: bilangan genap")

        kali := n + 1

        total := n * kali

        fmt.Println("Hasil perkalian dengan bilangan
berikutnya", n, "*", kali, "=", total)

    }

}

```

Screenshot program

The screenshot shows a code editor interface with a dark theme. On the left, the code is written in Go:

```

1 package main
2 import "fmt"
3
4 func main(){    main redeclared in this block
5     var n int
6     fmt.Scan(&n)
7
8     switch {
9     case n % 10 == 0:
10        fmt.Println("Kategori: bilangan kelipatan 10")
11        bagi := n / 10
12        fmt.Println("Hasil pembagian antara", n, "/", "10", "=", bagi)
13    case n % 5 == 0 && n != 5:
14        fmt.Println("Kategori: bilangan kelipatan 5")
15        pangkat := n * n
16        fmt.Println("Hasil kuadrat dari", n,"^2", "=", pangkat )
17    case n % 2 == 1:
18        fmt.Println("Kategori: bilangan ganjil")
19        tambah := n + 1
20        total := n + tambah
21        fmt.Println("Hasil penjumlahan dengan bilangan berikutnya", n, "+", tambah, "=", total)
22    case n % 2 == 0:
23        fmt.Println("Kategori: bilangan genap")
24        kali := n + 1
25        total := n * kali
26        fmt.Println("Hasil perkalian dengan bilangan berikutnya", n, "*", kali, "=", total)
27    }
28 }

```

On the right, the terminal window shows the execution of the program:

- PS C:\GoLang> go run tugas3.go
- 5
- Kategori: bilangan ganjil
- Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
- PS C:\GoLang> go run tugas3.go
- 8
- Kategori: bilangan genap
- Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
- PS C:\GoLang> go run tugas3.go
- 25
- Kategori: bilangan kelipatan 5
- Hasil kuadrat dari 25 ^2 = 625
- PS C:\GoLang> go run tugas3.go
- 20
- Kategori: bilangan kelipatan 10
- Hasil pembagian antara 20 / 10 = 2

Deskripsi program

Program ini dirancang untuk mengklasifikasikan bilangan bulat ke dalam kategori kelipatan 10, kelipatan 5, ganjil, atau genap, serta melakukan operasi aritmatika spesifik berdasarkan klasifikasi tersebut melalui empat skenario logika. Pada kondisi pertama, jika bilangan habis bagi 10, sistem mengidentifikasinya sebagai "Kelipatan

10^n " dan membagi bilangan tersebut dengan 10. Kondisi kedua menangani bilangan yang habis dibagi 5 (kecuali angka 5 itu sendiri) sebagai "Kelipatan 5" dengan tindak lanjut operasi pemangkatan dua (kuadrat). Selanjutnya, kondisi ketiga mendeteksi bilangan ganjil untuk kemudian menjumlahkan bilangan awal dengan bilangan penerusnya ($n + (n+1)$), sedangkan kondisi terakhir mengidentifikasi bilangan genap dan mengalikan bilangan awal dengan bilangan penerusnya ($n * (n+1)$).