

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 11
SWITCH-CASE



Disusun oleh:
MUHAMMAD FAIZ MAULANA
109082500124
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 jam int
    var jamFormat int
    var perihal string

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

    switch jam {
    case 0:
        jamFormat = 12
        perihal = "AM"
    case 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11:
        jamFormat = jam
        perihal = "AM"
    case 12:
        jamFormat = 12
        perihal = "PM"
    case 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23:
        jamFormat = jam - 12
        perihal = "PM"
    default:
        fmt.Println("Jam tidak valid! Masukkan jam 0-23")
    }
}
```

```
}
```

```
    fmt.Printf("%d:00 = %d %s\n", jam, jamFormat,  
perihal)
```

```
}
```

Screenshot program

```
jam.go > ...  
1 package main  
2  
3 import "fmt"  
4  
5 func main() {  
6     var jam int  
7     var jamFormat int  
8     var perihal string  
9  
10    fmt.Print("Masukkan jam (0-23): ")  
11    fmt.Scan(&jam)  
12  
13    switch jam {  
14        case 0:  
15            jamFormat = 12  
16            perihal = "AM"  
17        case 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11:  
18            jamFormat = jam  
19            perihal = "AM"  
20        case 12:  
21            jamFormat = 12  
22            perihal = "PM"  
23        case 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23:  
24            jamFormat = jam - 12  
25            perihal = "PM"  
26    }  
27  
28    fmt.Printf("%d:00 = %d %s\n", jam, jamFormat,  
29    perihal)  
30}
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu_11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu_11\jam.go"  
Masukkan jam (0-23): 13  
13:00 - 1 PM  
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu_11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu_11\jam.go"  
Masukkan jam (0-23): 0  
0:00 - 12 AM  
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu_11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu_11\jam.go"  
Masukkan jam (0-23): 12  
12:00 - 12 PM
```

Deskripsi program

Program Go ini berfungsi untuk mengkonversi jam dalam format 24 jam (0–23) menjadi format 12 jam dengan keterangan AM atau PM. Pengguna diminta memasukkan sebuah angka jam melalui input keyboard, kemudian program menyimpan nilai tersebut dalam variabel jam. Melalui pernyataan switch, program menentukan nilai jamFormat dan perihal (AM/PM): jika jam 0 akan ditampilkan sebagai 12 AM, jam 1–11 tetap sama dengan label AM, jam 12 menjadi 12 PM, dan jam 13–23 dikonversi menjadi jamFormat = jam - 12 dengan label PM. Jika input di luar rentang 0–23, program menampilkan pesan bahwa jam tidak valid dan langsung berhenti. Terakhir, program menampilkan hasil konversi dalam format <jam_awal>:00 = <jamFormat> <AM/PM>.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var Tanaman string
    var keluaran string

    fmt.Println("Masukkan nama tanaman: ")
    fmt.Scan(&Tanaman)
    switch Tanaman {
        case "nepenthes":
            keluaran = "Termasuk Tanaman Karnivora\nAsli Indonesia"
        case "venus":
            keluaran = "Termasuk Tanaman Karnivora\nBukan Asli Indonesia"
        case "karedok":
            keluaran = "Tidak termasuk Tanaman Karnivora"
        default:
            keluaran = "tidak diketahui"
    }

    fmt.Println(keluaran)
}
```

Screenshot program

The screenshot shows a code editor with a Go file named `karnivora.go`. The code defines a `main` function that prompts the user for a plant name, checks it against a switch statement, and prints the result. Below the editor is a terminal window showing the execution of the program and its output for three different inputs: "nepenthes", "venus", and "karedok".

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var Tanaman string
7     var keluaran string
8
9     fmt.Println("Masukkan nama tanaman: ")
10    fmt.Scan(&Tanaman)
11    switch Tanaman {
12        case "nepenthes":
13            keluaran = "Termasuk Tanaman Karnivora\nAsli Indonesia"
14        case "venus":
15            keluaran = "Termasuk Tanaman Karnivora\nBukan Asli Indonesia"
16        case "karedok":
17            keluaran = "Tidak termasuk Tanaman Karnivora"
18        default:
19            keluaran = "tidak diketahui"
20    }
21
22    fmt.Println(keluaran)
23 }
```

TERMINAL

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

```
Masukkan nama tanaman: nepenthes
Termasuk Tanaman Karnivora
Asli Indonesia
PS C:\Users\user-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11\karnivora.go"
Masukkan nama tanaman: venus
Termasuk Tanaman Karnivora
Bukan Asli Indonesia
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11\karnivora.go"
Masukkan nama tanaman: karedok
Tidak termasuk Tanaman Karnivora
```

Deskripsi program

Program Go ini meminta pengguna memasukkan nama tanaman melalui keyboard dan menyimpannya ke dalam variabel `Tanaman`. Setelah itu, program menggunakan struktur kontrol `switch` untuk menentukan informasi tentang tanaman tersebut: jika input adalah "nepenthes", program menyatakan bahwa tanaman itu termasuk tanaman karnivora dan asli Indonesia; jika "venus", termasuk tanaman karnivora tetapi bukan asli Indonesia; jika "karedok", dinyatakan bukan tanaman karnivora; sedangkan untuk nama lain yang tidak dikenali, program menampilkan bahwa tanaman tersebut "tidak diketahui" melalui variabel `keluaran`.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var jenisKendaraan string
    var durasi int
    var tarif int
```

```
fmt.Print("Masukkan jenis kendaraan  
(Motor/Mobil/Truk) : ")  
  
fmt.Scan(&jenisKendaraan)  
  
fmt.Print("Masukkan durasi parkir (dalam jam) : ")  
fmt.Scan(&durasi)  
  
if durasi < 1 {  
    fmt.Println("Jenis kendaraan atau durasi parkir  
tidak valid")  
    fmt.Printf("Tarif Parkir: Rp 0\n")  
    return  
}  
  
switch {  
    case jenisKendaraan == "Motor" && durasi >= 1 &&  
    durasi <= 2:  
        tarif = 7000  
    case jenisKendaraan == "Motor" && durasi > 2:  
        tarif = 9000  
    case jenisKendaraan == "Mobil" && durasi >= 1 &&  
    durasi <= 2:  
        tarif = 15000  
    case jenisKendaraan == "Mobil" && durasi > 2:  
        tarif = 20000  
    case jenisKendaraan == "Truk" && durasi >= 1 &&  
    durasi <= 2:  
        tarif = 25000  
    case jenisKendaraan == "Truk" && durasi > 2:  
        tarif = 35000  
    default:  
        fmt.Println("Jenis kendaraan atau durasi parkir  
tidak valid")
```

```
    fmt.Printf("Tarif Parkir: Rp 0\n")  
    return  
}  
  
fmt.Printf("Tarif Parkir: Rp %d\n", tarif)  
}
```

Screenshot program

```
func main() {
    case jenisKendaraan == "MOTOR" && durasi >= 1 && durasi <= 1:
        tarif = 7000
    case jenisKendaraan == "Motor" && durasi > 2:
        tarif = 9000
    case jenisKendaraan == "Mobil" && durasi >= 1 && durasi <= 2:
        tarif = 15000
    case jenisKendaraan == "Mobil" && durasi > 2:
        tarif = 20000
    case jenisKendaraan == "Truk" && durasi >= 1 && durasi <= 2:
        tarif = 25000
    case jenisKendaraan == "Truk" && durasi > 2:
        tarif = 35000
    default:
        fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")
        fmt.Printf("Tarif Parkir: Rp 0\n")
        return
    }
    fmt.Printf("Tarif Parkir: Rp %d\n", tarif)
}
```

Deskripsi program

Program Go ini menghitung tarif parkir berdasarkan jenis kendaraan dan lama parkir. Pengguna diminta memasukkan jenis kendaraan (Motor/Mobil/Truk) dan durasi parkir dalam jam. Jika durasi kurang dari 1 jam, program langsung menyatakan data tidak valid dan tarif Rp 0. Dengan switch, program menentukan tarif: Motor Rp 7000 (1–2 jam) atau Rp 9000 (>2 jam), Mobil Rp 15000 (1–2 jam) atau Rp 20000 (>2 jam), dan Truk Rp 25000 (1–2 jam) atau Rp 35000 (>2 jam). Jika jenis kendaraan tidak sesuai, program juga menampilkan data tidak valid dan tarif Rp 0.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var pH float64
    var kategori int

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

    if pH < 0 || pH > 14 {
        fmt.Println("Input tidak valid, rentang pH 0 - 14")
        return
    }

    switch {
    case pH < 6.5:
        kategori = 0
    case pH >= 6.5 && pH <= 8.6:
        kategori = 1
    case pH > 8.6:
        kategori = 2
    }
}
```

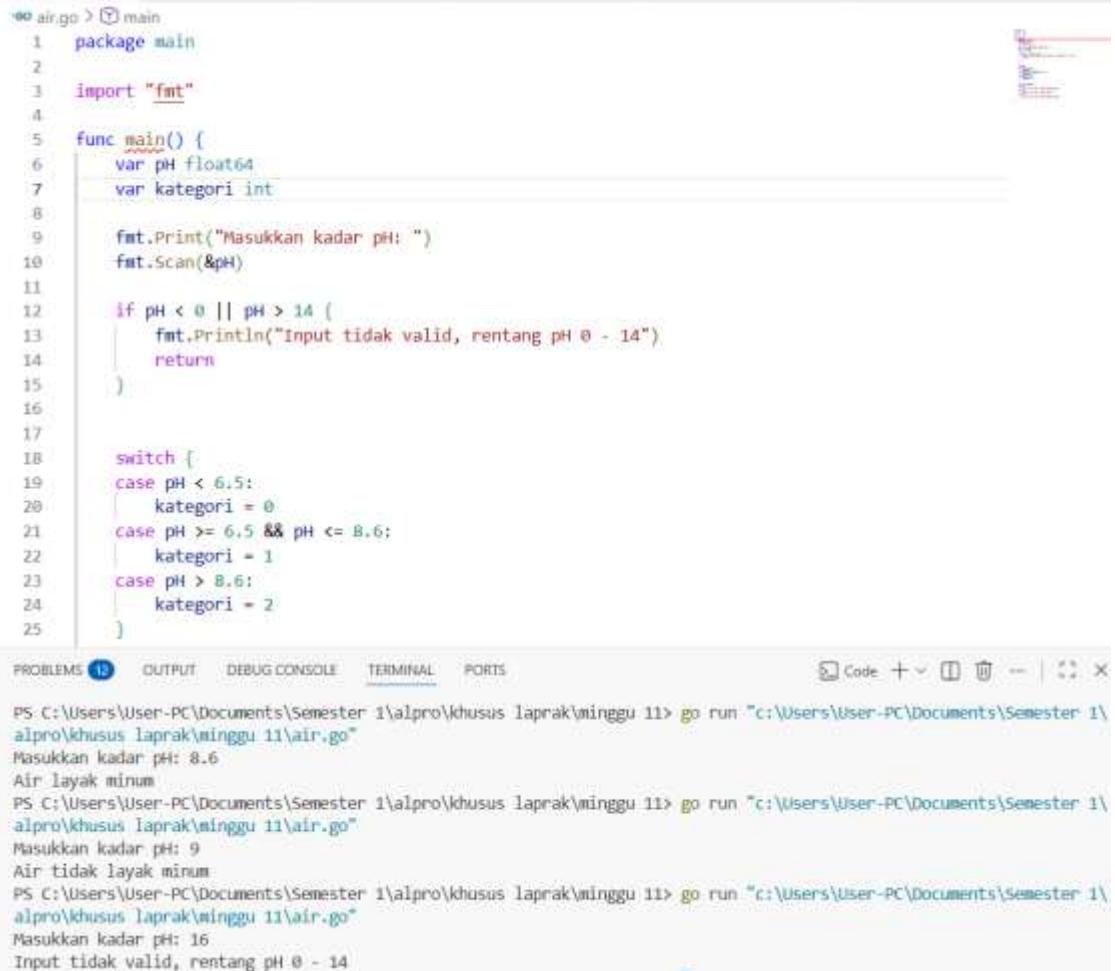
```

}

switch kategori {
    case 0:
        fmt.Println("Air tidak layak minum")
    case 1:
        fmt.Println("Air layak minum")
    case 2:
        fmt.Println("Air tidak layak minum")
}
}

```

Screenshoot program



```

air.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var pH float64
7     var kategori int
8
9     fmt.Print("Masukkan kadar pH: ")
10    fmt.Scan(&pH)
11
12    if pH < 0 || pH > 14 {
13        fmt.Println("Input tidak valid, rentang pH 0 - 14")
14        return
15    }
16
17
18    switch {
19    case pH < 6.5:
20        kategori = 0
21    case pH >= 6.5 && pH <= 8.6:
22        kategori = 1
23    case pH > 8.6:
24        kategori = 2
25    }

```

PROBLEMS 12 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11\air.go"
Masukkan kadar pH: 8.6
Air layak minum
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11\air.go"
Masukkan kadar pH: 9
Air tidak layak minum
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus_laprak\minggu 11\air.go"
Masukkan kadar pH: 16
Input tidak valid, rentang pH 0 - 14

```

Deskripsi program

Program Go ini digunakan untuk menentukan kelayakan air minum berdasarkan nilai pH. Pengguna diminta memasukkan kadar pH, lalu program mengecek apakah nilainya berada di rentang 0–14; jika di luar itu, ditampilkan pesan “Input tidak valid” dan program berhenti. Jika valid, program mengelompokkan pH ke dalam variabel kategori: pH < 6.5 (kategori 0), 6.5–8.6 (kategori 1), dan > 8.6 (kategori 2). Berdasarkan kategori tersebut, program mencetak apakah air “layak minum” (hanya kategori 1) atau “tidak layak minum” (kategori 0 dan 2).

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var jenisKendaraan string
    var durasi int
    var tarifPerJam, totalBiaya int

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

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

    if durasi < 1 {
        durasi = 1
    }

    switch jenisKendaraan {
    case "motor":
        tarifPerJam = 2000
    case "mobil":
        tarifPerJam = 5000
    case "truk":
```

```

        tarifPerJam = 8000

    default:

        fmt.Println("Jenis kendaraan tidak valid")

        return

    }

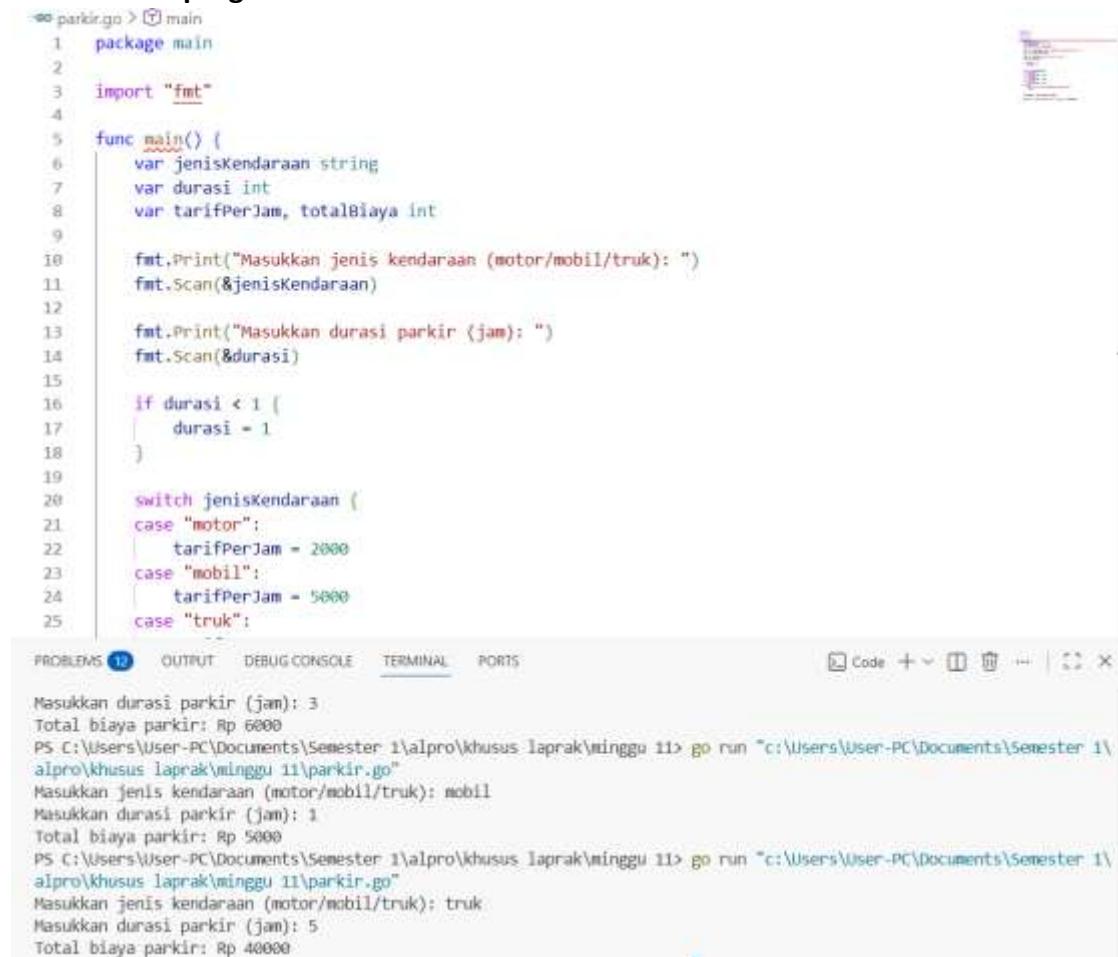
totalBiaya = tarifPerJam * durasi

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

}

```

Screenshoot 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 'parkir.go'. The user inputs the duration of parking (3 hours) and the type of vehicle (mobil). The program calculates and prints the total parking fee (Rp 6000).

```

$ go run c:/Users/User-PC/Documents/Semester 1\alpro\khusus_laprak\minggu 11\parkir.go
Masukkan durasi parkir (jam): 3
Total biaya parkir: Rp 6000

```

Deskripsi program

Program Go ini menghitung total biaya parkir berdasarkan jenis kendaraan dan durasi parkir. Pengguna diminta memasukkan jenis kendaraan (motor/mobil/truk) dan lama parkir dalam jam. Jika durasi yang diinput kurang dari 1 jam, program otomatis mengubahnya menjadi 1 jam. Dengan menggunakan switch, program menentukan tarif per jam: motor Rp 2000, mobil Rp 5000, dan truk Rp 8000. Jika jenis kendaraan tidak sesuai dengan ketiga pilihan tersebut, program menampilkan pesan "Jenis kendaraan tidak valid" dan berhenti. Total biaya dihitung dengan mengalikan tarif per jam dengan durasi parkir, kemudian hasilnya ditampilkan dalam format rupiah.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var bilangan int
    var kategori int

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

    switch {
        case bilangan%10 == 0:
            kategori = 1
        case bilangan%5 == 0:
            kategori = 2
        case bilangan%2 == 1 || bilangan%2 == -1:
            kategori = 3
        case bilangan%2 == 0:
            kategori = 4
    }

    switch kategori {
        case 1:
            hasil := bilangan / 10
    }
}
```

```

        fmt.Printf("Kategori: Bilangan Kelipatan 10\n")

        fmt.Printf("Hasil pembagian antara %d / 10 = %d\n",
bilangan, hasil)

    case 2:

        hasil := bilangan * bilangan

        fmt.Printf("Kategori: Bilangan Kelipatan 5\n")

        fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", bilangan,
hasil)

    case 3:

        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)

    case 4:

        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)

    }

}

```

Screenshot program

```

ssal3.go > main
5 func main() {
10     bilangan := 0
11
12     switch {
13         case bilangan%10 == 0:
14             kategori = 1
15         case bilangan%5 == 0:
16             kategori = 2
17         case bilangan%2 == 0 || bilangan%2 == -1:
18             kategori = 3
19         case bilangan%2 == 1:
20             kategori = 4
21     }
22
23     switch kategori {
24         case 1:
25             hasil := bilangan / 10
26             fmt.Printf("Kategori: Bilangan Kelipatan 10\n")
27             fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", bilangan, hasil)
28         case 2:
29             hasil := bilangan * bilangan
30             fmt.Printf("Kategori: Bilangan Kelipatan 5\n")
31             fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", bilangan, hasil)
32         case 3:
33             hasil := bilangan + (bilangan + 1)
}

```

The screenshot shows the Go code editor interface with the file `ssal3.go` open. The code defines a `main` function that takes a number `bilangan` as input and prints its category based on its remainder when divided by 10, 5, or 2. It then performs different operations based on the category: division by 10, multiplication by itself, or addition of itself and one. The terminal below shows the execution of the program with inputs 72, 25, and 20, and outputs the corresponding categories and results.

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Kategori: Bilangan Genap
Hasil pembagian dengan bilangan berikutnya 8 * 9 = 72
PS C:\Users\User-PC\Documents\Semester 3\alpro\khusus_laprak\minggu 11> go run "c:\users\user\pc\documents\semester 3\alpro\khusus_laprak\minggu 11\ssal3.go"
Masukkan bilangan bulat: 25
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 25 ^2 = 625
PS C:\Users\User-PC\Documents\Semester 3\alpro\khusus_laprak\minggu 11> go run "c:\users\user\pc\documents\semester 3\alpro\khusus_laprak\minggu 11\ssal3.go"
Masukkan bilangan bulat: 20
Kategori: Bilangan Kelipatan 10
Hasil pembagian antara 20 / 10 = 2

```

Deskripsi program

Program Go ini mengklasifikasikan sebuah bilangan bulat ke beberapa kategori lalu melakukan operasi berbeda sesuai kategorinya. Setelah pengguna memasukkan bilangan, switch pertama menentukan kategori: jika kelipatan 10 → kategori 1, jika kelipatan 5 → kategori 2, jika ganjil (termasuk negatif ganjil) → kategori 3, dan jika genap → kategori 4. switch kedua menjalankan operasi: kategori 1 membagi bilangan dengan 10, kategori 2 menghitung kuadrat, kategori 3 menjumlahkan bilangan dengan bilangan berikutnya, kategori 4 mengalikan bilangan dengan bilangan berikutnya, lalu menampilkan hasilnya.