

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 11
SWITCH - CASE**



Disusun oleh:

RAFIF ARFA DHIAPRAJA

109082500041

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 jam24 int

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

    fmt.Scan(&jam24)

    var jam12 int
    var periode string

    switch {

    case jam24 == 0:

        jam12 = 12

        periode = "AM"

    case jam24 >= 1 && jam24 <= 11:

        jam12 = jam24

        periode = "AM"

    case jam24 == 12:

        jam12 = 12

        periode = "PM"

    case jam24 >= 13 && jam24 <= 23:
```

```

        jam12 = jam24 - 12

        periode = "PM"

    default:

        fmt.Println("Input tidak valid. Masukkan jam
antara 0-23.")

        return

    }

    fmt.Print(jam12, " ", periode)

}

```

Screenshoot program

The screenshot shows a Go IDE with three tabs: switch1.go, switch2.go, and switch3.go. The active tab is switch1.go, which contains the following code:

```

5 func main() {
6     fmt.Print("Masukkan jam 0-23: ")
7     fmt.Scan(&jam24)
8
9     var jam12 int
10    var periode string
11
12
13    switch {
14    case jam24 == 0:
15        jam12 = 12
16        periode = "AM"
17    case jam24 >= 1 && jam24 <= 11:
18        jam12 = jam24
19        periode = "AM"
20    case jam24 == 12:
21        jam12 = 12
22        periode = "PM"
23    case jam24 >= 13 && jam24 <= 23:
24        jam12 = jam24 - 12
25        periode = "PM"
26    default:
27        fmt.Println("Input tidak valid. Masukkan jam antara 0-23.")
28        return
29    }
30 }

```

Below the code editor is a terminal window showing the execution of the program. The user runs the command `go run switch1.go` and provides input for the time. The output shows the converted 12-hour time and the period (AM or PM).

```

PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch1.go
Masukkan jam 0-23: 13
1 PM
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch1.go
Masukkan jam 0-23: 0
12 AM
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch1.go
Masukkan jam 0-23: 12
12 PM
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE>

```

On the right side of the terminal, there is a small window titled "NAMA" showing the user's details:

```

NAMA : RAFIF ARFA DHIAPRAJA
KELAS : S1IF-13-07
NIM : 109082500041

```

Deskripsi program

Pada program ini, saya membuat sebuah program untuk mengonversi jam dari format 24 jam ke format 12 jam (AM/PM). Variabel jam24 digunakan untuk menyimpan jam dalam format 24 jam yang dimasukkan oleh pengguna, sedangkan variabel jam12 digunakan untuk menyimpan hasil konversi jam dalam format 12 jam, dan variabel periode digunakan untuk menyimpan keterangan waktu yaitu AM atau PM. Program dimulai dengan menampilkan pesan "Masukkan jam 0–23:" untuk meminta pengguna memasukkan jam dalam format 24 jam. Nilai tersebut kemudian dibaca dan disimpan ke dalam variabel jam24 menggunakan `fmt.Scan(&jam24)`. Setelah itu, program menggunakan struktur `switch case` untuk menentukan hasil konversi. Jika jam24 bernilai 0, maka jam12 diatur menjadi 12 dan periode diisi dengan "AM". Jika jam24 berada di antara 1 sampai 11, maka nilainya tetap dan periodenya adalah "AM". Jika jam24 bernilai 12, maka jam12 tetap 12 dan periodenya adalah "PM". Jika jam24 berada di antara 13 sampai 23, maka jam12 akan dikurangi 12 dan periodenya diisi dengan "PM". Jika input berada di luar rentang 0 sampai 23, program akan menampilkan pesan bahwa input tidak valid dan program dihentikan. Terakhir, program menampilkan hasil konversi jam dalam format 12 jam beserta periodenya menggunakan `fmt.Print(jam12, " ", periode)`.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    fmt.Print("Masukkan tanaman : ")

    var tanaman string

    fmt.Scan(&tanaman)

    switch tanaman {

    case "nepenthes" :

        fmt.Println("Termasuk tanaman karnivora")

        fmt.Println("Asli Indonesia")
```

```

        case "venus" :

            fmt.Println("Termasuk tanaman karnivora")

            fmt.Println("bukan asli indonesia")

        default :

            fmt.Println("Bukan tanaman karnivora")

    }

}

```

Screenshoot program

```

5 func main() {
6     fmt.Print("Masukkan tanaman : ")
7     var tanaman string
8     fmt.Scan(&tanaman)
9
10    switch tanaman {
11    case "nepenthes" :
12        fmt.Println("Termasuk tanaman karnivora")
13        fmt.Println("Asli Indonesia")
14    case "venus" :
15        fmt.Println("Termasuk tanaman karnivora")
16        fmt.Println("bukan asli indonesia")
17    default :
18        fmt.Println("Bukan tanaman karnivora")
19    }
20 }

```

```

PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch2.go
Masukkan tanaman : nepenthes
Termasuk tanaman karnivora
Asli Indonesia
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch2.go
Masukkan tanaman : venus
Termasuk tanaman karnivora
bukan asli indonesia
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch2.go
Masukkan tanaman : karedok
Bukan tanaman karnivora
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE>

```

NAMA : RAFIF ARFA DHIAPRAJA
 KELAS : S1IF-13-07
 NIM : 109082500041

Deskripsi program

Pada program ini, saya membuat sebuah program untuk menentukan apakah tanaman yang dimasukkan termasuk tanaman karnivora atau bukan, serta untuk mengetahui asal tanamannya. Variabel tanaman digunakan untuk menyimpan nama tanaman dalam bentuk string yang dimasukkan oleh pengguna. Program dimulai dengan menampilkan pesan "Masukkan tanaman : " untuk meminta pengguna memasukkan nama tanaman. Data yang dimasukkan kemudian dibaca dan disimpan ke dalam variabel tanaman menggunakan `fmt.Scan(&tanaman)`. Setelah itu, program menggunakan struktur `switch case` untuk melakukan pengecekan berdasarkan nama

tanaman yang dimasukkan. Jika pengguna memasukkan "nepenthes", maka program akan menampilkan bahwa tanaman tersebut termasuk tanaman karnivora dan berasal dari Indonesia. Jika pengguna memasukkan "venus", maka program akan menampilkan bahwa tanaman tersebut termasuk tanaman karnivora tetapi bukan berasal dari Indonesia. Jika input yang dimasukkan tidak sesuai dengan kedua nama tanaman tersebut, maka program akan menampilkan pesan bahwa tanaman tersebut bukan tanaman karnivora. Dengan demikian, program ini dapat mengelompokkan jenis tanaman berdasarkan input pengguna dan memberikan informasi apakah tanaman tersebut termasuk karnivora serta asalnya.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var jeniskendaraan string

    var durasiparkir int

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

    fmt.Scan(&jeniskendaraan)

    fmt.Print("Masukkan durasi parkir (dalam jam): ")

    fmt.Scan(&durasiparkir)

    switch {

    case jeniskendaraan == "mobil" && durasiparkir <=
2:

        fmt.Println("tarif parkir: Rp 15000")

    case jeniskendaraan == "mobil" && durasiparkir > 2:

        fmt.Println("tarif parkir: Rp 20000")
```

```
        case jeniskendaraan == "motor" && durasiparkir <=
2:

            fmt.Println("tarif parkir: Rp 7000")

        case jeniskendaraan == "motor" && durasiparkir > 2:

            fmt.Println("tarif parkir: Rp 9000")

        case jeniskendaraan == "truk" && durasiparkir <= 2:

            fmt.Println("tarif parkir: Rp 25000")

        case jeniskendaraan == "truk" && durasiparkir > 2:

            fmt.Println("tarif parkir: Rp 35000")

        default:

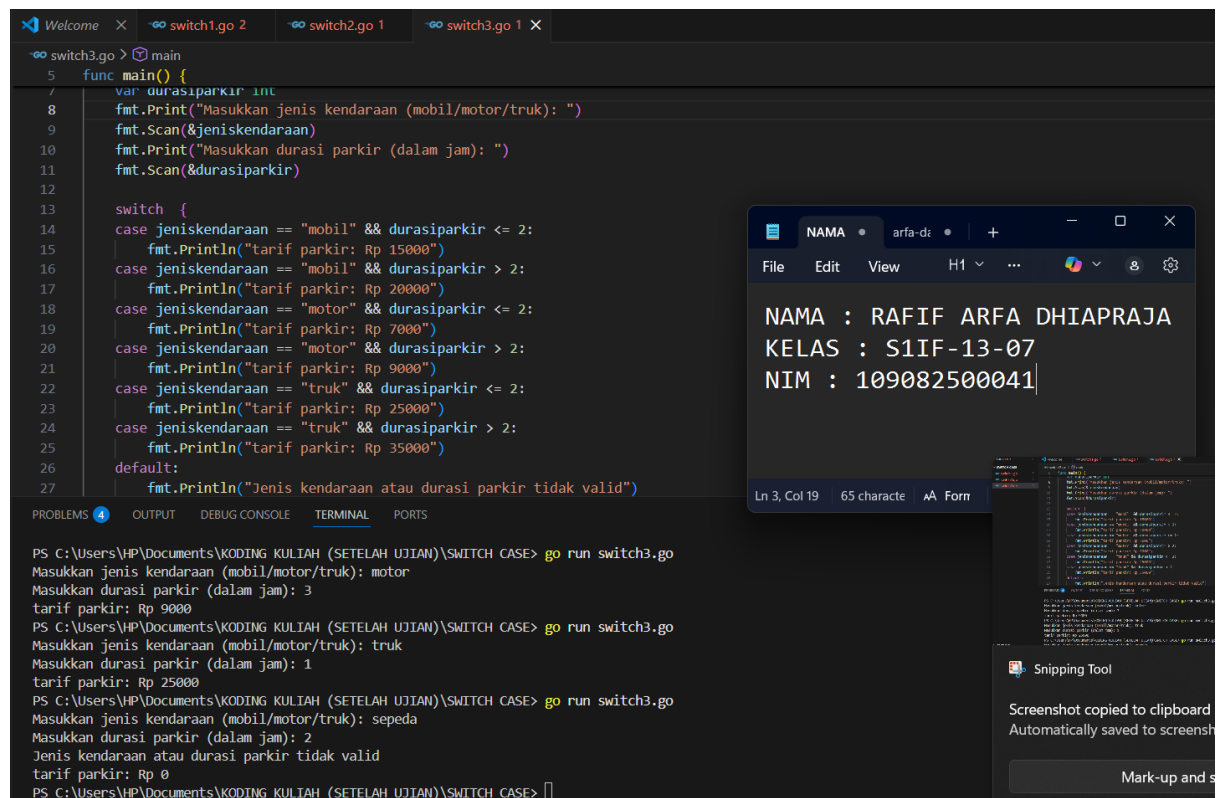
            fmt.Println("Jenis kendaraan atau durasi parkir
tidak valid")

            fmt.Println("tarif parkir: Rp 0")

    }

}
```

Screenshoot program



The screenshot shows a Go program running in a terminal window. The program is a switch statement that calculates parking fees based on vehicle type and duration. The terminal output shows the program running successfully for three different inputs: a car parked for 3 hours (Rp 20,000), a truck parked for 1 hour (Rp 25,000), and a motorcycle parked for 2 hours (Rp 9,000). A fourth input, 'sepeda' parked for 2 hours, results in a default message 'Jenis kendaraan atau durasi parkir tidak valid' and a fee of Rp 0.

```
switch3.go > main
5 func main() {
7     var durasiparkir int
8     fmt.Println("Masukkan jenis kendaraan (mobil/motor/truk): ")
9     fmt.Scan(&jeniskendaraan)
10    fmt.Println("Masukkan durasi parkir (dalam jam): ")
11    fmt.Scan(&durasiparkir)
12
13    switch {
14    case jeniskendaraan == "mobil" && durasiparkir <= 2:
15        fmt.Println("tarif parkir: Rp 15000")
16    case jeniskendaraan == "mobil" && durasiparkir > 2:
17        fmt.Println("tarif parkir: Rp 20000")
18    case jeniskendaraan == "motor" && durasiparkir <= 2:
19        fmt.Println("tarif parkir: Rp 7000")
20    case jeniskendaraan == "motor" && durasiparkir > 2:
21        fmt.Println("tarif parkir: Rp 9000")
22    case jeniskendaraan == "truk" && durasiparkir <= 2:
23        fmt.Println("tarif parkir: Rp 25000")
24    case jeniskendaraan == "truk" && durasiparkir > 2:
25        fmt.Println("tarif parkir: Rp 35000")
26    default:
27        fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")
28    }
29 }
```

Terminal Output:

```
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch3.go
Masukkan jenis kendaraan (mobil/motor/truk): motor
Masukkan durasi parkir (dalam jam): 3
tarif parkir: Rp 20000
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch3.go
Masukkan jenis kendaraan (mobil/motor/truk): truk
Masukkan durasi parkir (dalam jam): 1
tarif parkir: Rp 25000
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE> go run switch3.go
Masukkan jenis kendaraan (mobil/motor/truk): sepeda
Masukkan durasi parkir (dalam jam): 2
Jenis kendaraan atau durasi parkir tidak valid
tarif parkir: Rp 0
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\SWITCH CASE>
```

The text editor shows the user input for the first run:

```
NAMA : RAFIF ARFA DHIAPRAJA
KELAS : S1IF-13-07
NIM : 109082500041
```

Deskripsi program

Pada program ini, saya membuat sebuah program untuk menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parkir yang dimasukkan oleh pengguna. Program menggunakan dua variabel, yaitu jeniskendaraan bertipe string untuk menyimpan jenis kendaraan (mobil, motor, atau truk), dan durasiparkir bertipe integer untuk menyimpan lama parkir dalam satuan jam. Program diawali dengan menampilkan pesan kepada pengguna untuk memasukkan jenis kendaraan, kemudian input tersebut dibaca dan disimpan ke dalam variabel jeniskendaraan menggunakan `fmt.Scan`. Setelah itu, program meminta pengguna memasukkan durasi parkir dalam jam, lalu menyimpannya ke dalam variabel durasiparkir. Selanjutnya, program menggunakan struktur switch tanpa kondisi langsung, sehingga setiap case berisi kondisi logika yang menggabungkan jenis kendaraan dan lama parkir. Jika kendaraan adalah mobil dan durasi parkir kurang dari atau sama dengan 2 jam, maka tarif yang ditampilkan adalah Rp 15.000. Jika lebih dari 2 jam, tarifnya menjadi Rp 20.000. Untuk motor, tarifnya adalah Rp 7.000 jika parkir sampai 2 jam dan Rp 9.000 jika lebih dari 2 jam. Sedangkan untuk truk, tarifnya adalah Rp 25.000 jika parkir sampai 2 jam dan Rp 35.000 jika lebih dari 2 jam. Jika input jenis kendaraan atau durasi parkir tidak sesuai dengan kondisi yang tersedia, maka program akan mengeksekusi bagian default dengan menampilkan pesan bahwa data tidak valid dan tarif parkir bernilai Rp 0. Dengan demikian, program ini dapat menentukan tarif parkir secara otomatis berdasarkan jenis kendaraan dan lama waktu parkir yang dimasukkan oleh pengguna.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var ph float64

    fmt.Print("Masukkan kadar pH: ")

    fmt.Scan(&ph)

    switch {

    case ph < 0 || ph > 14:

        fmt.Println("Input tidak valid, rentang pH 0 - 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 Go program in VS Code and its execution in a terminal. The program is a switch-case exercise for checking if water is suitable for drinking based on pH. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var ph float64
7     fmt.Print("Masukkan kadar pH: ")
8     fmt.Scan(&ph)
9
10    switch {
11    case ph < 0 || ph > 14:
12        fmt.Println("Input tidak valid, rentang pH 0 - 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 }
```

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

```
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol1.go
Masukkan kadar pH: 8.6
Air layak Minum
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol1.go
Masukkan kadar pH: 9
Air tidak layak Minum
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol1.go
Masukkan kadar pH: 16
Input tidak valid, rentang pH 0 - 14
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> 
```

Deskripsi program

Pada program ini, saya membuat sebuah program untuk menentukan apakah air layak diminum atau tidak berdasarkan kadar pH yang dimasukkan oleh pengguna. Program menggunakan satu variabel bertipe float64, yaitu ph, untuk menyimpan nilai pH. Program diawali dengan menampilkan pesan "Masukkan kadar pH:" untuk meminta pengguna memasukkan nilai pH air. Nilai tersebut kemudian dibaca dan disimpan ke dalam variabel ph menggunakan fmt.Scan. Setelah nilai pH diperoleh, program menggunakan switch dengan kondisi logika untuk melakukan pengecekan. Jika nilai pH yang dimasukkan kurang dari 0 atau lebih dari 14, maka program akan menampilkan pesan bahwa input tidak valid karena nilai pH seharusnya berada pada rentang 0 sampai 14. Jika nilai pH berada pada rentang 6.5 sampai 8.6, maka air dinyatakan layak minum. Selain dari rentang tersebut, program akan menampilkan bahwa air tidak layak minum. Dengan demikian, program ini dapat membantu menentukan kelayakan air untuk diminum berdasarkan nilai pH yang dimasukkan oleh pengguna.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var kendaraan string

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

    fmt.Scan(&kendaraan)

    var durasi int

    fmt.Print("Masukkan durasi parkir (dalam jam): ")

    fmt.Scan(&durasi)

    total := 0

    switch kendaraan {

    case "mobil":

        total = durasi * 5000

        fmt.Println("Rp ", total)

    case "motor":

        total = durasi * 2000

        fmt.Println("Rp ", total)

    case "truk":

        total = durasi * 8000
```

```

        fmt.Println("Rp", total)

    default:

        fmt.Println("Jenis kendaraan tidak dikenali")

    }

}

```

Screenshoot program

The screenshot displays the Go source code for a parking fee calculator and its execution results. The code defines a `main` function that prompts the user for vehicle type and parking duration, then calculates the total fee using a `switch` statement. The terminal output shows three successful calculations and one default case. A separate window shows the user's personal information.

```

func main() {
    var kendaraan string
    fmt.Print("Masukkan jenis kendaraan (mobil/motor/truk): ")
    fmt.Scan(&kendaraan)
    var durasi int
    fmt.Print("Masukkan durasi parkir (dalam jam): ")
    fmt.Scan(&durasi)

    total := 0

    switch kendaraan {
    case "mobil":
        total = durasi * 5000
        fmt.Println("Rp ", total)
    case "motor":
        total = durasi * 2000
        fmt.Println("Rp ", total)
    case "truk":
        total = durasi * 8000
        fmt.Println("Rp", total)
    default:
        fmt.Println("Jenis kendaraan tidak dikenali")
    }
}

```

Terminal Output:

```

PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol2.go
Masukkan jenis kendaraan (mobil/motor/truk): motor
Masukkan durasi parkir (dalam jam): 3
Rp 6000
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol2.go
Masukkan jenis kendaraan (mobil/motor/truk): mobil
Masukkan durasi parkir (dalam jam): 1
Rp 5000
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol2.go
Masukkan jenis kendaraan (mobil/motor/truk): truk
Masukkan durasi parkir (dalam jam): 5
Rp 40000

```

User Information Window:

```

NAMA : RAFIF ARFA DHIAPRAJA
KELAS : S1IF-13-07
NIM : 109082500041

```

Deskripsi program

Pada program ini, saya membuat sebuah program untuk menghitung total biaya parkir berdasarkan jenis kendaraan dan durasi parkir yang dimasukkan oleh pengguna. Program menggunakan dua variabel utama, yaitu kendaraan bertipe string untuk menyimpan jenis kendaraan, dan durasi bertipe integer untuk menyimpan lama parkir dalam satuan jam. Program diawali dengan menampilkan pesan "Masukkan jenis

kendaraan (mobil/motor/truk):” agar pengguna memasukkan jenis kendaraannya. Setelah itu, program meminta pengguna memasukkan durasi parkir dalam jam melalui pesan “Masukkan durasi parkir (dalam jam):”. Kedua input tersebut dibaca menggunakan `fmt.Scan`. Selanjutnya, program membuat variabel `total` yang diinisialisasi dengan nilai 0 untuk menyimpan total biaya parkir. Program kemudian menggunakan struktur `switch` untuk menentukan tarif berdasarkan jenis kendaraan. Jika kendaraan adalah mobil, maka biaya parkir dihitung dengan rumus durasi dikali 5000. Jika kendaraan adalah motor, maka biaya dihitung dengan durasi dikali 2000. Jika kendaraan adalah truk, maka biaya dihitung dengan durasi dikali 8000. Setelah itu, hasil perhitungan langsung ditampilkan dalam bentuk rupiah. Jika pengguna memasukkan jenis kendaraan selain mobil, motor, atau truk, maka program akan menampilkan pesan bahwa jenis kendaraan tidak dikenali. Dengan demikian, program ini dapat digunakan untuk menghitung total biaya parkir secara otomatis berdasarkan jenis kendaraan dan lama parkir yang dimasukkan oleh pengguna.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Print("Masukkan sebuah bilangan: ")
    fmt.Scan(&n)

    switch {
        case n%10 == 0:
            fmt.Println("Kategantaraori: Bilangan  
Kelipatan 10")
            fmt.Printf("Hasil pembagian  %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 adalah
%d\n", n, n*n)

    case n%2 == 1:
        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))

    }
}

```

Screenshoot program

The screenshot shows a Go program in VS Code. The editor displays the source code for `latsol3.go`, which uses a `switch` statement to perform calculations based on input numbers. The terminal window shows the execution output for three different inputs: 8, 25, and 20. A separate window displays the user's personal information.

Source Code (latsol3.go):

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     fmt.Print("Masukkan sebuah bilangan: ")
8     fmt.Scan(&n)
9
10    switch {
11    case n%10 == 0:
12        fmt.Println("Kategorisasi: Bilangan Kelipatan 10")
13        fmt.Printf("Hasil pembagian %d / 10= %d\n", n, n/10)
14    case n%5 == 0 && n != 5:
15        fmt.Println("Kategori: Bilangan Kelipatan 5")
16        fmt.Printf("hasil kuadrat dari %d adalah %d\n", n, n*n)
17    case n%2 == 1:
18        fmt.Println("Kategori: bilangan Ganjil")
19        fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", n, n+1, n+(n+1))
20    case n%2 == 0:
21        fmt.Println("Kategori: Bilangan Genap")
22        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n", n, n+1, n*(n+1))
23    }
24 }

```

Terminal Output:

```

Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol3.go
Masukkan sebuah bilangan: 8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol3.go
Masukkan sebuah bilangan: 25
Kategori: Bilangan Kelipatan 5
hasil kuadrat dari 25 adalah 625
PS C:\Users\HP\Documents\KODING KULIAH (SETELAH UJIAN)\Switch Case Latihan> go run latsol3.go
Masukkan sebuah bilangan: 20
Kategorisasi: Bilangan Kelipatan 10
Hasil pembagian 20 / 10= 2

```

Personal Information Window:

```

NAMA : RAFIF ARFA DHIAPRAJA
KELAS : S1IF-13-07
NIM : 109082500041

```

Deskripsi program

Pada program ini, saya membuat sebuah program untuk mengelompokkan sebuah bilangan berdasarkan kategorinya dan menampilkan hasil operasi tertentu sesuai dengan kategori bilangan tersebut. Program menggunakan satu variabel bertipe integer, yaitu n , yang digunakan untuk menyimpan bilangan yang dimasukkan oleh pengguna. Program dimulai dengan menampilkan pesan "Masukkan sebuah bilangan:" untuk meminta pengguna memasukkan sebuah bilangan. Nilai yang dimasukkan kemudian dibaca menggunakan `fmt.Scan(&n)`. Setelah itu, program menggunakan struktur `switch` dengan kondisi untuk mengecek jenis bilangan berdasarkan beberapa aturan. Jika bilangan habis dibagi 10 ($n \% 10 == 0$), maka bilangan tersebut dikategorikan sebagai kelipatan 10. Pada kondisi ini, program menampilkan hasil pembagian bilangan tersebut dengan 10. Jika bilangan habis dibagi 5 dan bukan bernilai 5, maka bilangan dikategorikan sebagai kelipatan 5, dan program menampilkan hasil kuadrat dari bilangan tersebut. Jika bilangan ganjil ($n \% 2 == 1$), maka program menampilkan bahwa bilangan tersebut adalah bilangan ganjil serta menampilkan hasil penjumlahan bilangan dengan bilangan berikutnya. Jika bilangan genap ($n \% 2 == 0$), maka program menampilkan bahwa bilangan tersebut adalah bilangan genap dan menampilkan hasil perkalian bilangan dengan bilangan berikutnya. Melalui program ini, setiap bilangan yang dimasukkan akan dikenali kategorinya, kemudian program akan menampilkan hasil operasi yang sesuai dengan kategori bilangan tersebut.