

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 11
SWITCH - CASE**



Disusun oleh:

FERDINAND AXEL VALERIAN

109082500154

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 jam int
    fmt.Print("Masukkan jam (0-23): ")
    fmt.Scan(&jam)

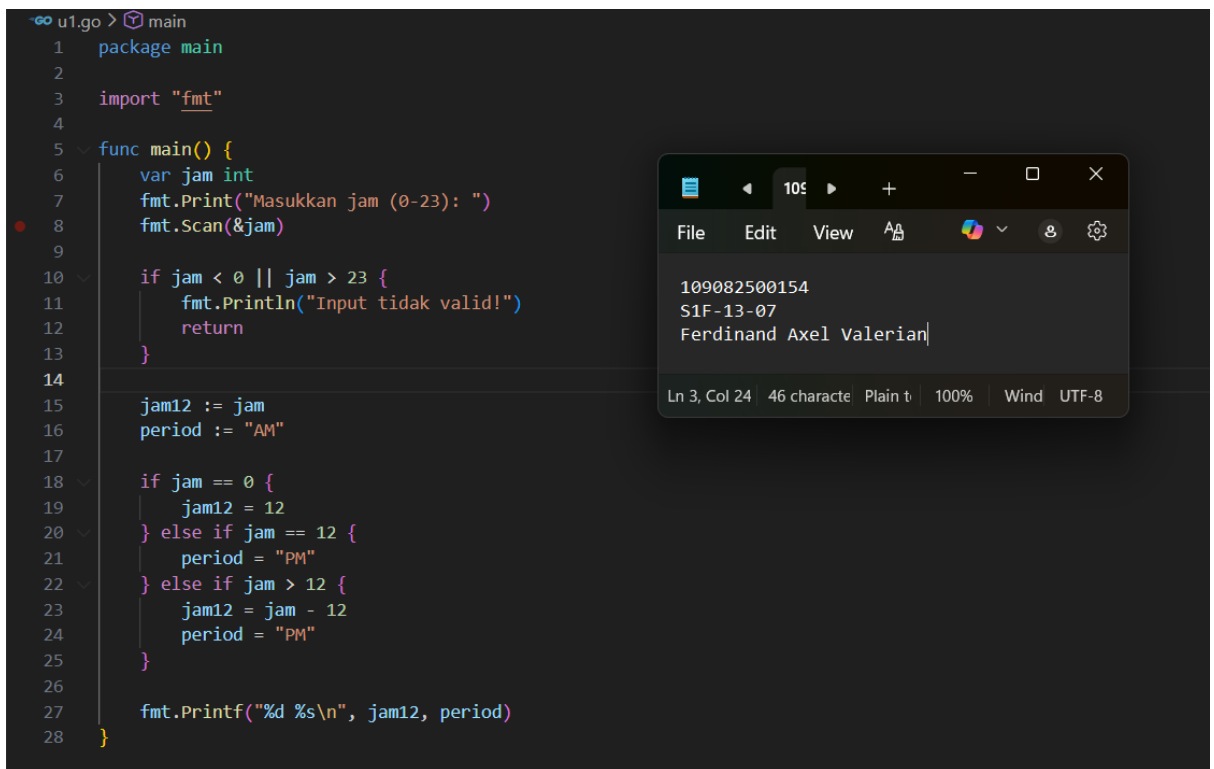
    if jam < 0 || jam > 23 {
        fmt.Println("Input tidak valid!")
        return
    }

    jam12 := jam
    period := "AM"

    if jam == 0 {
        jam12 = 12
    } else if jam == 12 {
        period = "PM"
    } else if jam > 12 {
        jam12 = jam - 12
        period = "PM"
    }

    fmt.Printf("%d %s\n", jam12, period)
}
```

Screenshoot program



```
u1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var jam int
7     fmt.Print("Masukkan jam (0-23): ")
8     fmt.Scan(&jam)
9
10    if jam < 0 || jam > 23 {
11        fmt.Println("Input tidak valid!")
12        return
13    }
14
15    jam12 := jam
16    period := "AM"
17
18    if jam == 0 {
19        jam12 = 12
20    } else if jam == 12 {
21        period = "PM"
22    } else if jam > 12 {
23        jam12 = jam - 12
24        period = "PM"
25    }
26
27    fmt.Printf("%d %s\n", jam12, period)
28 }
```

109082500154
S1F-13-07
Ferdinand Axel Valerian

Ln 3, Col 24 | 46 character | Plain text | 100% | Window | UTF-8

```
● PS D:\vscode\College\Modul 11> go run u1.go
Masukkan jam (0-23): 13
1 PM
● PS D:\vscode\College\Modul 11> go run u1.go
Masukkan jam (0-23): 0
12 AM
● PS D:\vscode\College\Modul 11> go run u1.go
Masukkan jam (0-23): 12
12 PM
❖ PS D:\vscode\College\Modul 11> 
```

Deskripsi program

Program ini dibuat untuk mengkonversi jam dari 24 jam menjadi 12 jam. Dengan cara anda menginput suatu bilangan dalam 24 jam, lalu program akan memproses inputan yang anda input. Output akan menghasilkan jam dalam 12 jam bukan dalam 24 jam. Output juga akan menunjukkan apakah itu AM (pagi) atau PM (siang/sore/malam).

2. Guided 2

Source Code

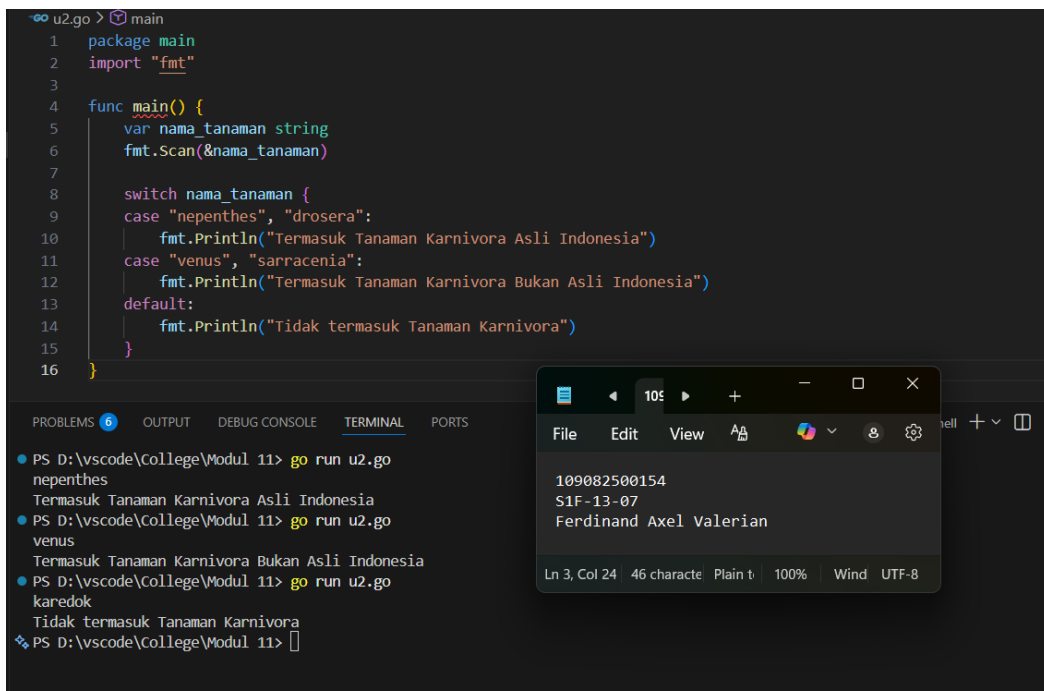
```
package main

import "fmt"

func main() {
    var nama_tanaman string
    fmt.Scan(&nama_tanaman)

    switch nama_tanaman {
    case "nepenthes", "drosera":
        fmt.Println("Termasuk Tanaman Karnivora Asli Indonesia")
    case "venus", "sarracenia":
        fmt.Println("Termasuk Tanaman Karnivora Bukan Asli Indonesia")
    default:
        fmt.Println("Tidak termasuk Tanaman Karnivora")
    }
}
```

Screenshoot program



```
u2.go > main
1 package main
2 import "fmt"
3
4 func main() {
5     var nama_tanaman string
6     fmt.Scan(&nama_tanaman)
7
8     switch nama_tanaman {
9     case "nepenthes", "drosera":
10         fmt.Println("Termasuk Tanaman Karnivora Asli Indonesia")
11     case "venus", "sarracenia":
12         fmt.Println("Termasuk Tanaman Karnivora Bukan Asli Indonesia")
13     default:
14         fmt.Println("Tidak termasuk Tanaman Karnivora")
15     }
16 }
```

PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS D:\vscode\College\Modul 11> go run u2.go
nepenthes
Termasuk Tanaman Karnivora Asli Indonesia
- PS D:\vscode\College\Modul 11> go run u2.go
venus
Termasuk Tanaman Karnivora Bukan Asli Indonesia
- PS D:\vscode\College\Modul 11> go run u2.go
karedok
Tidak termasuk Tanaman Karnivora

PS D:\vscode\College\Modul 11>

File Edit View 105
109082500154
S1F-13-07
Ferdinand Axel Valerian
Ln 3, Col 24 46 character Plain text 100% Window UTF-8

Deskripsi program

Program ini digunakan untuk mengidentifikasi apakah suatu tanaman termasuk jenis karnivora dan menentukan asal-usulnya dari Indonesia atau bukan. Pengguna cukup memasukkan nama tanaman, kemudian program akan memberikan informasi apakah tanaman tersebut karnivora dan asal wilayahnya.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var bilangan int

    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&bilangan)

    switch {
    case bilangan%10 == 0:
        hasil := float64(bilangan) / 10.0
        fmt.Printf("Kategori: Bilangan Kelipatan 10\n")
        fmt.Printf("Hasil pembagian antara %d / 10 = %.0f\n", bilangan, hasil)

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

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

```

default:

    hasil := bilangan + (bilangan + 1)

    fmt.Printf("Kategori: Bilangan Ganjil\n")

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

                bilangan, bilangan+1, hasil)

    }

}

```

Screenshoot program

The screenshot shows a Go program in a code editor (VS Code) and its terminal output. The code is a switch statement that categorizes a number based on its remainder when divided by 10, 5, or 2. The terminal output shows the program running and displaying the category and result for the input 109082500154.

```

1 package main
2 import "fmt"
3
4 func main() {
5     var bilangan int
6
7     fmt.Print("Masukkan bilangan: ")
8     fmt.Scan(&bilangan)
9
10    switch {
11    case bilangan%10 == 0:
12        hasil := float64(bilangan) / 10.0
13        fmt.Printf("Kategori: Bilangan Kelipatan 10\n")
14        fmt.Printf("Hasil pembagian antara %d / 10 = %.0f\n", bilangan, hasil)
15
16    case bilangan%5 == 0:
17        hasil := bilangan * bilangan
18        fmt.Printf("Kategori: Bilangan Kelipatan 5\n")
19        fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", bilangan, hasil)
20
21    case bilangan%2 == 0:
22        hasil := bilangan * (bilangan + 1)
23        fmt.Printf("Kategori: Bilangan Genap\n")
24        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n",
25                bilangan, bilangan+1, hasil)
26
27    default:
28        hasil := bilangan + (bilangan + 1)
29        fmt.Printf("Kategori: Bilangan Ganjil\n")
30        fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n",
31                bilangan, bilangan+1, hasil)
32    }
33 }

```

Terminal Output:

```

109082500154
S1F-13-07
Ferdinand Axel Valerian
Ln 3, Col 24 46 character Plain text 100% Wind UTF-8

```

Deskripsi program

Program ini dibuat untuk menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parkir. Terdapat tiga jenis kendaraan dengan tarif berbeda: motor dikenakan tarif Rp 7.000 untuk 1-2 jam dan Rp 9.000 jika lebih dari 2 jam; mobil dikenakan tarif Rp 15.000 untuk 1-2 jam dan Rp 20.000 jika lebih dari 2 jam; truk dikenakan tarif Rp 25.000 untuk 1-2 jam dan Rp 35.000 jika lebih dari 2 jam. Program menerima input jenis kendaraan dan durasi parkir dalam jam, kemudian mengeluarkan tarif yang sesuai atau pesan error jika input tidak valid.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var pH float64

    fmt.Print("Masukkan kadar pH air: ")
    fmt.Scan(&pH)

    if pH < 0 || pH > 14 {
        fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
    } else if pH >= 6.5 && pH <= 8.6 {
        fmt.Println("Air layak minum")
    } else {
        fmt.Println("Air tidak layak minum")
    }
}
```

Screenshoot program

```
ti.go > main
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    if pH < 0 || pH > 14 {
11        fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
12    } else if pH >= 6.5 && pH <= 8.6 {
13        fmt.Println("Air layak minum")
14    } else {
15        fmt.Println("Air tidak layak minum")
16    }
17 }
```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS D:\vscode\College\Modul 11> go run ti.go
Masukkan kadar pH air: 8.6
Air layak minum
- PS D:\vscode\College\Modul 11> go run ti.go
Masukkan kadar pH air: 9
Air tidak layak minum
- PS D:\vscode\College\Modul 11> go run ti.go
Masukkan kadar pH air: 16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.

PS D:\vscode\College\Modul 11>

Deskripsi program

Program ini digunakan untuk menentukan apakah air dengan kadar pH tertentu layak diminum atau tidak. Program menerima input berupa bilangan desimal yang mewakili kadar pH air, kemudian melakukan pengecekan terhadap rentang pH yang ditentukan. Air dianggap layak minum jika memiliki kadar pH antara 6.5 hingga 8.6 (inklusif). Jika pH di bawah 6.5 atau di atas 8.6, air dianggap tidak layak minum. Program juga memvalidasi input agar berada dalam rentang pH normal yaitu 0 hingga 14, dan akan memberikan pesan error jika input di luar rentang tersebut.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var jenis string

    var jam int

    fmt.Print("Masukkan jenis kendaraan dan durasi (contoh: motor 3): ")

    fmt.Scan(&jenis, &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")
    return
}

total := tarifPerJam * jam
fmt.Printf("Rp %d\n", total)
}
```

Screenshoot program

The image shows a Go program in a VS Code editor and its execution output in the terminal. The program calculates the total parking fee based on the vehicle type and duration. The terminal shows three test cases: motor (Rp 6000), mobil (Rp 5000), and truk (Rp 40000).

```
t2.go > main
1 package main
2 import "fmt"
3
4 func main() {
5     var jenis string
6     var jam int
7
8     fmt.Print("Masukkan jenis kendaraan dan durasi (contoh: motor 3): ")
9     fmt.Scan(&jenis, &jam)
10
11     if jam < 1 {
12         jam = 1
13     }
14
15     var tarifPerJam int
16
17     switch jenis {
18     case "motor":
19         tarifPerJam = 2000
20     case "mobil":
21         tarifPerJam = 5000
22     case "truk":
23         tarifPerJam = 8000
24     default:
25         fmt.Println("Jenis kendaraan tidak valid")
26         return
27     }
28
29     total := tarifPerJam * jam
30     fmt.Printf("Rp %d\n", total)
31 }
```

Terminal Output:

```
PS D:\vscode\College\Modul 11> go run t2.go
Masukkan jenis kendaraan dan durasi (contoh: motor 3): motor 3
Rp 6000
PS D:\vscode\College\Modul 11> go run t2.go
Masukkan jenis kendaraan dan durasi (contoh: motor 3): mobil 1
Rp 5000
PS D:\vscode\College\Modul 11> go run t2.go
Masukkan jenis kendaraan dan durasi (contoh: motor 3): truk 5
Rp 40000
PS D:\vscode\College\Modul 11>
```

Deskripsi program

Program ini menghitung total biaya parkir berdasarkan jenis kendaraan dan durasi parkir. Terdapat tiga jenis kendaraan dengan tarif per jam yang berbeda: motor dikenakan tarif Rp 2.000 per jam, mobil dikenakan tarif Rp 5.000 per jam, dan truk dikenakan tarif Rp 8.000 per jam. Program menerima input jenis kendaraan dan jumlah jam parkir, kemudian menghitung total biaya dengan mengalikan tarif per jam dengan durasi parkir. Jika durasi parkir kurang dari 1 jam, program akan menganggapnya sebagai 1 jam. Hasil keluaran berupa total biaya parkir dalam format mata uang Rupiah.

3. Tugas 3

Source code

```
package main
import "fmt"
```

```

func main() {
    var a int
    fmt.Print("Masukkan sebuah bilangan (ganjil, genap,
kelipatan 5, kelipatan 10) : ")
    fmt.Scanln(&a)

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

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

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

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

    default:
        fmt.Println("Bilangan tidak termasuk kategori yang
ditentukan.")
    }
}

```

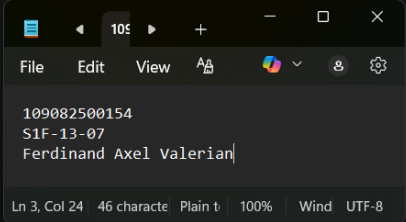
Screenshoot program

```

t3.go > main
1 package main
2 import "fmt"
3
4 func main() {
5     var a int
6     fmt.Print("Masukkan sebuah bilangan (ganjil, genap, kelipatan 5, kelipatan 10) : ")
7     fmt.Scanln(&a)
8
9     switch {
10    case a %2 != 0:
11        fmt.Println("Kategori: Bilangan Ganjil")
12        hasil := a + (a + 1)
13        fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", a , a +1, hasil)
14
15    case a %2 == 0:
16        fmt.Println("Kategori: Bilangan Genap")
17        hasil := a * (a + 1)
18        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", a , a +1, hasil)
19
20    case a %5 == 0:
21        fmt.Println("Kategori: Bilangan Kelipatan 5")
22        hasil := a ^ 2
23        fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", a , hasil)
24
25    case a %10 == 0:
26        fmt.Println("Kategori: Bilangan Kelipatan 10")
27        hasil := a / 10
28        fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", a , hasil)
29
30    default:
31        fmt.Println("Bilangan tidak termasuk kategori yang ditentukan.")
32    }
33 }

```

```
PS D:\vscode\College\Modul 11> go run t3.go
Masukkan sebuah bilangan (ganjil, genap, kelipatan 5, kelipatan 10) : 5
Kategori: Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS D:\vscode\College\Modul 11> go run t3.go
Masukkan sebuah bilangan (ganjil, genap, kelipatan 5, kelipatan 10) : 8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS D:\vscode\College\Modul 11> go run t3.go
Masukkan sebuah bilangan (ganjil, genap, kelipatan 5, kelipatan 10) : 25
Kategori: Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 25 + 26 = 51
PS D:\vscode\College\Modul 11> go run t3.go
Masukkan sebuah bilangan (ganjil, genap, kelipatan 5, kelipatan 10) : 20
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 20 * 21 = 420
PS D:\vscode\College\Modul 11> 
```



Deskripsi program

Program akan meminta pengguna memasukkan sebuah bilangan, kemudian menggunakan struktur switch tanpa ekspresi untuk mengevaluasi kondisi secara berurutan. Kode ini mengecek apakah bilangan tersebut ganjil terlebih dahulu, kemudian genap, lalu kelipatan 5, dan terakhir kelipatan 10. Namun, terdapat kelemahan dalam urutan pengecekan ini karena bilangan kelipatan 5 atau 10 yang juga merupakan bilangan ganjil atau genap akan diklasifikasikan secara keliru. Misalnya, bilangan 5 (yang merupakan ganjil dan kelipatan 5) akan diproses sebagai bilangan ganjil dan dijumlahkan dengan bilangan berikutnya, padahal seharusnya dihitung kuadratnya sesuai aturan kelipatan 5. Selain itu, terdapat kesalahan sintaksis karena operator `^` digunakan untuk operasi kuadrat, yang sebenarnya dalam Go adalah operator XOR, sehingga perhitungan kuadrat akan menghasilkan nilai yang salah.