

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 11
SWITCH - CASE**



Disusun oleh:

Janica Prima Ginting

109082500064

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 jam12, jam24 int

    var ampm string

    fmt.Scan(&jam24)

    switch {

    case jam24 == 0 :

        jam12 = 12

        ampm = "AM"

    case jam24 < 12 :

        jam12 = jam24

        ampm = "AM"

    case jam24 == 12 :

        jam12 = 12

        ampm = "PM"

    case jam24 > 12 :

        jam12 = jam24 - 12

        ampm = "PM"

    }

    fmt.Println(jam12, ampm)

}
```

Screenshoot program

```
Tugas1 > -go Tugas1.go > main
5 func main(){
9     switch {
10     case jam24 == 0 :
11         jam12 = 12
12         ampm = "AM"
13     case jam24 < 12 :
14         jam12 = jam24
15         ampm = "AM"
16     case jam24 == 12 :
17         jam12 = 12
18         ampm = "PM"
19     case jam24 > 12 :
20         jam12 = jam24 - 12
21         ampm = "PM"
22     }
23     fmt.Println(jam12, ampm)
24 }
```

File Edit View A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _ ` { | } ~

Nama : Janica Prima Ginting
NIM : 10908250064

Ln 1, Col 1 | 46 character | Plain t | 100% | Wind UTF-8

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas1/Tugas1.go
13
1 PM
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas1/Tugas1.go
0
12 AM
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas1/Tugas1.go
12
12 PM
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11>

Ln 23, Col 24 | Tab Size: 4 | UTF-8 | CRLF | Go | 1.25.1

Deskripsi program

Program golang ini berfungsi untuk melakukan konversi waktu dari bentuk 24 jam ke dalam bentuk 12 jam.

Cara Kerja Program :

Pertama variable jam12,jam24 dideklarasikan dengan tipe data integer

Kedua variable ampm dideklarasikan dengan tipe data string

Ketiga fmt.Scan untuk pengguna menginputkan bilangan bulat di jam24

Keempat masuk ke switch

- case jam24 sama dengan 0 , maka jam12 diisi dengan 12, ampm diisi dengan AM
- case jam24 lebih kecil dari 12 , maka jam12 diisi dengan jam24 , ampm diisi dengan AM
- case jam24 sama dengan 12 , maka jam12 diisi dengan 12 , ampm diisi dengan PM
- case jam24 lebih besar dari 12 , maka jam diisi dengan jam24 dikurang 12 , ampm diisi dengan PM

Terakhir fmt.Print(jam12, ampm) untuk mengeluarkan nilai dari jam12 dan ampm.

2. Guided 2

Source Code

```
package main

import "fmt"

func main(){

    var tnmn string

    fmt.Scan(&tnmn)

    switch tnmn {

    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.Print("Tidak termasuk Tanaman Karnivora")

    }

}
```

Screenshoot program

```
Tugas2 > -go Tugas2.go > main
1 package main
2
3 import "fmt"
4
5 func main(){
6     var tnmn string
7     fmt.Scan(&tnmn)
8     switch tnmn {
9     case "nepenthes" :
10         fmt.Println("Termasuk Tanaman Karnivora")
11         fmt.Println("Asli Indonesia")
12     case "venus" :
13         fmt.Println("Termasuk Tanaman Karnivora")
14         fmt.Println("Bukan Asli Indonesia")
15     default :
16         fmt.Print("Tidak termasuk Tanaman Karnivora")
17     }
```

File Edit View A A 100% Wind UTF-8

Nama : Janica Prima Ginting
NIM : 109082500064

Ln 1, Col 1 | 46 character Plain t | 100% | Wind UTF-8

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas2/Tugas2.go
nepenthes
Termasuk Tanaman Karnivora
Asli Indonesia
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas2/Tugas2.go
venus
Termasuk Tanaman Karnivora
Bukan Asli Indonesia
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas2/Tugas2.go
karedok
Tidak termasuk Tanaman Karnivora

Ln 6, Col 20 Tab Size: 4 UTF-8 CRLF { Go 1.25.1

Deskripsi program

Program Golang ini berfungsi untuk menentukan apakah tanaman tersebut termasuk tanaman karnivora atau tidak. Jika ya, apakah tanaman tersebut asli indonesia atau tidak.

Cara kerja program :

Pertama variable tnmn dideklarasikan dengan tipe data string

Kedua fmt.Scan untuk pengguna menginputkan huruf / teks

Ketiga masuk ke switch jika tnmn

Terakhir

case "nepenthes"

output "Termasuk Tanaman Karnivora"

output "Asli Indonesia"

case "venus"

output "Termasuk Tanaman Karnivora"

output "Bukan Asli Indonesia"

default :

/ diluar nepenthes / venus

output "Tidak termasuk Tanaman Karnivora"

3. Guided 3

Source Code

```
package main

import "fmt"

func main () {

    var kendaraan string

    var durasi int

    var tarif int

    fmt.Scan(&kendaraan)

    fmt.Scan(&durasi)

    fmt.Println("Masukkan jenis kendaraan  
(Motor/Mobil/Truk):", kendaraan)

    fmt.Println("Masukkan durasi parkir (dalam  
jam):", 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.Println("Tarif Parkir: Rp",tarif)

    }
}
```

Screenshoot program

The image shows a Go program in a text editor and its execution output in a terminal. The program is a parking fee calculator that takes vehicle type and duration as input and outputs the corresponding parking fee. The execution shows several test cases for different vehicle types and durations, including a default case for invalid input.

```
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11>
Motor
2
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 2
Tarif Parkir: Rp 7000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11>
Mobil
4
Masukkan jenis kendaraan (Motor/Mobil/Truk): Mobil
Masukkan durasi parkir (dalam jam): 4
Tarif Parkir: Rp 20000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas3/Tugas3.go
Motor
3
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 3
Tarif Parkir: Rp 9000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas3/Tugas3.go
Truk
1
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir (dalam jam): 1
Tarif Parkir: Rp 25000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas3/Tugas3.go
Sepeda
2
Masukkan jenis kendaraan (Motor/Mobil/Truk): Sepeda
Masukkan durasi parkir (dalam jam): 2
Jenis Kendaraan atau Durasi Parkir tidak Valid
Tarif Parkir: Rp 0
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11>
```

```
2
3 import "fmt"
4
5 func main(){
6     var kendaraan string
7     var durasi int
8     var tarif int
9     fmt.Scan(&kendaraan)
10    fmt.Scan(&durasi)
11    fmt.Println("Masukkan jenis kendaraan (Motor/Mobil/Truk):",kendaraan)
12    fmt.Println("Masukkan durasi parkir (dalam jam):",durasi)
13    switch {
14    case kendaraan == "Motor" && durasi >= 1 && durasi <= 2 :
15        |   tarif = 7000
16    case kendaraan == "Motor" && durasi > 2 :
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.Println("Tarif Parkir: Rp",tarif)
30 }
```

Deskripsi program

Program Golang ini berfungsi untuk menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parkir. Terdapat tiga jenis kendaraan yang dapat diparkir dengan tarif berbeda berdasarkan durasi parkir. Untuk motor, tarifnya adalah Rp 7.000 untuk durasi 1-2 jam, dan Rp 9.000 jika durasi lebih dari 2 jam. Mobil dikenakan tarif Rp 15.000 untuk 1-2 jam, dan Rp 20.000 untuk durasi lebih dari 2 jam. Sementara itu, truk dikenakan tarif Rp 25.000 untuk 1-2 jam, dan Rp 35.000 jika lebih dari 2 jam.

Cara kerja program :

Pertama variable kendaraan dideklarasikan dengan tipe data string

Kedua variable durasi,tarif dideklarasikan dengan tipe data integer

Ketiga Scan untuk menginputkan bilangan bulat kedalam var kendaraan dan durasi

Keempat fmt.Print untuk mengeluarkan output berupa teks

Kelima masuk kedalam switch

Keenam

case kendaraan sama dengan Motor AND durasi lebih besar dari 1 AND durasi lebih kecil dari 2, maka Tarif diisi dengan 7000

case kendaraan sama dengan Motor AND durasi lebih besar dari 2, maka Tarif diisi dengan 9000

case kendaraan sama dengan Mobil AND durasi lebih besar dari 1 AND durasi lebih kecil dari 2 , maka Tarif diisi dengan 15000

case kendaraan sama dengan Mobil AND durasi lebih besar dari 2, maka Tarif diisi dengan 20000

case kendaraan sama dengan Truk AND durasi lebih besar dari 1 AND durasi lebih kecil dari 2 , maka Tarif diisi dengan 25000

case kendaraan sama dengan Truk AND durasi lebih besar dari 2, maka Tarif diisi dengan 35000

Default : / Jika tidak ada diantara case diatas

fmt.Print untuk mengeluarkan output Jenis Kendaraan atau Durasi Parkir tidak Valid

Terakhir fmt.Print untuk mengeluarkan output dari nilai Tarif

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main(){

    var ph float64

    fmt.Scan(&ph)

    switch {

    case ph >= 6.5 && ph <= 8.6 :

        fmt.Println("Air layak minum")

    case ph < 6.5 || ph > 8.6 && ph <= 14 :

        fmt.Println("Air tidak layak minum")

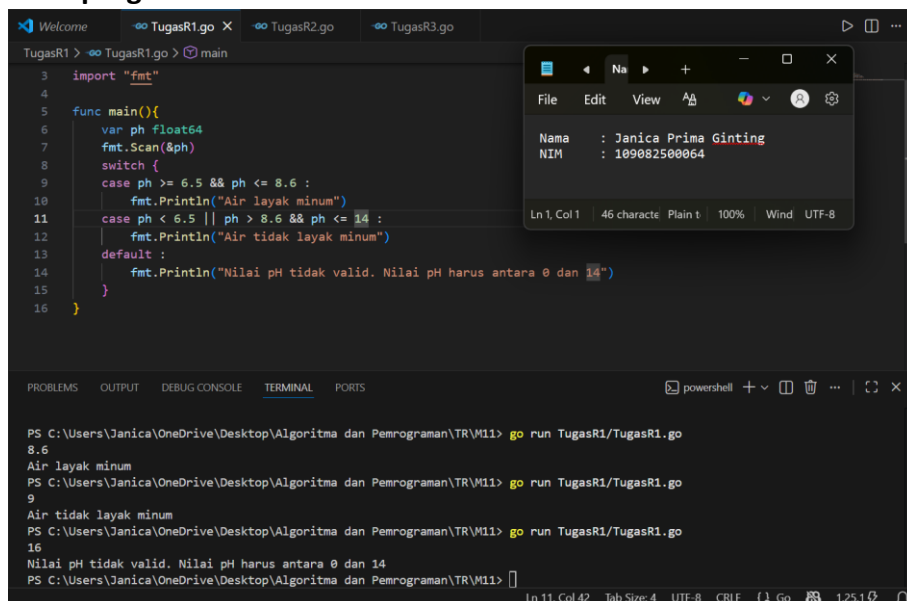
    default :

        fmt.Println("Nilai pH tidak valid. Nilai pH
        harus antara 0 dan 14")

    }

}
```

Screenshoot program



The screenshot shows a Go IDE with the following components:

- Editor:** Displays the source code for `TugasR1.go`. The code is a Go program that reads a pH value and prints a message based on its range.
- Terminal:** Shows the execution of the program. The user runs `go run TugasR1/TugasR1.go` three times, entering different pH values (8.6, 9, and 16) and receiving the corresponding output messages.
- Properties Panel:** Displays the file's name (`TugasR1.go`) and its location.

```
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\W11> go run TugasR1/TugasR1.go
8.6
Air layak minum
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\W11> go run TugasR1/TugasR1.go
9
Air tidak layak minum
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\W11> go run TugasR1/TugasR1.go
16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\W11>
```

Deskripsi program

Program Golang ini berfungsi untuk menentukan apakah kadar pH pada air yang diinput termasuk Air yang layak untuk diminum atau tidak.

Cara kerja program :

Pertama variable ph dideklarasikan dengan tipe data float64

Kedua fmt.Scan(&ph) untuk pengguna menginputkan berapa ph

Terakhir masuk ke switch

case ph lebih besar sama dengan 6.5 AND ph lebih kecil sama dengan 8.6

maka output Air layak Minum

case ph lebih kecil dari 6.5 OR ph lebih besar dari 8.6 AND lebih kecil sama dengan 14

maka ouput Air tidak layak Minum

default:

output Nilai pH tidak valid. Nilai pH harus antara 0 dan 14

2. Tugas 2

Source code

```
package main

import "fmt"

func main(){

    var kendaraan string

    var durasi int

    fmt.Scan(&kendaraan)

    fmt.Scan(&durasi)

    biaya := 0

    if durasi < 1 {

        durasi = 1

    }

    switch kendaraan {

    case "motor" :

        biaya = 2000

    case "mobil" :

        biaya = 5000

    case "truk" :

        biaya = 8000

    default :

        fmt.Println("Kendaraan Tidak Valid")

    }

    fmt.Println("Rp",biaya * durasi)

}
```

Screenshoot program

The screenshot shows the Visual Studio Code editor with the file `TugasR2.go` open. The code defines a `main` function that calculates parking fees based on vehicle type and duration. A floating window displays the user's name and NIM. The terminal shows the execution of the program for three different inputs: motor, mobil, and truk.

```
5 func main(){
10     biaya := 0
11     if durasi < 1 {
12         durasi = 1
13     }
14     switch kendaraan {
15     case "motor" :
16         biaya = 2000
17     case "mobil" :
18         biaya = 5000
19     case "truk" :
20         biaya = 8000
21     default :
22         fmt.Println("Kendaraan Tidak Valid")
23     }
24     fmt.Println("Rp",biaya * durasi)
25 }
```

motor
3
Rp 6000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M11> go run TugasR2/TugasR2.go
mobil
1
Rp 5000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M11> go run TugasR2/TugasR2.go
truk
5
Rp 40000

The screenshot shows the Visual Studio Code editor with the file `Tugas3.go` open. The code defines a `main` function that calculates parking fees based on vehicle type and duration. A floating window displays the user's name and NIM. The terminal shows the execution of the program for four different inputs: motor, mobil, truk, and sepeda.

```
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11>
Motor
2
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 2
Tarif Parkir: Rp 7000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11>
Mobil
4
Masukkan jenis kendaraan (Motor/Mobil/Truk): Mobil
Masukkan durasi parkir (dalam jam): 4
Tarif Parkir: Rp 20000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas3/Tugas3.go
Motor
3
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 3
Tarif Parkir: Rp 9000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas3/Tugas3.go
Truk
1
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir (dalam jam): 1
Tarif Parkir: Rp 25000
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11> go run Tugas3/Tugas3.go
Sepeda
2
Masukkan jenis kendaraan (Motor/Mobil/Truk): Sepeda
Masukkan durasi parkir (dalam jam): 2
Jenis Kendaraan atau Durasi Parkir tidak Valid
Tarif Parkir: Rp 0
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M11>
```

Deskripsi program

Program Golang ini berfungsi untuk menghitung tarif parkir berdasarkan jenis kendaraan dan durasi parkir yang dimasukkan oleh pengguna. Ada tiga jenis kendaraan: motor, mobil, dan truk, dengan masing-masing memiliki tarif parkir yang berbeda. Motor dikenakan tarif sebesar Rp 2.000 per jam, mobil sebesar Rp 5.000 per jam, dan truk sebesar Rp 8.000 per jam.

Cara kerja Program :

Pertama variable kendaraan dideklarasikan dengan tipe data string

Kedua variable durasi dideklarasikan dengan tipe data integer

Ketiga `fmt.Scan(&kendaraan)` untuk menginput kata , `fmt.Scan(&durasi)` untuk menginput angka

Keempat biaya disii nilai 0

Kelima fungsi if jika durasi lebih kecil dari 1 maka durasi sama dengan 1

Keenam masuk ke switch kendaraan

case motor maka biaya diberi nilai 2000

case mobil maka biaya diberi nilai 5000

case truk maka biaya diberi nilai 8000

default :

mengeluarkan output Kendaraan Tidak Valid

Terakhir `fmt.Println` untuk mengeluarkan output Rp dengan biaya dikali durasi.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {

    var bilangan int

    fmt.Scan(&bilangan)

    switch {

    case bilangan % 10 == 0 :

        kelipatan10 := bilangan / 10

        fmt.Println("Kategori: Bilangan Kelipatan 10")

        fmt.Printf("Hasil pembagian antara %d / 10 = %d",bilangan,kelipatan10)

    case bilangan % 5 == 0 && bilangan != 5 :

        kelipatan5 := bilangan * bilangan

        fmt.Println("Kategori: Bilangan Kelipatan 5")

        fmt.Printf("Hasil kuadrat dari %d^2 = %d",bilangan,kelipatan5)

    case bilangan % 2 == 0 :

        bilangangenap := bilangan * (bilangan + 1)

        fmt.Println("Kategori: Bilangan Genap")

        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d",bilangan,bilangan+1,bilangangenap)

    default :

        bilanganganjil := bilangan + (bilangan + 1)

        fmt.Println("Kategori: Bilagan Ganjil")

    }
```

```

        fmt.Printf("Hasil penjumlahan dengan bilangan
berikutnya %d + %d = %d",
bilangan,bilangan+1,bilanganganjil)

    }

}

```

Screenshoot program

The screenshot shows a Go program in VS Code and its execution output in the terminal. The program is a switch statement that categorizes a number and performs operations based on the category.

Source Code (TugasR3.go):

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan int
7     fmt.Scan(&bilangan)
8     switch {
9     case bilangan % 10 == 0 :
10        kelipatan10 := bilangan / 10
11        fmt.Println("Kategori: Bilangan Kelipatan 10")
12        fmt.Printf("Hasil pembagian antara %d / 10 = %d",bilangan,kelipatan10)
13    case bilangan % 5 == 0 && bilangan != 5 :
14        kelipatan5 := bilangan * bilangan
15        fmt.Println("Kategori: Bilangan Kelipatan 5")
16        fmt.Printf("Hasil kuadrat dari %d^2 = %d",bilangan,kelipatan5)
17    case bilangan % 2 == 0 :
18        bilanganenap := bilangan * (bilangan + 1)
19        fmt.Println("Kategori: Bilangan Genap")
20        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d",bilangan,bilangan+1,bilang
21    default :
22        bilanganganjil := bilangan + (bilangan + 1)
23        fmt.Println("Kategori: Bilagan Ganjil")
24        fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d", bilangan,bilangan+1,bilang
25    }
26 }

```

Terminal Output:

```

PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M11> go run TugasR3.go
5
Kategori: Bilagan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M11> go run TugasR3.go
8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M11> go run TugasR3.go
25
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 25^2 = 625
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M11> go run TugasR3.go
20
Kategori: Bilangan Kelipatan 10
Hasil pembagian antara 20 / 10 = 2
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M11>

```

Deskripsi program

Program Golang ini berfungsi untuk mengidentifikasi pola aritmatika berdasarkan bilangan yang diinputkan dan melakukan operasi matematika yang sesuai.

Ketentuan :

- a. Bilangan Ganjil : Menghitung penjumlahan antara bilangan yang diinput dengan bilangan berikutnya
- b. Bilangan Genap : Menghitung perkalian antara bilangan yang diinput dengan bilangan berikutnya
- c. Bilangan Kelipatan 5 : Menghitung hasil kuadrat dari bilangan yang diinputkan
- d. Bilangan Kelipatan 10 : Membagi bilangan yang diinputkan dengan bilangan 10

Cara kerja program :

Pertama variable bilangan dideklarasikan dengan tipe data integer

Kedua `fmt.Scan(&bilangan)` untuk pengguna menginputkan angka

Terakhir masuk ke switch

case bilangan dibagi 10 sisa sama dengan 0 maka

kelipatan10 diisi dengan bilangan dibagi 10

output Kategori: Bilangan Kelipatan 10

output hasil dari Hasil pembagian antara $\%d / 10 = \%d$, bilangan, kelipatan10

case bilangan dibagi 5 sisa sama dengan 0 AND bilangan tidak sama dengan 5 maka

kelipatan5 diisi dengan bilangan dikali bilangan

output Kategori: Bilangan Kelipatan 5

output hasil dari Hasil kuadrat dari $\%d^2 = \%d$, bilangan, kelipatan5

case bilangan dibagi 2 sisa sama dengan 0

bilangangenap diisi dengan bilangan dikali (bilangan ditambah 1)

output Kategori: Bilangan Genap

output hasil dari Hasil perkalian dengan bilangan berikutnya $\%d * \%d = \%d$, bilangan, bilangan+1, bilangangenap

default :

bilanganganjil diisi dengan bilangan ditambah (bilangan ditambah 1)

output Kategori: Bilangan Ganjil

output hasil dari Hasil penjumlahan dengan bilangan berikutnya $\%d + \%d = \%d$, bilangan, bilangan+1, bilanganganjil.