

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 1
“TIPE DATA & VARIABEL”



DISUSUN OLEH:
MUHAMMAD ZAKY MUBAROK
103112400073 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

Switch-Case

Switch-case adalah alternatif dari **else-if** yang digunakan untuk mempermudah pembacaan dan pengelolaan kode ketika ada banyak kondisi berbeda yang didasarkan pada nilai tertentu, bukan kondisi boolean. Dalam **switch-case**, pemilihan aksi dilakukan berdasarkan nilai dari ekspresi, bukan kondisi logis.

Komponen Switch-Case:

1. **Ekspresi:** Operasi yang menghasilkan nilai selain boolean.
2. **Nilai:** Pilihan case dari ekspresi yang ditentukan pada instruksi **switch-case**.
3. **Aksi:** Instruksi yang akan dieksekusi sesuai dengan nilai yang dihasilkan dari ekspresi. Jika tidak ada pilihan yang sesuai, aksi lain akan dieksekusi (default).

Switch-Case lebih efisien dan lebih mudah dibaca ketika ada banyak kondisi yang memeriksa nilai yang berbeda dari ekspresi yang sama

CONTOH SOAL

1. Latihan1

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:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR

PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\coso1\coso1.go"
13
1 PM
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\coso1\coso1.go"
0
12 AM
PS D:\Belajar Html\Modul11> 
```

Deskripsi Program:

- 1. Deklarasi Variabel:** Program mendeklarasikan tiga variabel:
 - jam12 untuk menyimpan jam dalam format 12 jam.
 - jam24 untuk menerima input jam dalam format 24 jam.
 - label untuk menyimpan label "AM" atau "PM".
- 2. Input:** Program membaca input dan menyimpannya dalam variabel jam24.
- 3. Switch:** Program menggunakan struktur kontrol *switch* untuk menentukan nilai dari jam12 dan label berdasarkan nilai jam24:
 - Case 1: Jika jam24 bernilai 0, berarti tengah malam (12 AM). Variabel jam12 diatur ke 12 dan label diatur ke "AM".
 - Case 2: Jika jam24 kurang dari 12, maka nilai jam12 sama dengan jam24 dan label diatur ke "AM" (misalnya, jam 8 pagi menjadi 8 AM).
 - Case 3: Jika jam24 bernilai 12, berarti tengah hari (12 PM). Variabel jam12 diatur ke 12 dan label diatur ke "PM".
 - Case 4: Jika jam24 lebih dari 12, maka nilai jam12 adalah jam24 - 12 dan label diatur ke "PM" (misalnya, jam 18 atau 6 sore menjadi 6 PM).
- 4. Output:** Program mencetak hasil konversi waktu dalam format 12 jam beserta labelnya.

Contoh penggunaan:

- Jika memasukkan 15 sebagai input, hasilnya akan menjadi 3 PM.
- Jika memasukkan 0 sebagai input, hasilnya akan menjadi 12 AM.

CONTOH SOAL

2. Latihan2

Source Code:

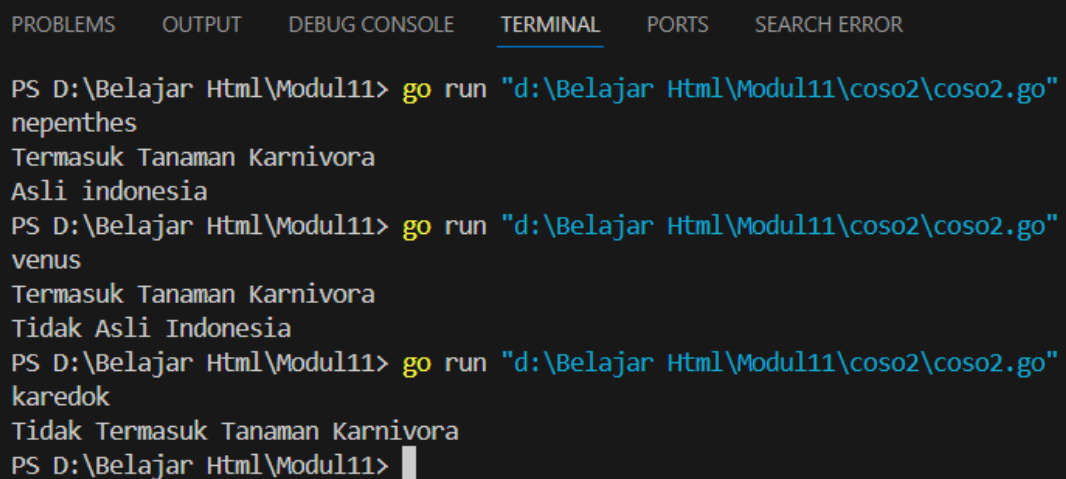
```
package main

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")
    }
}
```

Output:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR

PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\coso2\coso2.go"
nepenthes
Termasuk Tanaman Karnivora
Asli indonesia
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\coso2\coso2.go"
venus
Termasuk Tanaman Karnivora
Tidak Asli Indonesia
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\coso2\coso2.go"
karedok
Tidak Termasuk Tanaman Karnivora
PS D:\Belajar Html\Modul11> █
```

Deskripsi Program:

Deklarasi Variabel: Program mendeklarasikan satu variabel `nama_tanaman` yang bertipe string untuk menyimpan nama tanaman yang diinput.

Input: Program menggunakan `fmt.Scan(&nama_tanaman)` untuk membaca input dan menyimpannya dalam variabel `nama_tanaman`.

Switch: Program menggunakan struktur kontrol *switch* untuk memeriksa nilai dari `nama_tanaman` dan menentukan aksi yang sesuai:

- **Case "nepenthes" atau "drosera":** Jika `nama_tanaman` adalah "nepenthes" atau "drosera", program mencetak "Termasuk Tanaman Karnivora" dan "Asli Indonesia".
- **Case "venus" atau "sarracenia":** Jika `nama_tanaman` adalah "venus" atau "sarracenia", program mencetak "Termasuk Tanaman Karnivora" dan "Tidak Asli Indonesia".
- **Default:** Jika `nama_tanaman` bukan salah satu dari nilai yang ditentukan di atas, program mencetak "Tidak Termasuk Tanaman Karnivora".

CONTOH SOAL

3. Latihan3

Source Code:

```
package main

import "fmt"

func main() {
    var tipe_kendaraan string
    var durasi, tarif int

    fmt.Print("Masukkan Jenis Kendaraan (Motor/Mobil/Truk) : ")
    fmt.Scan(&tipe_kendaraan)
    fmt.Print("Masukkan Durasi Parkir (jam) : ")
    fmt.Scan(&durasi)

    switch {
        case tipe_kendaraan == "Motor" && durasi >= 1 && durasi <= 2 :
            tarif = 7000
        case tipe_kendaraan == "Motor" && durasi > 2 :
            tarif = 9000
        case tipe_kendaraan == "Mobil" && durasi >= 1 && durasi <= 2 :
            tarif = 15000
        case tipe_kendaraan == "Mobil" && durasi > 2 :
            tarif = 20000
        case tipe_kendaraan == "Truk" && durasi >= 1 && durasi <= 2 :
            tarif = 25000
        case tipe_kendaraan == "Truk" && durasi > 2 :
            tarif = 35000
        default:
            fmt.Println("Jenis kendaraan atau Durasi Parkir tidak Valid")
    }
    fmt.Printf("Tarif Parkir : Rp %d", tarif)
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR
Masukkan Jenis Kendaraan (Motor/Mobil/Truk) : Motor
Masukkan Durasi Parkir (jam) : 2
Tarif Parkir : Rp 7000
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\coso3\coso3.go"
Masukkan Jenis Kendaraan (Motor/Mobil/Truk) : Mobil
Masukkan Durasi Parkir (jam) : 4
Tarif Parkir : Rp 20000
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\coso3\coso3.go"
Masukkan Jenis Kendaraan (Motor/Mobil/Truk) : Sepeda
Masukkan Durasi Parkir (jam) : 2
Jenis kendaraan atau Durasi Parkir tidak Valid
Tarif Parkir : Rp 0
PS D:\Belajar Html\Modul11> 
```

Deskripsi Program:

Deklarasi Variabel: Program mendeklarasikan tiga variabel:

- **tipe_kendaraan (string):** Menyimpan jenis kendaraan yang dimasukkan.
- **durasi (int):** Menyimpan durasi parkir dalam jam yang dimasukkan.
- **tarif (int):** Menyimpan tarif parkir yang akan dihitung berdasarkan jenis kendaraan dan durasi parkir.

Input: Program meminta untuk memasukkan jenis kendaraan (Motor/Mobil/Truk) dan durasi parkir dalam jam. Input disimpan dalam variabel `tipe_kendaraan` dan `durasi`.

Switch: Program menggunakan struktur kontrol *switch* untuk menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parkir:

- **Motor:**
 - Jika durasi antara 1 dan 2 jam (termasuk), tarifnya adalah 7000.
 - Jika durasi lebih dari 2 jam, tarifnya adalah 9000.
- **Mobil:**
 - Jika durasi antara 1 dan 2 jam (termasuk), tarifnya adalah 15000.
 - Jika durasi lebih dari 2 jam, tarifnya adalah 20000.
- **Truk:**
 - Jika durasi antara 1 dan 2 jam (termasuk), tarifnya adalah 25000.
 - Jika durasi lebih dari 2 jam, tarifnya adalah 35000.
- **Default:**

- **Jika jenis kendaraan atau durasi parkir tidak valid, program mencetak pesan "Jenis kendaraan atau Durasi Parkir tidak Valid".**

Output: Program mencetak tarif parkir yang telah dihitung berdasarkan input

SOAL LATIHAN

Statement perulangan

1.

Source Code:

```
package main

import "fmt"

func main() {
    var ph float64
    var keluaran string
    fmt.Scan(&ph)

    switch {
    case ph >= 0 && ph < 6.5:
        keluaran = "Air tidak layak minum"
    case ph > 8.6 && ph <= 14:
        keluaran = "Air tidak layak minum"
    case ph >= 6.5 && ph <= 8.6:
        keluaran = "Air layak minum"
    default:
        keluaran = "Nilai pH tidak valid. Nilai pH harus antara 0 dan 14"
    }
    fmt.Println(keluaran)
}
```

Output:



```
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol1\latsol1.go"
8.6
Air layak minum
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol1\latsol1.go"
9
Air tidak layak minum
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol1\latsol1.go"
16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14
PS D:\Belajar Html\Modul11>
```

Deskripsikan Program :

Deklarasi Variabel: Program mendeklarasikan dua variabel:

- **ph (float64):** Menyimpan nilai pH yang diinput.
- **keluaran (string):** Menyimpan hasil evaluasi kelayakan air minum berdasarkan nilai pH.

Input: Program menggunakan `fmt.Scan(&ph)` untuk membaca input dan menyimpannya dalam variabel `ph`.

Switch: Program menggunakan struktur kontrol *switch* untuk menentukan hasil evaluasi berdasarkan nilai pH:

- **Case 1:** Jika `ph >= 0` dan `< 6.5`, maka keluaran diatur ke "Air tidak layak minum".
- **Case 2:** Jika `ph > 8.6` dan `<= 14`, maka keluaran diatur ke "Air tidak layak minum".
- **Case 3:** Jika `ph >= 6.5` dan `<= 8.6`, maka keluaran diatur ke "Air layak minum".
- **Default:** Jika `ph` tidak berada dalam rentang 0 hingga 14, keluaran diatur ke "Nilai pH tidak valid. Nilai pH harus antara 0 dan 14".

Output: Program mencetak hasil evaluasi kelayakan air minum berdasarkan nilai pH

SOAL LATIHAN

2. Latihan2 Source

Code:

```
package main

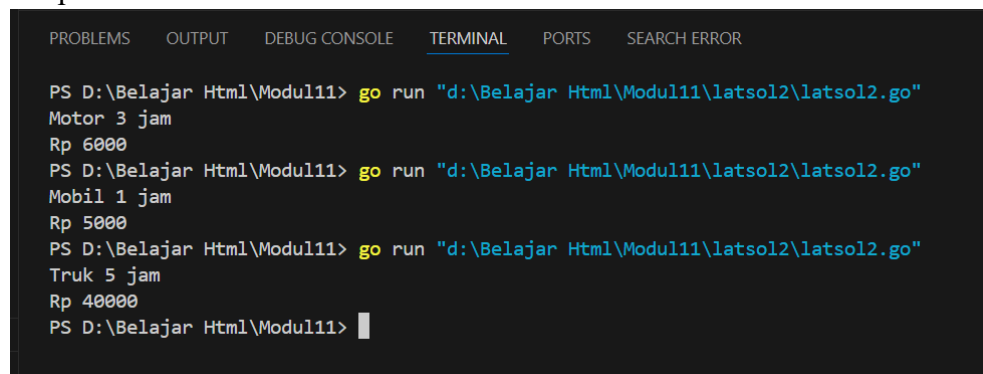
import "fmt"

func main() {
    var tipe string
    var jam, tarif int

    fmt.Scan(&tipe, &jam)

    switch {
    case tipe == "Motor" && jam == 1 && jam < 2:
        tarif = 2000
    case tipe == "Motor" && jam >= 2:
        tarif = jam * 2000
    case tipe == "Mobil" && jam == 1 && jam < 2:
        tarif = 5000
    case tipe == "Mobil" && jam >= 2:
        tarif = jam * 5000
    case tipe == "Truk" && jam == 1 && jam <= 2:
        tarif = 8000
    case tipe == "Truk" && jam >= 2:
        tarif = jam * 8000
    }
    fmt.Print("Rp ", tarif)
}
```

Output:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR

PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol2\latsol2.go"
Motor 3 jam
Rp 6000
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol2\latsol2.go"
Mobil 1 jam
Rp 5000
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol2\latsol2.go"
Truk 5 jam
Rp 40000
PS D:\Belajar Html\Modul11> 
```

Deskripsi Program:

Deskripsi Program:

Deklarasi Variabel: Program mendeklarasikan dua variabel:

tipe (string): Menyimpan jenis kendaraan yang diinput.

jam (int): Menyimpan durasi parkir dalam jam yang diinput.

tarif (int): Menyimpan tarif parkir yang akan dihitung berdasarkan jenis kendaraan dan durasi parkir.

Input: Program menggunakan `fmt.Scan(&tipe, &jam)` untuk membaca input dan menyimpannya dalam variabel `tipe` dan `jam`.

Switch: Program menggunakan struktur kontrol *switch* untuk menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parkir:

- **Motor:**
 - Jika jam bernilai 1 dan kurang dari 2, maka tarif diatur ke 2000.
 - Jika jam lebih dari atau sama dengan 2, maka tarif diatur ke $\text{jam} * 2000$.
- **Mobil:**
 - Jika jam bernilai 1 dan kurang dari 2, maka tarif diatur ke 5000.
 - Jika jam lebih dari atau sama dengan 2, maka tarif diatur ke $\text{jam} * 5000$.
- **Truk:**
 - Jika jam bernilai 1 dan kurang dari atau sama dengan 2, maka tarif diatur ke 8000.
 - Jika jam lebih dari atau sama dengan 2, maka tarif diatur ke $\text{jam} * 8000$.

Output: Program mencetak tarif parkir yang telah dihitung berdasarkan input dengan format `Rp {tarif}`.

SOAL LATIHAN

Statement perulangan

3. Latihan3

Source Code:

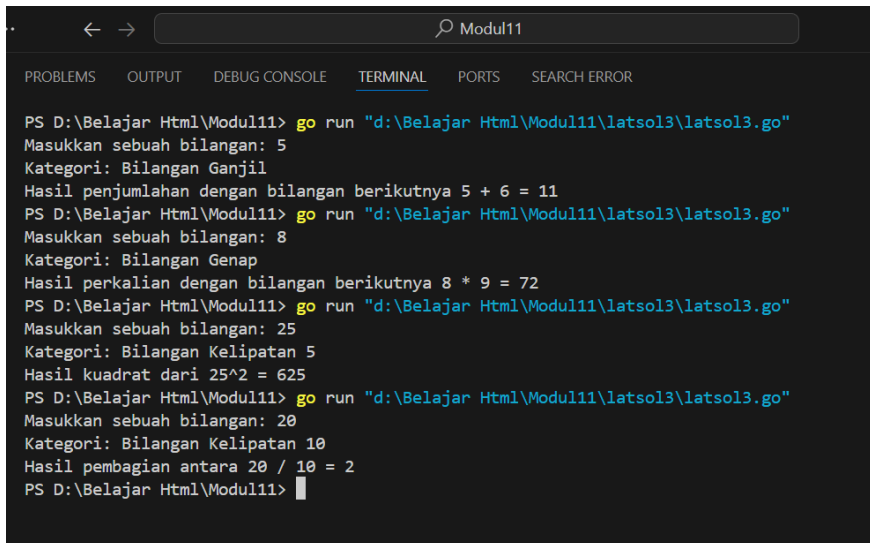
```
package main

import "fmt"

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

    switch {
    case bilangan%10 == 0:
        hasil := bilangan / 10
        fmt.Printf("Kategori: Bilangan Kelipatan 10\nHasil pembagian antara %d / 10 = %d\n", bilangan, hasil)
    case bilangan%5 == 0 && bilangan > 5:
        hasil := bilangan * bilangan
        fmt.Printf("Kategori: Bilangan Kelipatan 5\nHasil kuadrat dari %d^2 = %d\n", bilangan, hasil)
    case bilangan%2 == 0:
        hasil := bilangan * (bilangan + 1)
        fmt.Printf("Kategori: Bilangan Genap\nHasil perkalian dengan bilangan berikutnya %d * %d = %d\n", bilangan, bilangan+1, hasil)
    case bilangan%2 != 0:
        hasil := bilangan + (bilangan + 1)
        fmt.Printf("Kategori: Bilangan Ganjil\nHasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", bilangan, bilangan+1, hasil)
    default:
        fmt.Println("Nilai tidak valid.")
    }
}
```

Output:



```
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol3\latsol3.go"
Masukkan sebuah bilangan: 5
Kategori: Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol3\latsol3.go"
Masukkan sebuah bilangan: 8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol3\latsol3.go"
Masukkan sebuah bilangan: 25
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 25^2 = 625
PS D:\Belajar Html\Modul11> go run "d:\Belajar Html\Modul11\latsol3\latsol3.go"
Masukkan sebuah bilangan: 20
Kategori: Bilangan Kelipatan 10
Hasil pembagian antara 20 / 10 = 2
PS D:\Belajar Html\Modul11>
```

Deskripsi Program:

1. Deklarasi dan Input Bilangan:

- Program dimulai dengan mendeklarasikan variabel bilangan untuk menyimpan input.
- Menggunakan `fmt.Print` untuk menampilkan pesan kepada pengguna agar mereka memasukkan sebuah bilangan.
- `fmt.Scan(&bilangan)` digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel bilangan.

2. Pengkategorian Bilangan dan Operasi:

- Program menggunakan pernyataan `switch` untuk mengkategorikan bilangan yang dimasukkan pengguna berdasarkan beberapa kondisi.
- Kasus 1: `bilangan % 10 == 0`
 - Jika bilangan adalah kelipatan 10, hasilnya adalah pembagian bilangan dengan 10.
 - Menampilkan kategori dan hasil pembagian.
- Kasus 2: `bilangan % 5 == 0 && bilangan > 5`
 - Jika bilangan adalah kelipatan 5 dan lebih besar dari 5, hasilnya adalah kuadrat dari bilangan.
 - Menampilkan kategori dan hasil kuadrat.

- Kasus 3: $\text{bilangan} \% 2 == 0$
 - Jika bilangan adalah bilangan genap, hasilnya adalah perkalian bilangan dengan bilangan berikutnya.
 - Menampilkan kategori dan hasil perkalian.
- Kasus 4: $\text{bilangan} \% 2 != 0$
 - Jika bilangan adalah bilangan ganjil, hasilnya adalah penjumlahan bilangan dengan bilangan berikutnya.
 - Menampilkan kategori dan hasil penjumlahan.
- Kasus Default:
 - Jika bilangan tidak memenuhi salah satu kondisi di atas, menampilkan pesan bahwa nilai tidak valid.

Contoh Penggunaan:

- Jika input adalah 10, outputnya akan menunjukkan bahwa bilangan tersebut adalah kelipatan 10 dan memberikan hasil pembagian $10 / 10 = 1$.
- Jika input adalah 25, outputnya akan menunjukkan bahwa bilangan tersebut adalah kelipatan 5 dan memberikan hasil kuadrat $25^2 = 625$.
- Jika input adalah 6, outputnya akan menunjukkan bahwa bilangan tersebut adalah bilangan genap dan memberikan hasil perkalian $6 * 7 = 42$.
- Jika input adalah 5, outputnya akan menunjukkan bahwa bilangan tersebut adalah bilangan ganjil dan memberikan hasil penjumlahan $5 + 6 = 11$.

DAFTAR PUSTAKA

MODUL PRAKTIKUM 11 - SWITCH-CASE