

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL [No. MODUL]

[NAMA MODUL]



Disusun oleh:

HAFIZD SAMA'I SYAMSI

109082500183

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

```
-GO guided1.go .\ 5 X -GO soal1.go 1 -GO soal2.go
-GO guided1.go >  main
 1 package main
 2 import "fmt"
 3 func main() {
 4
 5     var jam12, jam24 int
 6     var label string
 7     fmt.Scan(&jam24)
 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
18     default:
19         jam12 = jam24 % 12
20         if jam12 == 0 {
21             jam12 = 12
22             label = "AM"
23         } else {
24             label = "PM"
25         }
26     }
27
28     fmt.Println(jam12, label)
29 }
```

PROBLEMS 14 OUTPUT DEBUG CONSOLE TERMINAL

```
PS D:\go\praktek11> go run guided1.go
13
1 PM
PS D:\go\praktek11> go run guided1.go
0
12 AM
PS D:\go\praktek11> go run guided1.go
12
12 PM
PS D:\go\praktek11>
```

Deskripsi program

Program ini dibuat untuk mengkonversi jam dalam format 24 jam menjadi format 12 jam dengan label AM atau PM.

Ketika program dijalankan, pengguna memasukkan sebuah angka (0–23) yang disimpan ke variabel jam24. Setelah itu, program memakai struktur switch tanpa ekspresi untuk mengecek empat kondisi kemungkinan jam:

- Jika $\text{jam24} == 0$

Program menganggap jam 0 sebagai Tengah malam. Jam 0 diubah menjadi 12 am dengan mengisi:

➤ Jam 12 = jam 24

➤ Label = "AM"

- Jika $\text{jam24} < 12$

Untuk jam 1 sampai 11, jam masih berada pada rentang pagi. Angka jam tidak perlu diubah, cukup menambah label AM:

➤ Jam 12 = jam 24

➤ Label = "AM"

- Jika $\text{jam24} == 12$

Jam 12 di format 24 jam brrti 12 siang, sehingga:

➤ Jam 12 = 12

➤ Label = "PM"

- Jika $\text{jam24} > 12$

Untuk jam 13 sampai 23 (sore-malam), jam harus dikoversi menjadi format 12 jam . Cara nya dengan mengurangi 12:

➤ Jam 12 = $\text{jam24} - 12$

➤ Label = "PM"

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

```
-GO guided1.go .\ 5 | -GO soal1.go 1 | -GO soal2.go 1 | -GO soal3.go 5  
-GO guided2.go > ...  
1 package main  
2 import "fmt"  
3 func main() {  
4  
5     var nama_tanaman string  
6     fmt.Scan(&nama_tanaman)  
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    }  
  
PROBLEMS 14 OUTPUT DEBUG CONSOLE TERMINAL PORTS  
  
PS D:\go\praktek11> go run guided2.go  
nepenthes  
Termasuk Tanaman Karnivora.  
Asli Indonesia.  
PS D:\go\praktek11> go run guided2.go  
venus  
Termasuk Tanaman Karnivora.  
Tidak Asli Indonesia.  
PS D:\go\praktek11> go run guided2.go  
karedok  
Tidak termasuk Tanaman Karnivora.  
PS D:\go\praktek11> 
```

Deskripsi program

Program ini membaca sebuah nama tanaman dari input, lalu mengecek jenisnya menggunakan switch. Jika pengguna memasukkan “nepenthes” atau “drosera”, program menganggap tanaman tersebut karnivora dan asli Indonesia, sehingga dua kalimat itu dicetak. Jika yang dimasukkan “venus” atau “sarracenia”, program tetap menyatakan tanaman tersebut karnivora, tetapi bukan asli Indonesia. Selain dari empat nama itu, program langsung menampilkan bahwa tanaman tersebut tidak termasuk tanaman karnivora. Program hanya memilih satu kondisi yang cocok dan mencetak hasil sesuai kategori tanaman yang dikenali.

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

```

go guided3.go > main
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    switch {
12        case kendaraan == "Motor" && durasi >= 1 && durasi <= 2:
13            tarif = 7000
14        case kendaraan == "Motor" && durasi > 2:
15            tarif = 9000
16        case kendaraan == "Mobil" && durasi >= 1 && durasi <= 2:

```

PROBLEMS 15 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

Jenis kendaraan atau durasi parkir tidak valid
Tarif Parkir: Rp 0
PS D:\go\praktek11> go run guided3.go
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 2
Tarif Parkir: Rp 7000
PS D:\go\praktek11> go run guided3.go
Masukkan jenis kendaraan (Motor/Mobil/Truk): Mobil
Masukkan durasi parkir (dalam jam): 4
Tarif Parkir: Rp 20000
PS D:\go\praktek11> go run guided3.go
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir (dalam jam): 1
Tarif Parkir: Rp 25000
PS D:\go\praktek11> 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 D:\go\praktek11>

```

Deskripsi program

Program ini digunakan untuk menghitung tarif parkir berdasarkan jenis kendaraan dan lamanya parkir. Pertama, program meminta pengguna memasukkan jenis kendaraan—Motor, Mobil, atau Truk—lalu membaca durasi parkir dalam jam. Setelah data diterima, program memakai switch tanpa parameter untuk mengecek kombinasi kondisi.

Setiap jenis kendaraan memiliki dua kategori tarif:

- Durasi 1–2 jam, dan
- Durasi lebih dari 2 jam.

Jika kendaraan adalah Motor, tarifnya Rp7.000 untuk 1–2 jam dan Rp9.000 untuk lebih dari 2 jam. Untuk Mobil, tarifnya Rp15.000 dan Rp20.000. Sedangkan Truk dikenai Rp25.000 untuk 1–2 jam dan Rp35.000 bila lewat dari 2 jam.

Jika kombinasi kendaraan atau durasi tidak masuk ke salah satu kondisi tersebut, program menampilkan pesan bahwa data tidak valid. Pada akhir program, tarif yang telah dihitung dicetak dengan format: Tarif Parkir: Rp <tarif>.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var pH float64

    fmt.Println("Masukan nilai pH")

    fmt.Scan(&pH)

    if pH < 0 || pH > 14 {

        fmt.Println("Input tidak valid. rentang pH 0 - 14")

    } else if pH >= 6.5 && pH <= 8.6{

        fmt.Println("Air layak minum")

    } else {

        fmt.Println("Air tidak layak minum")

    }
}
```

Screenshot program

```
Welcome          guided1.go  \ 2      guided2.go +      guided3.go +      Untitled +      guided1.go D:\...praktek10 +  W

soal1.go > main
3 func main() {
6     fmt.Scan(&pH)
7
8     if pH < 0 || pH > 14 {
9         fmt.Println("Input tidak valid. rentang pH 0 - 14")
10    } else if pH >= 6.5 && pH <= 8.6{
11        fmt.Println("Air layak minum")
12    } else {
13        fmt.Println("Air tidak layak minum")
14    }
15 }

PROBLEMS 5   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS
PS D:\go\praktek11> go run soal1.go
Masukan nilai pH 8.6
Air layak minum
PS D:\go\praktek11> go run soal1.go
Masukan nilai pH 9
Air tidak layak minum
PS D:\go\praktek11> go run soal1.go
Masukan nilai pH 16
Input tidak valid. rentang pH 0 - 14
PS D:\go\praktek11> [REDACTED]

File Edit View AA 8
NAMA :HAFIZD SAMA'I SYAMSI
KELAS : SLIF-13-07
NIM : 109082500183
```

Deskripsi program

Program ini membaca sebuah nilai pH dari pengguna, lalu mengecek apakah air tersebut layak untuk diminum. Setelah pengguna memasukkan angka pH, program terlebih dahulu memeriksa apakah nilai tersebut berada di luar rentang 0–14. Jika iya, program menampilkan pesan bahwa input tidak valid karena pH hanya boleh dalam batas tersebut.

Jika nilai pH masih dalam rentang normal, program melanjutkan ke pengecekan berikutnya. Air dianggap layak minum apabila nilai pH berada antara 6.5 hingga 8.6. Jika nilai pH berada di luar rentang itu, program menampilkan bahwa air tersebut tidak layak minum. Dengan urutan pengecekan seperti ini, program memastikan input valid dulu, baru menentukan kelayakan airnya.

2. Tugas 2

Source code

```
package main
import "fmt"
func main() {
    var jenis string
```

```

var jam int

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

fmt.Scan(&jenis)

fmt.Print("Masukan durasi parkir (jam)")

fmt.Scan(&jam)

if jam < 1 {

    jam = 1

}

var tarifPerjam int

switch jenis {

case "motor":

    tarifPerjam = 2000

case "mobil":

    tarifPerjam = 5000

case "truk":

    tarifPerjam = 8000

default:

    fmt.Println("Jenis kendaraan tidak valid!
(gunakan: motor, mobil, atau truk)")

    return

}

total := tarifPerjam * jam

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

```

Screenshoot program

```

    package main
    import "fmt"
    func main() {
        var jenis string
        var jam int
        fmt.Print("Masukan jenis kendaraan (motor/mobil/truk)")
        fmt.Scan(&jenis)
        fmt.Print("Masukan durasi parkir (jam)")
        fmt.Scan(&jam)
        if jam < 1 {
            jam = 1
        }
        tarif := 0
        if jenis == "motor" {
            tarif = jam * 2000
        } else if jenis == "mobil" {
            tarif = jam * 5000
        } else if jenis == "truk" {
            tarif = jam * 8000
        }
        fmt.Println("Total biaya parkir adalah Rp", tarif)
    }

```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

r (jam)Total biaya parkir adalah Rp 6000
PS D:\go\praktek11> go run soal2.go
Masukan jenis kendaraan (motor/mobil/truk) mobil 1 jam
Masukan durasi parkir (jam)Jenis kendaraan tidak valid
PS D:\go\praktek11> go run soal2.go
Masukan jenis kendaraan (motor/mobil/truk) truk 5 jam
Masukan durasi parkir (jam)Total biaya parkir adalah Rp 40000
PS D:\go\praktek11>

```

File Edit View AA

NAMA : HAFIZD SAMA'I SYAMSI
KELAS : S1IF-13-07
NIM : 109082500183

Deskripsi program

Program ini digunakan untuk menghitung biaya parkir berdasarkan jenis kendaraan dan lamanya parkir. Pertama, program meminta pengguna memasukkan jenis kendaraan—motor, mobil, atau truk—serta durasi parkir dalam jam. Jika durasi yang dimasukkan kurang dari 1 jam, program secara otomatis mengubahnya menjadi 1, sehingga perhitungan tarif tidak pernah kurang dari satu jam.

Setelah itu, program menentukan tarif per jam menggunakan switch. Motor dikenakan tarif Rp2.000 per jam, mobil Rp5.000 per jam, dan truk Rp8.000 per jam. Bila pengguna memasukkan jenis kendaraan yang tidak dikenali, program menampilkan pesan kesalahan dan langsung berhenti.

Jika jenis kendaraan valid, program menghitung total biaya dengan mengalikan tarif per jam dengan jumlah jam parkir. Terakhir, hasil perhitungan ditampilkan dalam format: Total biaya parkir adalah Rp <total>.

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 && 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 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

Deskripsi program

Program ini membaca sebuah bilangan dari pengguna dan kemudian mengelompokkannya ke dalam kategori tertentu menggunakan switch berbasis kondisi. Setelah bilangan dimasukkan, program mengecek beberapa aturan secara berurutan.

Pertama, jika bilangan tersebut merupakan kelipatan 10, program menampilkan kategorinya dan langsung menghitung hasil pembagian bilangan itu dengan 10.

Jika bukan kelipatan 10, program mengecek apakah bilangan tersebut merupakan kelipatan 5, tetapi bukan angka 5 itu sendiri. Pada kategori ini, program menghitung dan menampilkan hasil kuadrat dari bilangan yang dimasukkan.

Jika kedua kondisi sebelumnya tidak terpenuhi, program melihat apakah bilangan tersebut genap. Untuk bilangan genap, program mengalikan angka tersebut dengan angka setelahnya dan menampilkan hasil perkalian tersebut.

Apabila seluruh kondisi tidak sesuai, berarti bilangan yang dimasukkan adalah ganjil, sehingga program menjumlahkan bilangan itu dengan bilangan berikutnya dan menampilkan hasilnya.