

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 11
Switch-Case



Disusun oleh:

DADI MAULANA MUHAMMAD

109082500080

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1

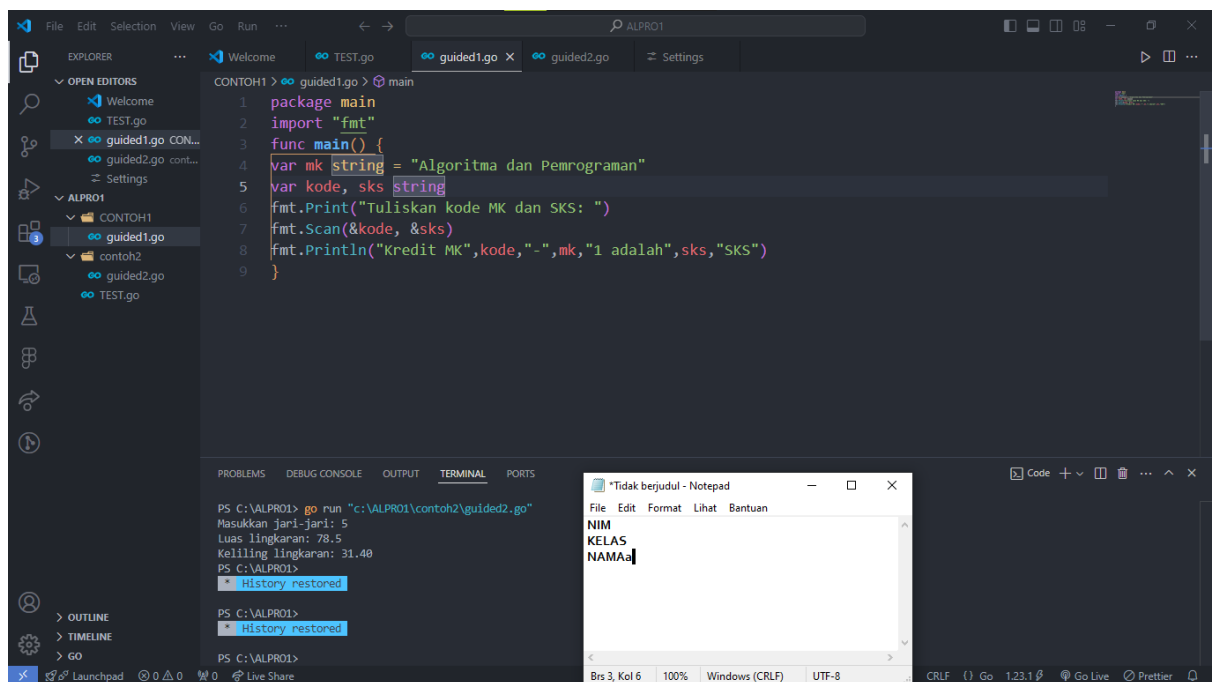
Source Code

```
//SALIN KODE KESINI  
  
ATURAN:  
  
PENULISAN SEESUAI MODUL 1  
  
GUNAKAN FONT Courier New ukuran 11pt dengan spasi baris  
dan paragraf 1,5
```

Screenshoot program

//tambahkan tangkapan layar dari program (boleh lebih dari 1 jika diperlukan)

CONTOH TANGKAPAN LAYAR:



Deskripsi program

Jelaskan kode yang ada di source code, semakin detil semakin baik nilainya

2. Guided 2

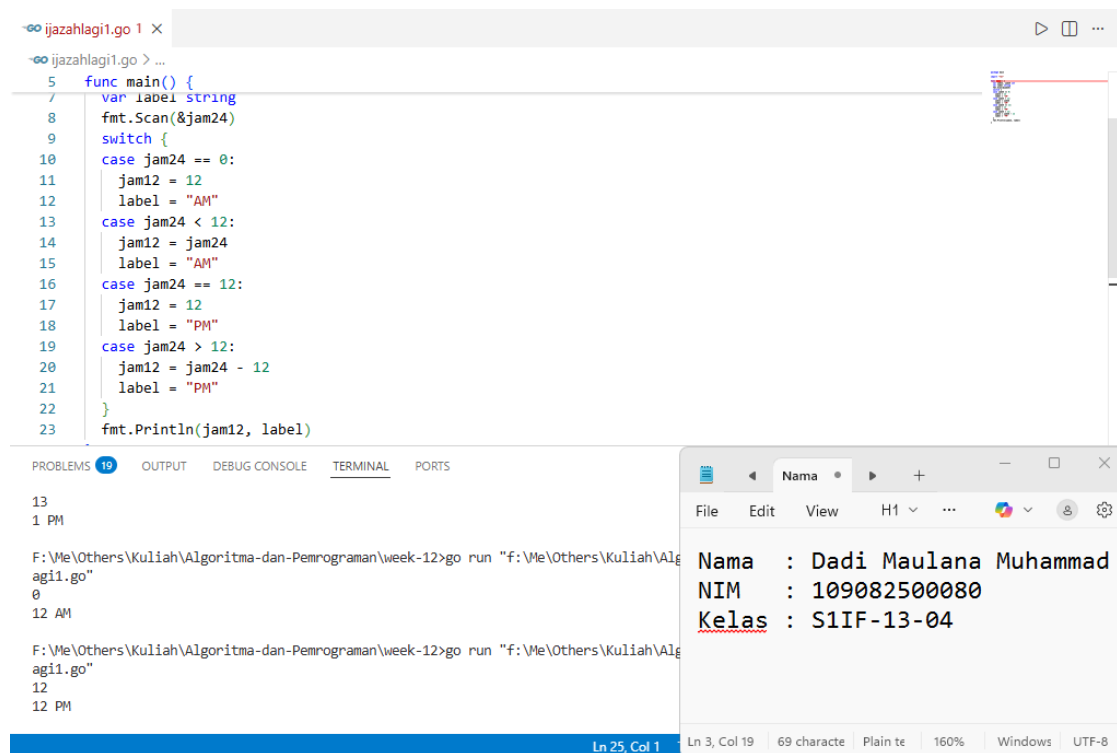
Source Code

```
p 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)
}
```

Screenshoot program



The screenshot shows a Go program in a text editor and its execution output in the terminal. The program is a function `main()` that reads a 24-hour time value from the command line, converts it to a 12-hour format, and prints the result with an AM or PM label.

```
5 func main() {  
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        label = "PM"  
18     case jam24 > 12:  
19        jam12 = jam24 - 12  
20        label = "PM"  
21     }  
22     fmt.Println(jam12, label)  
23 }
```

The terminal output shows the program being run with the input `0`, resulting in `12 AM`, and then with the input `12`, resulting in `12 PM`.

```
13  
14 1 PM  
  
F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg  
agi1.go"  
0  
12 AM  
  
F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg  
agi1.go"  
12  
12 PM
```

Deskripsi program

Program ini mengonversi format waktu dari sistem 24 jam ke sistem 12 jam. Program terlebih dahulu membaca sebuah angka yang mewakili jam dalam format 24 jam dan menyimpannya pada variabel `jam24`. Selanjutnya, program menggunakan struktur *switch* untuk menentukan jam yang sesuai dalam format 12 jam serta menentukan label waktu, yaitu "AM" atau "PM".

Jika nilai `jam24` bernilai 0, maka program mengubahnya menjadi pukul 12 AM. Jika `jam24` berada di bawah 12 tetapi tidak 0, maka jam tersebut tetap sama dan diberi label AM. Bila `jam24` sama dengan 12, program mengubahnya menjadi pukul 12 PM. Sedangkan untuk nilai `jam24` yang lebih besar dari 12, program mengurangi nilai jam dengan 12 dan memberikan label PM.

Terakhir, hasil konversi berupa jam dalam format 12 jam beserta label AM atau PM ditampilkan ke layar. Dengan demikian, program membantu pengguna membaca format waktu dengan lebih mudah dalam sistem 12 jam.

3. Guided 3

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.")
    }
}
```

Screenshoot program

The screenshot shows a Go program in a text editor and its execution output in a terminal. The program is named `ijazahlagilagi.go` and is located in the directory `F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12`. The program's logic is as follows:

```
5 func main() {
7     fmt.Scan(&nama_tanaman)
8     switch nama_tanaman {
9     case "nepenthes", "drosera":
10        fmt.Println("Termasuk Tanaman Karnivora.")
11        fmt.Println("Asli Indonesia.")
12    case "venus", "sarracenia":
13        fmt.Println("Termasuk Tanaman Karnivora.")
14        fmt.Println("Tidak Asli Indonesia.")
15    default:
16        fmt.Println("Tidak termasuk Tanaman Karnivora.")
17    }
18 }
19
```

The terminal output shows the program being run with the following inputs and outputs:

```
nepenthes
Termasuk Tanaman Karnivora.
Asli Indonesia.

F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg
agilagi.go"
venus
Termasuk Tanaman Karnivora.
Tidak Asli Indonesia.

F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg
agilagi.go"
karedok
```

The output also shows a window titled "Nama" with the following information:

```
Nama : Dadi Maulana Muhammad
NIM : 109082500080
Kelas : S1IF-13-04
```

Deskripsi program

Program ini mengidentifikasi apakah suatu tanaman yang dimasukkan oleh pengguna merupakan tanaman karnivora serta menentukan asalnya. Mula-mula program membaca sebuah input berupa nama tanaman dan menyimpannya dalam variabel `nama_tanaman`. Kemudian program menggunakan struktur `switch` untuk mencocokkan nama tanaman tersebut dengan beberapa kategori yang telah ditentukan.

Jika tanaman yang dimasukkan adalah "nepenthes" atau "drosera", program menampilkan bahwa tanaman tersebut termasuk tanaman karnivora dan merupakan tanaman asli Indonesia. Jika tanaman yang dimasukkan adalah "venus" atau "sarracenia", program tetap menyatakan tanaman tersebut sebagai tanaman karnivora, tetapi bukan asli Indonesia. Apabila nama tanaman tidak sesuai dengan salah satu kategori di atas, program menampilkan bahwa tanaman tersebut tidak termasuk tanaman karnivora.

Dengan demikian, program ini membantu pengguna mengenali jenis tanaman berdasarkan nama yang diberikan.

4. Guided 4

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)
}
```

Screenshoot program

The screenshot shows a Go program in a code editor and its execution in a terminal. The program is a parking fee calculator. It defines a `main` function that prompts the user for vehicle type and duration, then calculates the fee based on a `switch` statement. The terminal shows two runs: one for a truck with a 1-hour duration resulting in a 0 fee (invalid input), and another for a motorcycle with a 2-hour duration resulting in a 7000 fee.

```
ijazahlagipalsu.go 1 x
ijazahlagipalsu.go > ...
5 func main() {
17     tarif = 9000
18     case kendaraan == "Mobil" && durasi >= 1 && durasi <= 2:
19         tarif = 15000
20     case kendaraan == "Mobil" && durasi > 2:
21         tarif = 20000
22     case kendaraan == "Truk" && durasi >= 1 && durasi <= 2:
23         tarif = 25000
24     case kendaraan == "Truk" && durasi > 2:
25         tarif = 35000
26     default:
27         fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")
28     }
29     fmt.Printf("Tarif Parkir: Rp %d\n", tarif)
30 }
31

PROBLEMS 23 OUTPUT DEBUG CONSOLE TERMINAL PORTS

F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg
agilagipalsu.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): truk
Masukkan durasi parkir (dalam jam): 1
Jenis kendaraan atau durasi parkir tidak valid
Tarif Parkir: Rp 0

F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg
agilagipalsu.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): sepeda
Masukkan durasi parkir (dalam jam): 2
Jenis kendaraan atau durasi parkir tidak valid
```

Deskripsi program

Program ini digunakan untuk menghitung tarif parkir berdasarkan jenis kendaraan dan durasi parkir dalam jam. Pada awal program, pengguna diminta memasukkan jenis kendaraan (Motor, Mobil, atau Truk) dan durasi parkir. Kedua nilai tersebut kemudian diproses menggunakan struktur *switch* untuk menentukan tarif yang sesuai.

Penentuan tarif dibedakan berdasarkan dua kategori durasi, yaitu 1–2 jam dan lebih dari 2 jam. Untuk kendaraan **Motor**, tarifnya sebesar Rp7000 untuk durasi 1–2 jam, dan Rp9000 jika durasi lebih lama. Untuk kendaraan **Mobil**, tarifnya Rp15000 untuk 1–2 jam dan Rp20000 jika lebih dari 2 jam. Sedangkan untuk kendaraan **Truk**, tarif yang dikenakan adalah Rp25000 untuk 1–2 jam dan Rp35000 jika melebihi 2 jam.

Jika pengguna memasukkan jenis kendaraan yang tidak dikenali atau durasi yang tidak sesuai aturan, program akan menampilkan pesan bahwa input tidak valid. Setelah tarif berhasil ditentukan, program akan menampilkan total biaya parkir dalam format “Rp <tarif>”.

Dengan demikian, program ini membantu pengguna dalam menghitung tarif parkir berdasarkan jenis kendaraan dan lamanya parkir.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var ph float64
    fmt.Print("Masukkan pH: ")
    fmt.Scan(&ph)

    switch {
    case ph < 0 || ph > 14:
        fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")

    case ph >= 6.5 && ph <= 8.6:
        fmt.Println("Air layak minum")

    default:
        fmt.Println("Air tidak layak minum")
    }
}
```

Screenshot program

The screenshot shows a Go program in a text editor and its execution output in a terminal. The program, named `ijazahPalsu.go`, defines a `main` function that takes a float64 value `ph` as input. It uses a `switch` statement to validate the pH value. If the pH is less than 0 or greater than 14, it prints "Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.". If the pH is between 6.5 and 8.6, it prints "Air layak minum". Otherwise, it prints "Air tidak layak minum".

```
4 func main() {
5     var ph float64
6     fmt.Print("Masukkan pH: ")
7     fmt.Scan(&ph)
8
9
10    switch {
11    case ph < 0 || ph > 14:
12        fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
13
14    case ph >= 6.5 && ph <= 8.6:
15        fmt.Println("Air layak minum")
16
17    default:
18        fmt.Println("Air tidak layak minum")
19    }
20 }
21
```

The terminal output shows the program being run from the directory `F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12`. It prompts for a pH value. When the user enters 8.6, it prints "Air layak minum". When the user enters 9, it prints "Air tidak layak minum". When the user enters 16, it prints "Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.".

```
F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "F:\Me\Others\Kuliah\Alg
alsu.go"
Masukkan pH: 9
Air tidak layak minum

F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "F:\Me\Others\Kuliah\Alg
alsu.go"
Masukkan pH: 16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.
```

Deskripsi program

Program ini memeriksa kelayakan air minum berdasarkan nilai pH yang diinput oleh pengguna. Mula-mula program meminta pengguna memasukkan nilai pH dan menyimpannya dalam variabel `ph`. Kemudian program menggunakan struktur *switch* untuk menentukan kategori dari nilai pH tersebut. Jika nilai pH berada di luar rentang 0 hingga 14, program menampilkan pesan bahwa nilai pH tidak valid. Jika nilai pH berada dalam rentang 6,5 sampai 8,6, maka program menyatakan bahwa air tersebut layak minum. Selain dari kondisi tersebut, air dianggap tidak layak minum. Hasil analisis kemudian ditampilkan pada layar.

2. Tugas 2

Source code

```
package main

import "fmt"

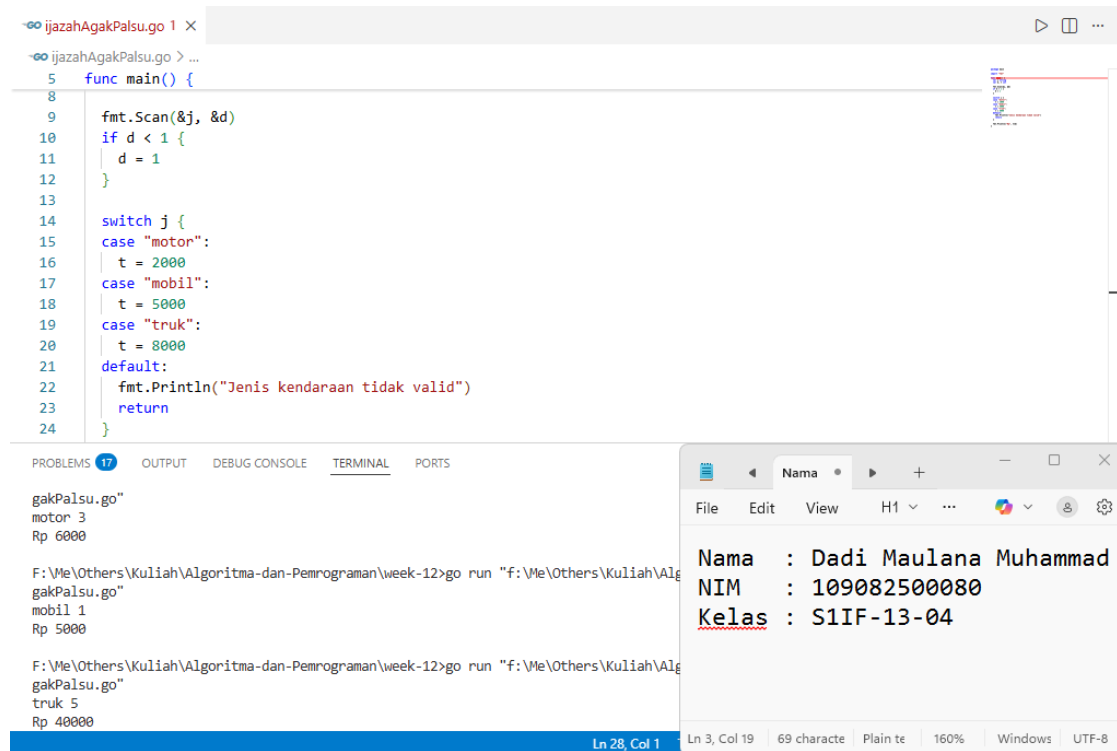
func main() {
    var j string
    var d, t int

    fmt.Scan(&j, &d)
    if d < 1 {
        d = 1
    }

    switch j {
    case "motor":
        t = 2000
    case "mobil":
        t = 5000
    case "truk":
        t = 8000
    default:
        fmt.Println("Jenis kendaraan tidak valid")
        return
    }

    fmt.Println("Rp", t*d)
}
```

Screenshoot program



The screenshot shows a Go program in a text editor and its execution output in a terminal. The program is named `ijazahAgakPalsu.go`. The code defines a `main` function that takes two arguments: `j` (vehicle type) and `d` (duration in hours). It uses a `switch` statement to determine the parking rate per hour based on the vehicle type: `motor` (Rp 2000), `mobil` (Rp 5000), and `truk` (Rp 8000). If the vehicle type is not recognized, it prints an error message. The duration `d` is checked to ensure it is at least 1 hour; if not, it is set to 1. The total cost is calculated and printed in the format "Rp <jumlah>".

```
5 func main() {  
8  
9     fmt.Scan(&j, &d)  
10    if d < 1 {  
11        d = 1  
12    }  
13  
14    switch j {  
15    case "motor":  
16        t = 2000  
17    case "mobil":  
18        t = 5000  
19    case "truk":  
20        t = 8000  
21    default:  
22        fmt.Println("Jenis kendaraan tidak valid")  
23        return  
24    }
```

The terminal output shows three test cases:

```
gakPalsu.go"  
motor 3  
Rp 6000  
  
F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg  
gakPalsu.go"  
mobil 1  
Rp 5000  
  
F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Alg  
gakPalsu.go"  
truk 5  
Rp 40000
```

Deskripsi program

Program ini menghitung biaya parkir berdasarkan jenis kendaraan dan durasi parkir yang dimasukkan oleh pengguna. Program terlebih dahulu membaca input berupa jenis kendaraan (motor, mobil, atau truk) dan durasi parkir dalam jam. Jika durasi parkir kurang dari satu jam, program akan menetapkannya menjadi satu jam karena durasi minimum yang dihitung adalah satu jam.

Selanjutnya, program menggunakan struktur *switch* untuk menentukan tarif parkir per jam sesuai jenis kendaraan: motor dikenakan tarif Rp2000 per jam, mobil Rp5000 per jam, dan truk Rp8000 per jam. Jika pengguna memasukkan jenis kendaraan yang tidak dikenali, program akan menampilkan pesan bahwa jenis kendaraan tidak valid dan menghentikan proses.

Jika jenis kendaraan valid, program menghitung total biaya parkir dengan mengalikan tarif per jam dengan durasi parkir, kemudian menampilkan total biaya dalam format "Rp <jumlah>" ke layar.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var x int
    fmt.Scan(&x)

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

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

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

    case x%2 == 0 && x%10 != 0 && x%5 != 0:
        fmt.Println("Kategori: Bilangan Genap")
        hasil := x * (x + 1)
        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d
* %d = %d\n", x, x+1, hasil)
    }
}
```

Screenshoot program

The screenshot shows a Go program in a code editor and its execution output in a terminal. The program is named `ijaszsahPalsuBanget.go` and is located in the directory `F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12`. The program uses the `fmt` package and a `switch` statement to categorize a number `x` based on its divisibility by 10 and 5. The terminal output shows two runs: one for `x=25` (category: Bilangan Kelipatan 5, result: 625) and one for `x=20` (category: Bilangan Kelipatan 10, result: 2).

```
2
3 import "fmt"
4
5 func main() {
6     var x int
7     fmt.Scan(&x)
8
9     switch {
10    case x%10 == 0:
11        fmt.Println("Kategori: Bilangan Kelipatan 10")
12        hasil := x / 10
13        fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", x, hasil)
14
15    case x%5 == 0:
16        fmt.Println("Kategori: Bilangan Kelipatan 5")
17        hasil := x * x
18        fmt.Printf("Hasil kuadrat dari %d^2 = %d\n", x, hasil)
19    }
```

PROBLEMS 17 OUTPUT DEBUG CONSOLE TERMINAL PORTS

F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12\ijaszsahPalsuBanget.go"

25

Kategori: Bilangan Kelipatan 5

Hasil kuadrat dari 25^2 = 625

F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12>go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-12\ijaszsahPalsuBanget.go"

20

Kategori: Bilangan Kelipatan 10

Hasil pembagian antara 20 / 10 = 2

Ln 29, Col 5

Deskripsi program

Program ini mengelompokkan sebuah bilangan yang dimasukkan oleh pengguna ke dalam kategori tertentu, kemudian melakukan operasi perhitungan sesuai kategori tersebut. Program membaca sebuah bilangan bulat dan menyimpannya dalam variabel `x`. Setelah itu, program menggunakan struktur `switch` untuk menentukan kategori bilangan berdasarkan sifat pembagiannya.

Jika bilangan merupakan kelipatan 10, program mengkategorikannya sebagai "Bilangan Kelipatan 10" dan menghitung hasil pembagian bilangan tersebut dengan 10. Jika bukan kelipatan 10 namun kelipatan 5, bilangan masuk kategori "Bilangan Kelipatan 5" dan program menghitung kuadrat dari bilangan tersebut.

Apabila bilangan tersebut bukan kelipatan 5 atau 10 dan bernilai ganjil, program mengkategorikannya sebagai "Bilangan Ganjil", lalu menghitung penjumlahan bilangan dengan bilangan setelahnya. Jika bilangan termasuk genap namun bukan kelipatan 5 maupun 10, program menandainya sebagai "Bilangan Genap" dan menghitung hasil perkalian bilangan dengan bilangan berikutnya.

Setelah perhitungan sesuai kategori dilakukan, hasilnya akan ditampilkan ke layar. Program memastikan setiap bilangan hanya masuk ke satu kategori yang paling tepat.