

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 11
SWITCH CASE**



Disusun oleh:

FAREL TRI JULIAN

109082500163

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

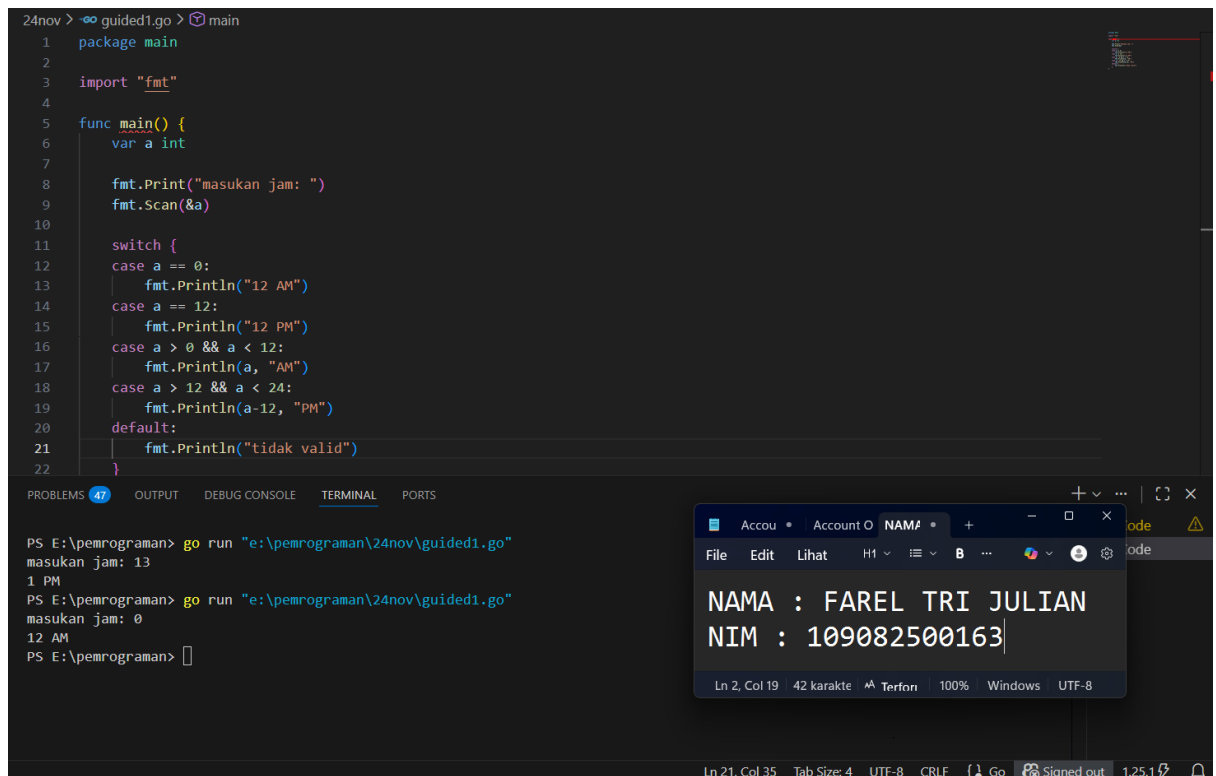
Source Code

```
package main

import "fmt"

func main() {
    var a int
    fmt.Print("masukan jam: ")
    fmt.Scan(&a)
    switch {
    case a == 0:
        fmt.Println("12 AM")
    case a == 12:
        fmt.Println("12 PM")
    case a > 0 && a < 12:
        fmt.Println(a, "AM")
    case a > 12 && a < 24:
        fmt.Println(a-12, "PM")
    default:
        fmt.Println("tidak valid")
    }
}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The program is a simple time converter that takes an integer input and outputs the time in 12-hour AM/PM format. The terminal shows the program being run three times with different inputs: 13, 0, and 0. The output for each run is: 1 PM, 12 AM, and 12 AM respectively. A small window in the foreground displays the output of the program: NAMA : FAREL TRI JULIAN and NIM : 109082500163.

```
24nov > go guided1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var a int
7
8     fmt.Print("masukan jam: ")
9     fmt.Scan(&a)
10
11     switch {
12     case a == 0:
13         fmt.Println("12 AM")
14     case a == 12:
15         fmt.Println("12 PM")
16     case a > 0 && a < 12:
17         fmt.Println(a, "AM")
18     case a > 12 && a < 24:
19         fmt.Println(a-12, "PM")
20     default:
21         fmt.Println("tidak valid")
22     }
23 }
```

```
PS E:\pemrograman> go run "e:\pemrograman\24nov\guided1.go"
masukan jam: 13
1 PM
PS E:\pemrograman> go run "e:\pemrograman\24nov\guided1.go"
masukan jam: 0
12 AM
PS E:\pemrograman>
```

NAMA : FAREL TRI JULIAN
NIM : 109082500163

Deskripsi program

Program ini mengonversi input waktu dalam format 24 jam (integer) menjadi format 12 jam (AM/PM) menggunakan struktur kontrol switch case. Setelah menerima input angka dari pengguna yang disimpan dalam variabel `a`, program mengevaluasi nilai tersebut dengan beberapa kondisi: angka 0 dikonversi menjadi "12 AM", angka 12 menjadi "12 PM", rentang 1-11 dicetak langsung dengan akhiran "AM", dan rentang 13-23 nilainya dikurangi 12 lalu dicetak dengan akhiran "PM". Jika input pengguna berada di luar rentang 0 hingga 23, blok default akan dieksekusi untuk menampilkan pesan "tidak valid" sebagai penanganan kesalahan.

2. Guided 2

Source Code

```
package main

import "fmt"

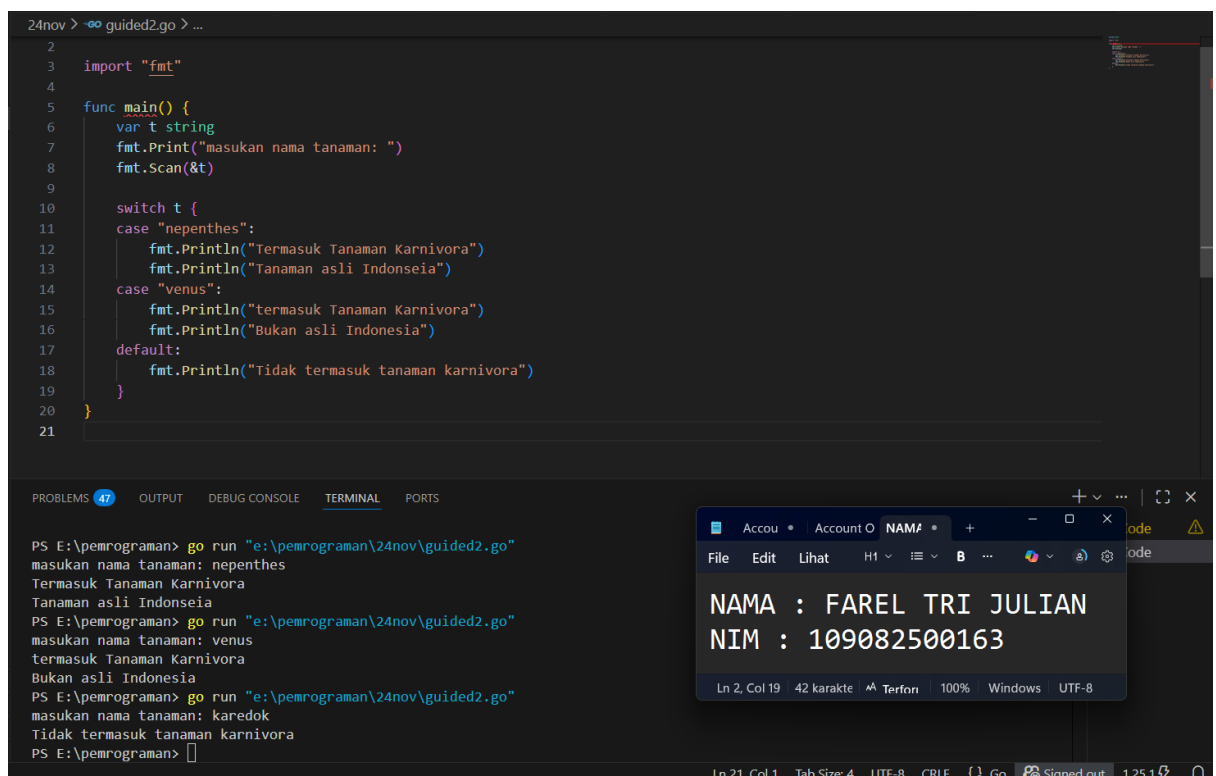
func main() {
    var t string

    fmt.Print("masukan nama tanaman: ")

    fmt.Scan(&t)

    switch t {
    case "nepenthes":
        fmt.Println("Termasuk Tanaman Karnivora")
        fmt.Println("Tanaman asli Indonseia")
    case "venus":
        fmt.Println("termasuk Tanaman Karnivora")
        fmt.Println("Bukan asli Indonesia")
    default:
        fmt.Println("Tidak termasuk tanaman karnivora")
    }
}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The program uses a switch statement to categorize plants based on their names.

```
24nov > -go guided2.go > ...
2
3 import "fmt"
4
5 func main() {
6     var t string
7     fmt.Print("masukan nama tanaman: ")
8     fmt.Scan(&t)
9
10    switch t {
11    case "nepenthes":
12        fmt.Println("Termasuk Tanaman Karnivora")
13        fmt.Println("Tanaman asli Indonseia")
14    case "venus":
15        fmt.Println("termasuk Tanaman Karnivora")
16        fmt.Println("Bukan asli Indonesia")
17    default:
18        fmt.Println("Tidak termasuk tanaman karnivora")
19    }
20 }
21
```

The terminal output shows the program being run three times with different inputs:

```
PS E:\pemrograman> go run "e:\pemrograman\24nov\guided2.go"
masukan nama tanaman: nepenthes
Termasuk Tanaman Karnivora
Tanaman asli Indonseia
PS E:\pemrograman> go run "e:\pemrograman\24nov\guided2.go"
masukan nama tanaman: venus
termasuk Tanaman Karnivora
Bukan asli Indonesia
PS E:\pemrograman> go run "e:\pemrograman\24nov\guided2.go"
masukan nama tanaman: karedok
Tidak termasuk tanaman karnivora
PS E:\pemrograman>
```

Deskripsi program

Program ini bertujuan untuk mengidentifikasi kategori dan asal tanaman berdasarkan input nama yang diberikan pengguna menggunakan struktur kontrol switch. Variabel string `t` digunakan untuk menampung input, yang kemudian dicocokkan dengan dua kasus spesifik: jika input adalah "nepenthes", program mencetak informasi bahwa tanaman tersebut adalah karnivora asli Indonesia, sedangkan jika "venus", program mencetak bahwa itu adalah tanaman karnivora yang bukan asli Indonesia. Apabila input tidak sesuai dengan kedua nama tersebut, blok default akan berjalan dan menampilkan pesan umum bahwa tanaman tersebut tidak termasuk jenis karnivora.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var kendaraan string
    var jam int

    fmt.Print("masukan jenis kendaraan
(mototr/mobil/truk): ")

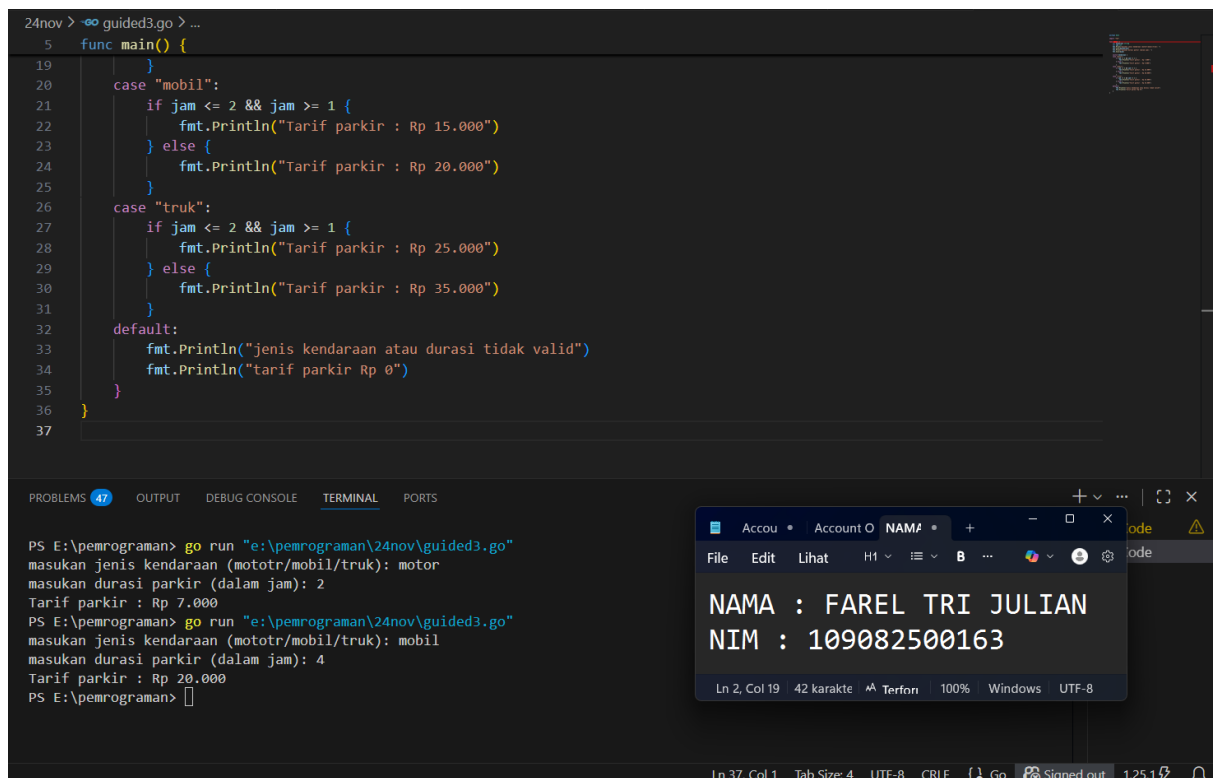
    fmt.Scan(&kendaraan)

    fmt.Print("masukan durasi parkir (dalam jam): ")
    fmt.Scan(&jam)

    switch kendaraan {
    case "motor":
        if jam <= 2 && jam >= 1 {
            fmt.Println("Tarif parkir : Rp 7.000")
        } else {
            fmt.Println("Tarif parkir : Rp 9.000")
        }
    case "mobil":
        if jam <= 2 && jam >= 1 {
            fmt.Println("Tarif parkir : Rp 15.000")
        } else {
            fmt.Println("Tarif parkir : Rp 20.000")
        }
    case "truk":
        if jam <= 2 && jam >= 1 {
            fmt.Println("Tarif parkir : Rp 25.000")
        } else {
            fmt.Println("Tarif parkir : Rp 35.000")
        }
    default:
```

```
        fmt.Println("jenis kendaraan atau durasi tidak  
valid")  
        fmt.Println("tarif parkir Rp 0")  
    }  
}
```

Screenshoot program



The screenshot shows a Go program in a VS Code editor and its execution in a terminal. The program is a parking fee calculator that takes vehicle type and duration as input and outputs the corresponding fee.

```
24nov > -eo guided3.go > ...
5 func main() {
19 }
20 case "mobil":
21     if jam <= 2 && jam >= 1 {
22         fmt.Println("Tarif parkir : Rp 15.000")
23     } else {
24         fmt.Println("Tarif parkir : Rp 20.000")
25     }
26 case "truk":
27     if jam <= 2 && jam >= 1 {
28         fmt.Println("Tarif parkir : Rp 25.000")
29     } else {
30         fmt.Println("Tarif parkir : Rp 35.000")
31     }
32 default:
33     fmt.Println("jenis kendaraan atau durasi tidak valid")
34     fmt.Println("tarif parkir Rp 0")
35 }
36 }
37 }
```

The terminal output shows the program being run twice. In the first run, the user inputs 'motor' and '2', resulting in a parking fee of Rp 7.000. In the second run, the user inputs 'mobil' and '4', resulting in a parking fee of Rp 20.000.

```
PS E:\pemrograman> go run "e:\pemrograman\24nov\guided3.go"
masukan jenis kendaraan (mototr/mobil/truk): motor
masukan durasi parkir (dalam jam): 2
Tarif parkir : Rp 7.000
PS E:\pemrograman> go run "e:\pemrograman\24nov\guided3.go"
masukan jenis kendaraan (mototr/mobil/truk): mobil
masukan durasi parkir (dalam jam): 4
Tarif parkir : Rp 20.000
PS E:\pemrograman>
```

Deskripsi program

Program ini dirancang untuk menghitung biaya parkir berdasarkan dua input pengguna, yaitu jenis kendaraan (motor, mobil, atau truk) dan durasi parkir dalam satuan jam. Kode ini menggunakan struktur kontrol utama switch untuk mendeteksi jenis kendaraan, yang di dalamnya terdapat logika percabangan bersarang (nested if-else) untuk menentukan tarif spesifik berdasarkan lama waktu parkir; durasi 1 hingga 2 jam dikenakan tarif dasar, sedangkan durasi lainnya (biasanya lebih lama) dikenakan tarif yang lebih tinggi. Jika jenis kendaraan yang diinput tidak sesuai dengan tiga pilihan yang tersedia, blok default akan dieksekusi untuk memberikan notifikasi bahwa input tidak valid dan menetapkan tarif menjadi Rp0.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var ph float64

    fmt.Print("masukan kadar pH: ")

    fmt.Scan(&ph)

    switch {
    case ph >= 6.5 && ph <= 8.6:
        fmt.Println("Air layak minum")
    case ph < 6.5 || ph > 8.6:
        fmt.Println("Air tidak layak minum")
    case ph > 14:
        fmt.Println("Input tidak valid")
    }
}
```

Screenshoot program

The screenshot shows a Go program in a dark-themed editor. The program is named `latihan1.go` and is in the `main` package. It imports the `fmt` package and defines a `main` function. Inside the function, a `float64` variable `ph` is declared. The program prompts the user to enter a pH value and reads it. It then uses a `switch` statement to check the pH value. If the pH is between 6.5 and 8.6, it prints "Air layak minum". If the pH is less than 6.5 or greater than 8.6, it prints "Air tidak layak minum". If the pH is greater than 14, it prints "Input tidak valid".

```
24nov > go latihan1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var ph float64
7
8     fmt.Print("masukan kadar pH: ")
9     fmt.Scan(&ph)
10
11     switch {
12     case ph >= 6.5 && ph <= 8.6:
13         fmt.Println("Air layak minum")
14     case ph < 6.5 || ph > 8.6:
15         fmt.Println("Air tidak layak minum")
16     case ph > 14:
17         fmt.Println("Input tidak valid")
18     }
19 }
20
```

The terminal output shows the program being run from the `PS E:\pemrograman>` prompt. The user enters `8.6`, and the program outputs `Air layak minum`. The user then enters `9`, and the program outputs `Air tidak layak minum`. The user enters `16`, and the program outputs `Air tidak layak minum`. The user enters `15`, and the program outputs `Air tidak layak minum`.

```
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan1.go"
masukan kadar pH: 8.6
Air layak minum
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan1.go"
masukan kadar pH: 9
Air tidak layak minum
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan1.go"
masukan kadar pH: 16
Air tidak layak minum
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan1.go"
masukan kadar pH: 15
Air tidak layak minum
```

Deskripsi program

Program ini berfungsi untuk menentukan kelayakan konsumsi air berdasarkan input kadar pH bertipe desimal (`float64`) menggunakan struktur kontrol `switch` tanpa ekspresi awal. Kode ini mengevaluasi input dengan logika boolean: jika nilai pH berada dalam rentang 6.5 hingga 8.6, program mencetak "Air layak minum", sedangkan jika nilai berada di luar rentang tersebut (kurang dari 6.5 atau lebih dari 8.6), program mencetak "Air tidak layak minum". Terdapat juga baris kode yang dimaksudkan untuk menangani input di atas 14 sebagai "Input tidak valid".

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var kendaraan string

    var jam int

    fmt.Scan(&kendaraan, &jam)

    switch kendaraan {

    case "motor":

        tarif := 2000 * jam

        fmt.Println("Rp", tarif)

    case "mobil":

        tarif := 5000 * jam

        fmt.Println("Rp", tarif)

    case "truk":

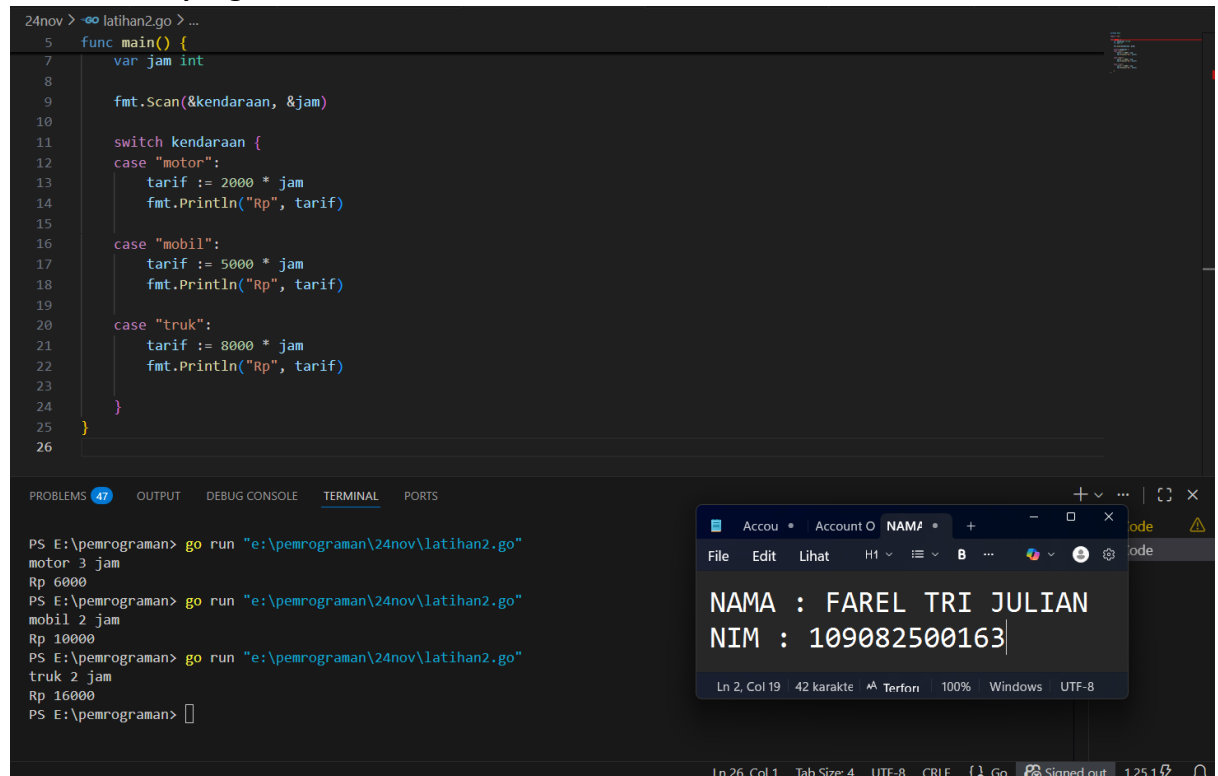
        tarif := 8000 * jam

        fmt.Println("Rp", tarif)

    }

}
```

Screenshoot program



```
24nov > -o latihan2.go > ...
5 func main() {
7     var jam int
8
9     fmt.Scan(&kendaraan, &jam)
10
11     switch kendaraan {
12     case "motor":
13         tarif := 2000 * jam
14         fmt.Println("Rp", tarif)
15
16     case "mobil":
17         tarif := 5000 * jam
18         fmt.Println("Rp", tarif)
19
20     case "truk":
21         tarif := 8000 * jam
22         fmt.Println("Rp", tarif)
23     }
24 }
25
26
```

```
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan2.go"
motor 3 jam
Rp 6000
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan2.go"
mobil 2 jam
Rp 10000
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan2.go"
truk 2 jam
Rp 16000
PS E:\pemrograman>
```

The screenshot shows a Go program in VS Code that calculates parking fees based on vehicle type and duration. The program uses a switch statement to handle three vehicle types: motor (Rp2,000/hour), mobil (Rp5,000/hour), and truk (Rp8,000/hour). The terminal output shows three successful runs with the correct calculations. A small window in the foreground displays the user's name and NIM.

Deskripsi program

Program pini berfungsi sebagai **kalkulator tarif parkir sederhana** yang menghitung total biaya secara linear berdasarkan jenis kendaraan dan durasi parkir. Program membaca dua input sekaligus dari pengguna, yaitu jenis kendaraan (string) dan lama waktu parkir dalam jam (integer), lalu menggunakan struktur kontrol switch untuk menentukan tarif per jam yang berlaku: Rp2.000 untuk motor, Rp5.000 untuk mobil, dan Rp8.000 untuk truk. Di dalam setiap kasus, program mengalikan tarif dasar tersebut dengan durasi jam yang diinputkan dan langsung mencetak total biayanya ke layar, namun program ini belum memiliki penanganan kesalahan (blok default) jika jenis kendaraan yang dimasukkan tidak sesuai dengan ketiga pilihan tersebut.

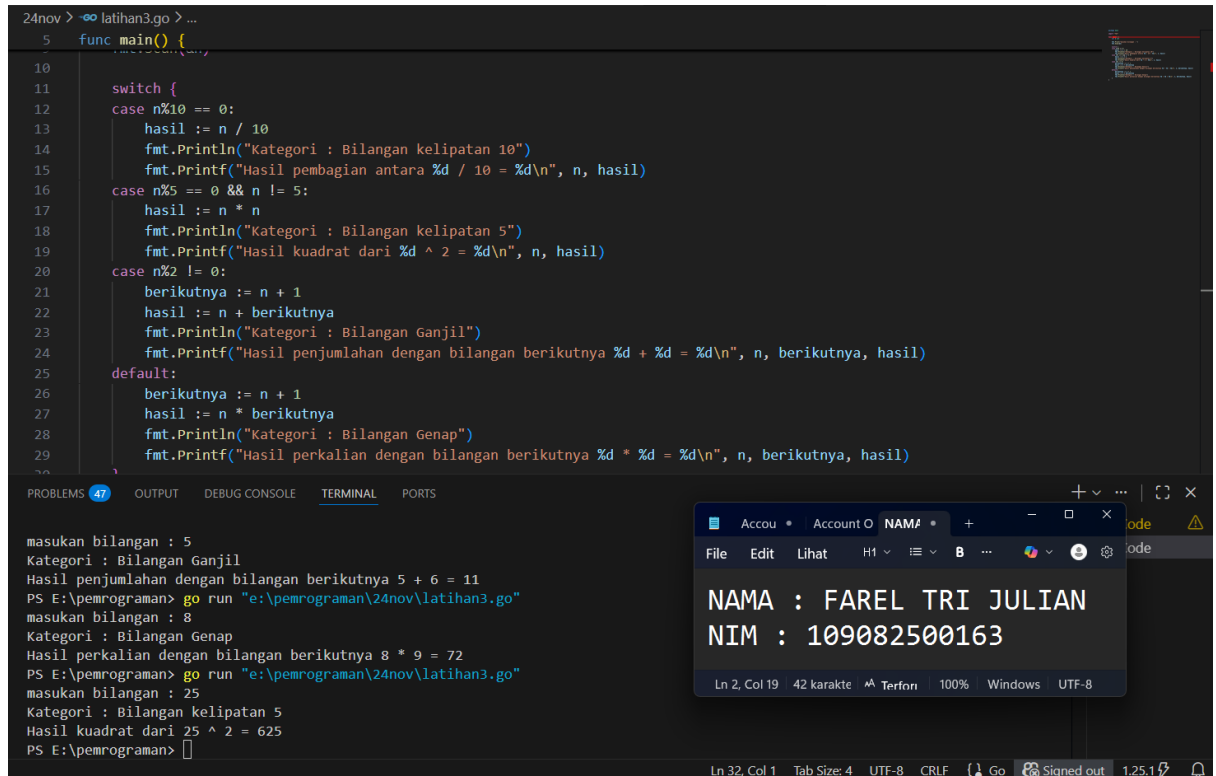
3. Tugas 3

Source code

```
package main
import "fmt"
func main() {
    var n int
    fmt.Print("masukan bilangan : ")
    fmt.Scan(&n)

    switch {
    case n%10 == 0:
        hasil := n / 10
        fmt.Println("Kategori : Bilangan kelipatan 10")
        fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", n, hasil)
    case n%5 == 0 && n != 5:
        hasil := n * n
        fmt.Println("Kategori : Bilangan kelipatan 5")
        fmt.Printf("Hasil kuadrat dari %d ^ 2 = %d\n", n, hasil)
    case n%2 != 0:
        berikutnya := n + 1
        hasil := n + berikutnya
        fmt.Println("Kategori : Bilangan Ganjil")
        fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", n, berikutnya, hasil)
    default:
        berikutnya := n + 1
        hasil := n * berikutnya
        fmt.Println("Kategori : Bilangan Genap")
        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", n, berikutnya, hasil)
    }
}
```

Screenshoot program



```
24nov > go latihan3.go > ...
5 func main() {
10
11     switch {
12     case n%10 == 0:
13         hasil := n / 10
14         fmt.Println("Kategori : Bilangan kelipatan 10")
15         fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", n, hasil)
16     case n%5 == 0 && n != 5:
17         hasil := n * n
18         fmt.Println("Kategori : Bilangan kelipatan 5")
19         fmt.Printf("Hasil kuadrat dari %d ^ 2 = %d\n", n, hasil)
20     case n%2 != 0:
21         berikutnya := n + 1
22         hasil := n + berikutnya
23         fmt.Println("Kategori : Bilangan Ganjil")
24         fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", n, berikutnya, hasil)
25     default:
26         berikutnya := n + 1
27         hasil := n * berikutnya
28         fmt.Println("Kategori : Bilangan Genap")
29         fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", n, berikutnya, hasil)
30 }
31 }
```

masukan bilangan : 5
Kategori : Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan3.go"
masukan bilangan : 8
Kategori : Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS E:\pemrograman> go run "e:\pemrograman\24nov\latihan3.go"
masukan bilangan : 25
Kategori : Bilangan kelipatan 5
Hasil kuadrat dari 25 ^ 2 = 625
PS E:\pemrograman>

NAMA : FAREL TRI JULIAN
NIM : 109082500163

Deskripsi program

Program ini memproses sebuah bilangan bulat (integer) dari input pengguna untuk menentukan kategori dan melakukan operasi aritmatika spesifik menggunakan struktur kontrol switch dengan evaluasi berjenjang. Logika program bekerja berdasarkan prioritas kondisi: pertama mengecek apakah bilangan merupakan kelipatan 10 untuk melakukan operasi pembagian, kemudian mengecek kelipatan 5 (selain angka 5) untuk menghitung kuadratnya, dilanjutkan dengan pengecekan bilangan ganjil untuk dijumlahkan dengan angka urutan berikutnya, dan terakhir menggunakan blok default untuk menangani sisa bilangan genap dengan mengalikannya bersama angka setelahnya, di mana setiap kondisi akan menampilkan nama kategori serta hasil perhitungannya ke layar.