

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 11

Switch – Case



Disusun oleh:

ALIF RIZKI PANGESTU

109082530004

S1IF-13-04

**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 terminal window with the following content:

```
laprak2 > go run "d:\laprak2\guided.1.11.go"
1 package main
2
3 import "fmt"
4
5 func main(){
6     var jam24, jam12 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 PORTS

```
PS D:\laprak2> go run "d:\laprak2\guided.1.11.go"
13
1 PM
PS D:\laprak2> go run "d:\laprak2\guided.1.11.go"
0
12 AM
PS D:\laprak2> go run "d:\laprak2\guided.1.11.go"
12
12 PM
PS D:\laprak2>
```

A separate window shows the output of the program:

File	Edit	View	ALIF RIZKI PANGESTU
			109082530004
			S1IF-13-04

Ln 3, Col 11 | 44 character Plain t | 130% Wind UTF-8

Deskripsi program

Program Go pada gambar tersebut berfungsi untuk mengonversi format waktu 24-jam menjadi format 12-jam menggunakan struktur kontrol **switch case**. Di dalam fungsi main, program mendeklarasikan variabel jam24, jam12, dan label, kemudian membaca input angka jam melalui fmt.Scan. Setelah itu nilai jam24 diperiksa dengan switch: jika nilainya **0**, maka jam tengah malam dengan label "**AM**"; jika kurang dari **12**, maka jam tetap sama dan label "**AM**"; jika tepat **12**, maka tetap **12** tetapi label menjadi "**PM**"; dan jika lebih dari **12**, maka jam dikurangi 12 dan diberi label "**PM**". Pada akhir program, hasil konversi—yaitu jam format 12-jam dan label AM/PM—ditampilkan melalui fmt.Println.

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 terminal window and a code editor window.

Terminal Window:

```
laprak2 > go guided.11.2.go
1 package main
2 import "fmt"
3
4 func main(){
5     var nama_tanaman string
6     fmt.Scan(&nama_tanaman)
7     switch nama_tanaman {
8
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 }
```

Code Editor Window:

ALIF RIZKI PANGESTU
109082530004
S1IF-13-04

Ln 3, Col 11 | 44 character Plain text 130% Wind UTF-8

Deskripsi program

Program Go pada gambar tersebut berfungsi untuk mengidentifikasi apakah suatu nama tanaman termasuk tanaman karnivora atau bukan menggunakan struktur switch case. Program dimulai dengan deklarasi variabel `nama_tanaman`, lalu membaca input dari pengguna melalui `fmt.Scan`. Selanjutnya, nilai input diperiksa menggunakan switch: jika pengguna memasukkan "nephentes" atau "drosera", program menampilkan bahwa tanaman tersebut merupakan tanaman karnivora dan asli Indonesia; jika input adalah "venus" atau "sarracenia", program juga menyatakan bahwa tanaman tersebut termasuk tanaman karnivora namun tidak berasal dari Indonesia; sedangkan jika nama tanaman tidak termasuk dalam daftar kasus yang telah ditentukan, bagian default dijalankan sehingga program menampilkan bahwa tanaman tersebut tidak termasuk tanaman karnivora. Hasil output ditampilkan ke layar menggunakan `fmt.Println`.

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 code editor interface with a terminal window below it. The terminal window displays the execution of a Go program named 'guided.11.3.go'. The program prompts the user for the type of vehicle and parking duration, then calculates and prints the parking fee based on a switch-case logic. The terminal also shows the student's name and ID.

```
laprak2 > go guided.11.3.go
func main() {
    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)
}

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Masukkan durasi parkir (dalam jam): 3
Tarif Parkir: Rp 9000
PS D:\laprak2> go run "d:\laprak2\guided.11.3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir (dalam jam): 1
Tarif Parkir: Rp 25000
PS D:\laprak2> go run "d:\laprak2\guided.11.3.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 D:\laprak2>
```

ALIF RIZKI PANGESTU
109082530004
S1IF-13-04

Ln 3, Col 11 | 44 character Plain text 130% Wind UTF-8

Deskripsi program

Program Go di atas merupakan aplikasi sederhana untuk menghitung tarif parkir berdasarkan jenis kendaraan dan lama parkir dalam jam. Program meminta input dari pengguna berupa jenis kendaraan (Motor/Mobil/Truk) dan durasi parkir, kemudian menentukan tarif menggunakan struktur *switch-case* dengan kondisi berbeda untuk tiap jenis kendaraan dan lama parkir. Untuk motor, tarifnya Rp7.000 untuk durasi 1–2 jam dan Rp9.000 jika lebih dari 2 jam; mobil dikenakan Rp15.000 untuk 1–2 jam dan Rp20.000 jika lebih; sedangkan truk dikenai Rp25.000 untuk 1–2 jam dan Rp35.000 jika durasi melebihi 2 jam. Jika pengguna memasukkan kendaraan selain tiga pilihan tersebut atau durasi tidak sesuai logika kondisi, program akan menampilkan bahwa input tidak valid. Setelah menentukan tarif, program menampilkan hasil biaya parkir sesuai aturan yang berlaku.

TUGAS

1. Tugas 1

Source code

```
package main
import "fmt"

func main(){
    var ph float64

    fmt.Print("Masukkan kadar pH air: ")
    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")
    }
}
```

Screenshoot program

The screenshot shows a terminal window and a code editor side-by-side.

Terminal:

```
laprak2 > soal.1.11.go
1 package main
2 import "fmt"
3
4 func main(){
5     var ph float64
6
7     fmt.Print("Masukkan kadar pH air: ")
8     fmt.Scan(&ph)
9
10    switch {
11    case ph < 0 || ph > 14:
12        fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
13    case ph >= 6.5 && ph <= 8.6:
14        fmt.Println("Air layak minum")
15    default:
16        fmt.Println("Air tidak layak minum")
17    }
18 }
```

Code Editor:

The code editor displays the same Go code as the terminal. On the right, there is a status bar showing:

File Edit View ALIF RIZKI PANGESTU 109082530004 S1IF-13-04 Ln 3, Col 11 | 44 character Plain t 100% Wind UTF-8

Deskripsi program

Program Go tersebut berfungsi untuk menentukan kelayakan air berdasarkan nilai pH yang diinput oleh pengguna. Program meminta pengguna memasukkan kadar pH air, kemudian menggunakan struktur *switch-case* untuk mengevaluasi nilai tersebut. Jika nilai pH kurang dari 0 atau lebih dari 14, maka program menampilkan bahwa nilai pH tidak valid karena berada di luar rentang pH standar. Apabila nilai pH berada pada kisaran 6.5 hingga 8.6, air dinyatakan layak minum, sedangkan nilai di luar kisaran tersebut namun masih dalam batas 0–14 akan menghasilkan output bahwa air tidak layak minum. Dengan demikian, program dapat membantu menentukan kualitas air berdasarkan tingkat keasamannya.

2. Tugas 2

Source code

```
package mai

import "fmt"

func main() {
    var jenis string
    var durasi int
    var tarifPerJam 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
    }

    switch jenis {
    case "motor":
        tarifPerJam = 2000
    case "mobil":
        tarifPerJam = 5000
    case "truk":
        tarifPerJam = 8000
    default:
        fmt.Println("Jenis kendaraan tidak valid.")
        return
    }

    totalBiaya := tarifPerJam * durasi
```

```

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

}

```

Screenshot program

The screenshot shows a code editor with a terminal window integrated. The terminal window displays the execution of a Go program named 'soal.2.11.go'. The user inputs 'motor' and '3' for the vehicle type and duration. The program calculates a total parking fee of Rp 6000. When 'mobil' is input instead, it calculates Rp 5000. Finally, 'truk' is input, resulting in a total fee of Rp 40000.

```

laprak2 > soal.2.11.go
5 func main() {
11     fmt.Scan(&jenis)
12
13     fmt.Print("Masukkan durasi parkir (jam): ")
14     fmt.Scan(&durasi)
15
16     if durasi < 1 {
17         durasi = 1
18     }
19
20     switch jenis {
21     case "motor":
22         tarifPerJam = 2000
23     case "mobil":
24         tarifPerJam = 5000
25     case "truk":
26         tarifPerJam = 8000
27     default:
28         fmt.Println("Jenis kendaraan tidak valid.")
29         return
30     }
31
32     totalBiaya := tarifPerJam * durasi
33     fmt.Printf("Total biaya parkir: Rp %d\n", totalBiaya)

```

PROMPTS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

Masukkan jenis kendaraan (motor/mobil/truk): motor
Masukkan durasi parkir (jam): 3
Total biaya parkir: Rp 6000
PS D:\laprak2> go run "d:\laprak2\soal.2.11.go"
Masukkan jenis kendaraan (motor/mobil/truk): mobil
Masukkan durasi parkir (jam): 1
Total biaya parkir: Rp 5000
PS D:\laprak2> go run "d:\laprak2\soal.2.11.go"
Masukkan jenis kendaraan (motor/mobil/truk): truk
Masukkan durasi parkir (jam): 5
Total biaya parkir: Rp 40000

```

File Edit View ALIF RIZKI PANGESTU
109082530004
S1IF-13-04

Ln 3, Col 11 | 44 character Plain t 100% Wind UTF-8

Deskripsi program

Program Go tersebut digunakan untuk menghitung total biaya parkir berdasarkan jenis kendaraan dan lama durasi parkir yang dimasukkan oleh pengguna. Pengguna diminta menginput jenis kendaraan (motor, mobil, atau truk) dan durasi parkir dalam jam, kemudian program memastikan bahwa durasi minimal 1 jam jika pengguna memasukkan angka kurang dari 1. Selanjutnya, struktur *switch-case* digunakan untuk menentukan tarif per jam sesuai jenis kendaraan, yaitu Rp 2000 untuk motor, Rp 5000 untuk mobil, dan Rp 8000 untuk truk; jika jenis tidak dikenali maka program menampilkan pesan error dan berhenti. Setelah tarif ditentukan, program menghitung total biaya parkir dengan mengalikan tarif per jam dan durasi, kemudian menampilkan hasil akhir biaya yang harus dibayar.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Print("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 terminal window with the following content:

```
laprak2 > -o soal.3.11.go
3
4  func main() {
5      var n int
6      fmt.Println("Masukkan angka: ")
7      fmt.Scan(&n)
8
9      switch {
10         case n%10 == 0:
11             fmt.Println("Kategori: Bilangan Kelipatan 10")
12             fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", n, n/10)
13
14         case n%5 == 0 && n != 5:
15             fmt.Println("Kategori: Bilangan Kelipatan 5")
16             fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", n, n*n)
17
18         case n%2 != 0:
19             fmt.Println("Kategori: Bilangan Ganjil")
20             fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", n, n+1, n+(n+1))
21
22         case n%2 == 0:
23             fmt.Println("Kategori: Bilangan Genap")
24             fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", n, n+1, n*(n+1))
25     }
26 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\laprak2> go run "d:\laprak2\soal.3.11.go"
Masukkan angka: 5
Kategori: Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS D:\laprak2> go run "d:\laprak2\soal.3.11.go"
Masukkan angka: 8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS D:\laprak2> go run "d:\laprak2\soal.3.11.go"
Masukkan angka: 25
Kategori: Bilangan Kelipatan 5
```

File Edit View

ALIF RIZKI PANGESTU
109082530004
S1IF-13-04|

Ln 3, Col 11 | 44 character Plain t 100% Wind UTF-8

Deskripsi program

Program Go ini berfungsi untuk mengelompokkan sebuah bilangan berdasarkan karakteristik tertentu lalu menampilkan hasil operasi yang sesuai dengan kategorinya. Pengguna diminta memasukkan sebuah angka dan program memprosesnya menggunakan struktur *switch-case* dengan beberapa kondisi, yaitu jika bilangan merupakan kelipatan 10 maka program menampilkan kategori tersebut dan membagi bilangan dengan 10; jika bilangan merupakan kelipatan 5 namun bukan kelipatan 10, program menampilkannya sebagai kelipatan 5 dan menghitung kuadrat bilangan; jika bilangan ganjil maka program menambahkan angka tersebut dengan angka setelahnya; dan jika bilangan genap maka program mengalikannya dengan bilangan berikutnya. Setiap kondisi menghasilkan output yang berbeda sesuai kategori sehingga program mampu memberikan informasi serta perhitungan berdasarkan sifat bilangan yang diinput.