

**LAPORAN PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN 1**  
**MODUL 11**  
**“SWITCH-CASE”**



**DISUSUN OLEH:**  
**ANGGUN WAHYU WIDIYANA**  
**103112480280**  
**S1 IF-12-01**  
**DOSEN:**  
**Yohani Setiya Rafika Nur, M. Kom.**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2024/2025**

## DASAR TEORI

### Paradigma Percabangan

Pada dasarnya switch-case merupakan alternatif dari penggunaan else-if, tetapi pemilihan aksi tidak dilakukan berdasarkan kondisi, tetapi suatu nilai atau operasi tipe data dasar yang tidak menghasilkan boolean.

Penulisan switch case terdiri dari komponen berikut:

- Ekspresi**, merupakan operasi tipe data yang menghasilkan **nilai** selain tipe data boolean,
- Nilai**, pilihan case dari **ekspresi** yang dideklarasikan pada instruksi switch-case.
- Aksi**, merupakan kumpulan instruksi yang akan dieksekusi sesuai dengan nilai yang dihasilkan dari ekspresi yang terdapat pada switch-case. **Aksi lain** hanya akan dieksekusi apabila tidak ada pilihan nilai yang sesuai dengan hasil ekspresi.

Berikut adalah notasi dari switch-case

Notasi dalam psedeucode	Notasi dalam bahasa Go
depend on expresi nilai_1: // aksi 1 nilai_2: // aksi 2 nilai_3: // aksi 3 nilai_4: // aksi 4 .. dst. // dst. end	<b>switch</b> ekspresi <b>case</b> nilai_1: // aksi 1 <b>case</b> nilai_2: // aksi 2 <b>case</b> nilai_3: // aksi 3 <b>case</b> nilai_4: // aksi 4 <b>Default:</b> // aksi lain }

Kata kunci **default** digunakan untuk menyatakan pilihan untuk **aksi lain**. Selain itu terdapat juga penggunaan switch-case sebagai pengganti if-then, yang mana pada setiap case yang didefinisikan adalah kondisi dari struktur kontrol percabangan,

```
switch {  
    case kondisi_1 : // aksi kondisi 1  
    case kondisi_2 : // aksi kondisi 2  
    case kondisi_2 : // aksi kondisi 3  
    default : // aksi lain  
}
```

### Implementasi menggunakan Go

Sebagai contoh sederhana adalah program yang digunakan untuk menentukan batas atas dan bawah suatu indeks nilai mutu.

---

```

1  switch indeks {
2  case 'A':
3      batasA = 100
4      batasB = 75
5  case 'B':
6      batasA = 75
7      batasB = 65
8  case 'C':
9      batasA = 65
10     batasB = 50
11 default:
12     batasA = 50
13     batasB = 0
14 }
15 fmt.Printf( "Rentang nilai %v adalah: %v..%v\n", indeks, batasB,
    batasA )

```

```

16 switch {
17 case nilai > 75 && adaTubes:
18     indeks = 'A'
19 case nilai > 65:
20     indeks = 'B'
21 case nilai > 50 && pctHadir > 0.7:
22     indeks = 'C'
23 default:
24     indeks = 'F'
25 }
26 fmt.Printf( "Nilai %v dengan kehadiran %v%% dan buat tubes=%v,
    mendapat
27 indeks %c\n", nilai, pctHadir, adaTubes, indeks )

```

Contoh yang lain, misalnya suatu program digunakan untuk menampilkan hasil operasi aritmatika berdasarkan kriteria tertentu.

```

1  // filename: switchcase1.go
2  package main
3  import "fmt"
4
5  func main() {
6      var a int
7      var b int
8      fmt.Scan(&a, &b)
9      switch (a + b) % 4 {
10     case 0:
11         fmt.Println("Hasil dari", a, "+", b, "% 4 adalah", (a + b)
            % 4)
12         fmt.Println("Operasi: Penjumlahan")
13         fmt.Println("Hasil:", a, "+", b, "=", a+b)
14     case 1:
15         fmt.Println("Hasil dari", a, "+", b, "% 4 adalah", (a + b)
            % 4)
16         fmt.Println("Operasi: Pengurangan")
17         fmt.Println("Hasil:", a, "-", b, "=", a-b)
18     case 2:
19         fmt.Println("Hasil dari", a, "+", b, "% 4 adalah", (a + b)
            % 4)
20         fmt.Println("Operasi: Perkalian")
21         fmt.Println("Hasil:", a, "*", b, "=", a*b)

```

```

22     case 3:
23         fmt.Println("Hasil dari", a, "+", b, "% 4 adalah", (a + b)
24             % 4)
25         fmt.Println("Operasi: Pembagian")
26         if b != 0 {
27             fmt.Println("Hasil:", a, "/", b, "=", a/b)
28         } else {
29             fmt.Println("Error: Pembagi tidak boleh nol.")
30         }
31     default:
32         fmt.Println("Error: Pilihan tidak valid.")
33     }

```

C:\users\go\src\hello>go build switchcase1.go

C:\users\go\src\hello>switchcase1

10 6

Hasil dari 10 + 6 % 4 adalah 0

Operasi: Penjumlahan

Hasil: 10 + 6 = 16

C:\users\go\src\hello>switchcase1

7 4u

Hasil dari 7 + 4 % 4 adalah 3

Operasi: Pembagian

Hasil: 7 / 4 = 1

C:\users\go\src\hello>switchcase1

3 0

Hasil dari 3 + 0 % 4 adalah 3

Operasi: Pembagian

Error: Pembagi tidak boleh nol.

## CONTOH SOAL

### Latihan1

Buatlah Sebuah algoritma digunakan untuk melakukan konversi waktu dari bentuk 24 jam ke dalam bentuk 12 jam.

**Masukan** terdiri dari satu bilangan bulat dengan interval 0 s.d. 23 (inklusif) yang merepresentasikan jam dalam bentuk 24 jam.

**Keluaran** terdiri dari teks yang menyatakan format jam pada masukan dalam bentuk 12 jam.

**Contoh masukan dan keluaran:**

No	Masukan	Keluaran
1	13	1 PM
2	0	12 AM
3	12	12 PM

Source Code:

```
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)
}
```

Output:

```
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol1.go"
13
1 PM
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol1.go"
0
12 AM
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol1.go"
12
12 PM
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol1.go"
```

#### Deskripsi Program:

Program Go ini bertujuan untuk mengonversi waktu dari format 24 jam ke format 12 jam. Pengguna diminta untuk memasukkan waktu dalam bentuk bilangan bulat antara 0 hingga 23. Berdasarkan input yang diberikan, program menentukan jam dalam format 12 jam dan label AM atau PM yang sesuai. Jika waktu yang dimasukkan adalah 0, program akan menampilkan "12 AM"; jika kurang dari 12, jam ditampilkan langsung dengan label "AM"; jika sama dengan 12, ditampilkan "12 PM"; dan jika lebih dari 12, jam dikurangi 12 dengan label "PM". Program juga menangani input yang tidak valid dengan memberikan pesan kesalahan. Hasil konversi kemudian ditampilkan kepada pengguna.

---

## Latihan2

Buatlah program dengan bahasa Go yang digunakan untuk menentukan apakah tanaman tersebut termasuk tanaman karnivora atau tidak. Jika ya, apakah tanaman tersebut asli Indonesia atau tidak.

**Masukan** terdiri dari satu string, yakni nama tanaman tersebut.

**Keluaran** berupa teks "Termasuk Tanaman Karnivora" atau "Tidak termasuk Tanaman Karnivora", dan dilanjutkan "Asli Indonesia" atau "Bukan Asli Indonesia". Keluaran sesuai dengan masukan yang diberikan.

**Contoh masukan dan keluaran:**

No	Masukan	Keluaran
1	nepenthes	Termasuk Tanaman Karnivora Asli Indonesia
2	venus	Termasuk Tanaman Karnivora Bukan Asli Indonesia
3	karedok	Tidak termasuk Tanaman Karnivora

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("Bukan Asli Indonesia.")
    default:
        fmt.Println("Tidak termasuk Tanaman Karnivora.")
    }
}
```

Output:

```
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Coso12.go"
nepenthes
Termasuk Tanaman Karnivora.
Asli Indonesia.
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Coso12.go"
venus
Termasuk Tanaman Karnivora.
Bukan Asli Indonesia.
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Coso12.go"
karedok
Tidak termasuk Tanaman Karnivora.
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Coso12.go"
```

#### Deskripsi Program:

Program Go ini meminta pengguna untuk memasukkan nama tanaman dalam bentuk string. Menggunakan struktur switch, program memeriksa nama tanaman yang diinput dan menentukan kategori tanaman tersebut. Jika nama tanaman adalah "nepenthes" atau "drosera", program menyatakan bahwa tanaman tersebut termasuk dalam kategori tanaman karnivora dan asli Indonesia. Jika nama tanaman adalah "venus" atau "sarracenia", program juga menyatakan bahwa itu termasuk tanaman karnivora, tetapi bukan asli Indonesia. Untuk nama tanaman lainnya, program menyatakan bahwa tanaman tersebut tidak termasuk dalam kategori tanaman karnivora. Hasil keluaran ditampilkan sesuai dengan masukan yang diberikan.

---



### Latihan3

Buatlah program dalam bahasa Go yang 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.

**Masukan** terdiri dari 2 baris, baris pertama adalah string yang menyatakan jenis kendaraan sedangkan baris kedua merupakan bilangan bulat yang menyatakan durasi parkir dalam jam.

**Keluaran** berupa string yang menampilkan tarif parkir berdasarkan jenis kendaraan dan durasi parkir kendaraan atau "Jenis kendaraan atau durasi parkir tidak valid" jika jenis kendaraan selain dari motor, mobil dan truk.

#### Contoh masukan dan keluaran

No	Contoh masukan dan keluaran
1	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Motor</u> Masukkan durasi parkir (dalam jam): <u>2</u> Tarif Parkir: Rp 7000
2	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Mobil</u> Masukkan durasi parkir (dalam jam): <u>4</u> Tarif Parkir: Rp 20000
3	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Motor</u> Masukkan durasi parkir (dalam jam): <u>3</u> Tarif Parkir: Rp 9000
4	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Truk</u> Masukkan durasi parkir (dalam jam): <u>1</u> Tarif Parkir: Rp 25000
5	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Sepeda</u> Masukkan durasi parkir (dalam jam): <u>2</u> Jenis kendaraan atau durasi parkir tidak valid Tarif Parkir: Rp 0

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)
}

```

Output:

```

PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 2
Tarif Parkir: Rp 7000
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Mobil
Masukkan durasi parkir (dalam jam): 4
Tarif Parkir: Rp 20000
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 3
Tarif Parkir: Rp 9000
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir (dalam jam): 1
Tarif Parkir: Rp 25000
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\ContohSoal\Cosol3.go"
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\User\Documents\MODUL 11>

```

Deskripsi Program:

Program Go ini berfungsi untuk menentukan tarif parkir berdasarkan jenis kendaraan (Motor, Mobil, atau Truk) dan durasi parkir dalam jam. Pengguna diminta untuk memasukkan jenis kendaraan dan durasi parkir. Berdasarkan input tersebut, program menghitung tarif parkir sesuai dengan ketentuan yang telah ditetapkan: tarif berbeda untuk durasi 1-2 jam dan lebih dari 2 jam untuk setiap jenis kendaraan. Jika jenis kendaraan atau durasi parkir tidak valid, program akan menampilkan pesan kesalahan. Hasil tarif parkir kemudian ditampilkan kepada pengguna.

## SOAL LATIHAN

### Latihan1

Buatlah program dengan bahasa Go yang digunakan untuk menentukan apakah kadar pH pada air yang diinput termasuk Air yang layak untuk diminum atau tidak.

**Masukan** terdiri dari satu float, yakni kadar pH tersebut.

**Keluaran** berupa teks seperti contoh "Air Layak Minum" (ph  $\geq 6.5$  dan ph  $\leq 8.6$ ) atau "Air Tidak Layak Minum" (ph  $< 6.5$  dan ph  $> 8.6$ ), dan "Input tidak valid, rentang pH 0 -14" jika nilai float diatas 14. Keluaran sesuai dengan masukan yang diberikan.

### Contoh masukan dan keluaran

No	Masukan	Keluaran
1	8.6	Air layak minum
2	9	Air tidak layak minum
3	16	Nilai pH tidak valid. Nilai pH harus antara 0 dan 14

### Source Code:

```
package main
import "fmt"
func main() {
    var ph float64
    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.")
    }
}
```

### Output:

```
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol1.go"
8.6
Air layak minum.
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol1.go"
9
Air tidak layak minum.
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol1.go"
16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol1.go"
```

Deskripsi Program:

Program Go ini berfungsi untuk menentukan apakah kadar pH air yang diinput oleh pengguna layak untuk diminum. Pengguna diminta memasukkan nilai pH dalam bentuk float, dan program akan memvalidasi input tersebut. Jika nilai pH kurang dari 0 atau lebih dari 14, program akan menampilkan pesan bahwa nilai pH tidak valid. Jika nilai pH berada dalam rentang 6.5 hingga 8.6, program akan menyatakan bahwa air layak untuk diminum, sedangkan jika berada di luar rentang tersebut, program akan menginformasikan bahwa air tidak layak untuk diminum.

---

## Latihan2

Buatlah program dalam bahasa Go 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.

Program harus dapat menentukan tarif per jam berdasarkan jenis kendaraan, lalu mengalikan tarif tersebut dengan jumlah jam parkir untuk menghasilkan total biaya parkir. Jika durasi parkir kurang dari 1 jam, maka durasi tetap dianggap 1 jam.

**Masukan** berupa jenis kendaraan dan durasi parkir dalam jam.

**Keluaran** berupa total biaya parkir berdasarkan jenis kendaraan dan durasi parkir.

### Contoh masukan dan keluaran

No	Masukan	Keluaran
1	motor 3 jam	Rp 6000
2	mobil 1 jam	Rp 5000
3	truk 5 jam	Rp 40000

### Source Code:

```
package main
import "fmt"
func main() {
    var kendaraan string
    var durasi int
    var tarif int
    fmt.Scan(&kendaraan)
    fmt.Scan(&durasi)
    if durasi < 1 {
        durasi = 1
    }
    switch kendaraan {
    case "motor":
        tarif = durasi * 2000

    case "mobil":
        tarif = durasi * 5000
    case "truk":
        tarif = durasi * 8000
    default:
        fmt.Println("Jenis kendaraan atau durasi parkir tidak
valid")
    }
    fmt.Printf("Rp %d\n", tarif)
}
```

Output:

```
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol2.go"
motor 3 jam
Rp 6000
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol2.go"
mobil 1 jam
Rp 5000
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol2.go"
truk 5 jam
Rp 40000
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol2.go"
```

#### Deskripsi Program:

Program Go meminta pengguna untuk memasukkan jenis kendaraan (motor, mobil, atau truk) dan durasi parkir dalam jam. Jika durasi parkir kurang dari 1 jam, program akan menganggapnya sebagai 1 jam. Berdasarkan jenis kendaraan, program menghitung total biaya parkir dengan mengalikan tarif per jam yang sesuai dengan jumlah jam parkir. Jika pengguna memasukkan jenis kendaraan yang tidak valid, program akan menampilkan pesan kesalahan dan keluar tanpa menghitung biaya. Setelah semua input diproses, program mencetak total biaya parkir kepada pengguna.

### Latihan3

Buatlah program dengan bahasa Go yang digunakan untuk mengidentifikasi pola aritmatika berdasarkan bilangan yang diinputkan dan melakukan operasi matematika yang sesuai. Beberapa ketentuan kategori diantaranya:

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

**Masukan** terdiri dari satu bilangan bulat.

**Keluaran** berupa teks "Kategori: Bilangan Ganjil" dilanjutkan dengan "Hasil penjumlahan dengan bilangan berikutnya  $\%(\text{input}) + \%(\text{input}+1) = \%(\text{hasil})$ " untuk Bilangan Ganjil. "Kategori: Bilangan Genap" dilanjutkan dengan "Hasil perkalian dengan bilangan berikutnya  $\%(\text{input}) * \%(\text{input}+1) = \%(\text{hasil})$ " untuk Bilangan Genap. "Kategori: Bilangan Kelipatan 5" dilanjutkan dengan "Hasil kuadrat dari  $\%(\text{input})^2 = \%(\text{hasil})$ " untuk Bilangan Kelipatan 5. "Kategori: Bilangan Kelipatan 10" dilanjutkan dengan "Hasil pembagian antara  $\%(\text{input}) / 10 = \%(\text{hasil})$ " untuk Bilangan Kelipatan 10. Keluaran sesuai dengan masukan yang diberikan.

**Contoh masukan dan keluaran:**

No	Masukan	Keluaran
1	5	Kategori: Bilangan Ganjil Hasil penjumlahan dengan bilangan berikutnya $5 + 6 = 11$
2	8	Kategori: Bilangan Genap Hasil perkalian dengan bilangan berikutnya $8 * 9 = 72$
3	25	Kategori: Bilangan Kelipatan 5 Hasil kuadrat dari $25^2 = 625$
4	20	Kategori: Bilangan Kelipatan 10 Hasil pembagian antara $20 / 10 = 2$

Source Code:

```
package main
import "fmt"
func main() {
    var input int
```

```

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

```

### Output:

```

PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol3.go"
5
Kategori: Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol3.go"
8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol3.go"
25
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 25 ^2 = 625
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol3.go"
20
Kategori: Bilangan Kelipatan 10
Hasil pembagian antara 20 / 10 = 2
PS C:\Users\User\Documents\MODUL 11> go run "c:\Users\User\Documents\MODUL 11\LatihanSoal\Latsol3.go"

```

### Deskripsi Program:

Program Go ini meminta pengguna untuk memasukkan sebuah bilangan bulat. Berdasarkan nilai yang diinputkan, program akan mengidentifikasi kategori bilangan tersebut dan melakukan operasi matematika yang sesuai:

- Bilangan Kelipatan 10: Jika bilangan dapat dibagi habis oleh 10, program akan membagi bilangan tersebut dengan 10 dan menampilkan hasilnya.
- Bilangan Kelipatan 5: Jika bilangan dapat dibagi habis oleh 5 (kecuali 5 itu sendiri), program akan menghitung kuadrat dari bilangan tersebut.
- Bilangan Genap: Jika bilangan genap (tidak dapat dibagi habis oleh 2), program akan mengalikan bilangan tersebut dengan bilangan berikutnya.
- Bilangan Ganjil: Jika bilangan ganjil (tidak dapat dibagi habis oleh 2), program akan menjumlahkan bilangan tersebut dengan bilangan berikutnya.



## **DAFTAR PUSTAKA**

School of Computing. *Modul Praktikum 11 – Switch-Case Algoritma dan Pemrograman 1 SI Informatika*.2024