

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 11
SWITCH-CASE



Disusun oleh:

FIKRI LUQMAN MUKTABAR

109082500103

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 jam24, jam12 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

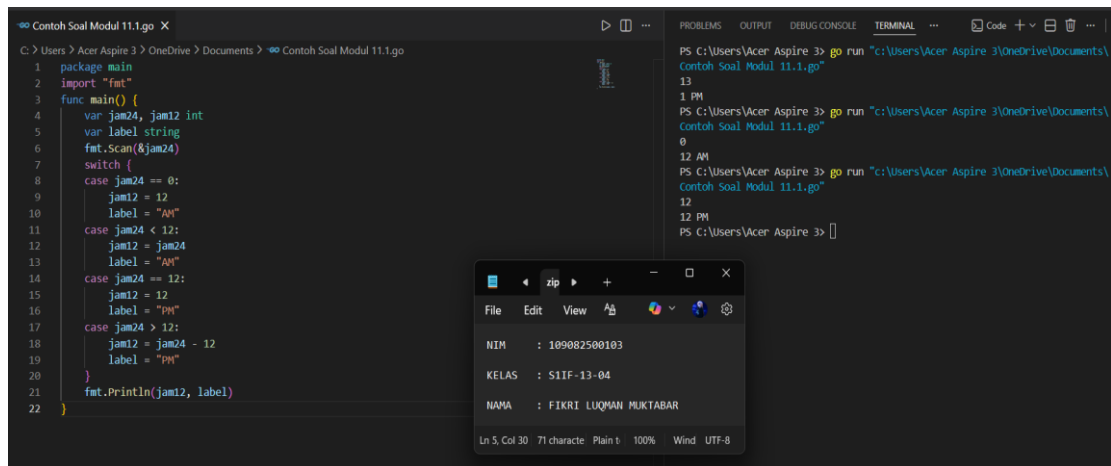
        label = "PM"

    }

    fmt.Println(jam12, label)

}
```

Screenshoot Program



```
1 package main
2 import "fmt"
3 func main() {
4     var jam24, jam12 int
5     var label string
6     fmt.Scan(&jam24)
7     switch {
8     case jam24 == 0:
9         jam12 = 12
10        label = "AM"
11    case jam24 < 12:
12        jam12 = jam24
13        label = "AM"
14    case jam24 == 12:
15        jam12 = 12
16        label = "PM"
17    case jam24 > 12:
18        jam12 = jam24 - 12
19        label = "PM"
20    }
21    fmt.Println(jam12, label)
22 }
```

```
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.1.go"
13
1 PM
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.1.go"
0
12 AM
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.1.go"
12
12 PM
PS C:\Users\Acer Aspire 3>
```

NIM : 109082500103
KELAS : SIIF-13-04
NAMA : FIKRI LUQMAN MUKTABAR

Deskripsi Program

Program ini digunakan untuk **mengonversi jam dalam format 24-jam (angka 0–23) menjadi format 12-jam (1–12) dan menambahkan label "AM" atau "PM"**. Berikut adalah penjelasannya; **package main** menandakan bahwa program ini merupakan program utama (bisa langsung dijalankan). **import "fmt"** mengimpor package fmt yang digunakan untuk input dan output, seperti membaca data (Scan) dan menampilkan hasil (Println). **func main() {...}** fungsi utama tempat eksekusi program dimulai. **var jam24, jam12 int** variabel jam24 menyimpan jam dalam format 24-jam yang dimasukkan pengguna (mis. 0...23). Sedangkan variabel jam12 menyimpan hasil konversi ke format 12-jam. **var label string** variabel label menyimpan "AM" atau "PM". **fmt.Scan(&jam24)** membaca input dan menyimpan nilainya ke dalam variabel 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" }** switch di sini tanpa ekspresi, sehingga setiap **case** adalah kondisi Boolean. Kemudian jam24 == 0 bermakna tengah malam sehingga akan menampilkan 12 AM. jam24 < 12 (bukan 0) artinya jam pagi biasa, misalnya 2: 2 AM. jam24 == 12 artinya siang hari pas: 12 PM. Jam24 > 12 berarti jam sore/malam, dikurangi 12 untuk format 12-jam, misalnya 14: 2 PM. **fmt.Println(jam12, label)** menampilkan hasil.

2. Guided 2

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 ")

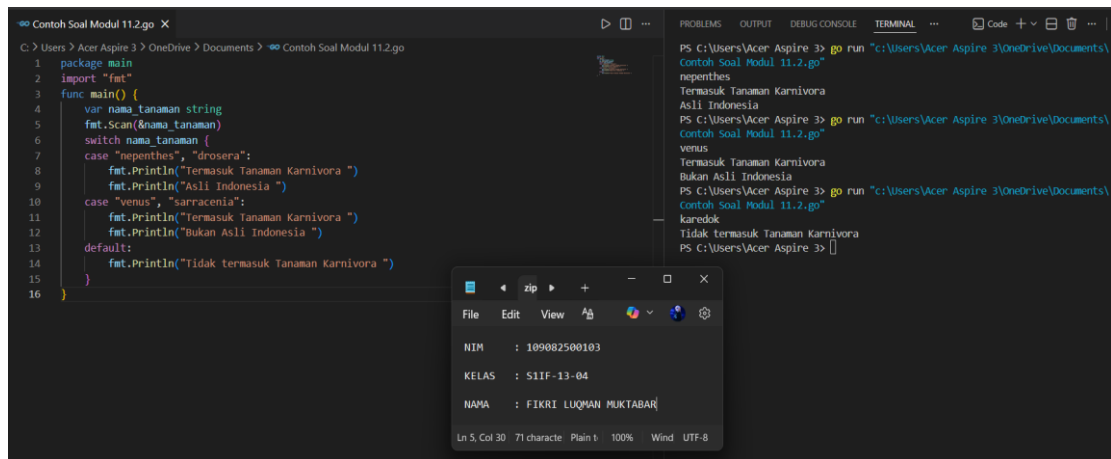
    default:

        fmt.Println("Tidak termasuk Tanaman
Karnivora ")

    }

}
```

Screenshoot Program



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

```
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.2.go"
nepenthes
Termasuk Tanaman Karnivora
Asli Indonesia
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.2.go"
venus
Termasuk Tanaman Karnivora
Bukan Asli Indonesia
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.2.go"
karedok
Tidak termasuk Tanaman Karnivora
PS C:\Users\Acer Aspire 3>
```

NIM : 109082500103
KELAS : SIIF-13-04
NAMA : FIKRI LUQMAN MUKTABAR

Deskripsi Program

Program ini digunakan untuk menerima inputan nama sebuah tanaman lalu menentukan apakah tanaman itu termasuk karnivora atau bukan dan apakah asli Indonesia atau bukan. Berikut adalah penjelasannya; **package main** menandakan bahwa program ini merupakan program utama (bisa langsung dijalankan). **import "fmt"** mengimpor package fmt yang digunakan untuk input dan output, seperti membaca data (Scan) dan menampilkan hasil (Println). **func main() {...}** fungsi utama tempat eksekusi program dimulai. **var nama_tanaman string** mendeklarasikan variabel nama_tanaman bertipe string untuk menyimpan input. **fmt.Scan(&nama_tanaman)** membaca input, dan menyimpan nilainya ke dalam variabel nama_tanaman. **switch nama_tanaman { ... }** struktur switch membandingkan nilai nama_tanaman dengan beberapa case. Pertama **case "nepenthes", "drosera":** Jika inputkan "nepenthes" atau "drosera", program akan mengeksekusi baris: **fmt.Println("Termasuk Tanaman Karnivora ")** **fmt.Println("Asli Indonesia ")**. Kedua **case "venus", "sarracenia":** Jika input "venus" atau "sarracenia", maka akan mengeksekusi baris: **fmt.Println("Termasuk Tanaman Karnivora ")** **fmt.Println("Bukan Asli Indonesia ")**. Yang terakhir **default:** Jika yang diinputkan bukan nama tanaman yang ada di case pertama maupun kedua dan hal selain nama tanaman, maka akan mengeksekusi baris: **fmt.Println("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.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 displays a Go IDE with the source code on the left and the terminal output on the right. The source code defines a program that calculates parking tariffs based on vehicle type and duration. The terminal shows the execution of the program with various inputs and the resulting output.

```

1 package main
2 import "fmt"
3 func main() {
4     var kendaraan string
5     var durasi int
6     var tarif int
7     fmt.Print("Masukkan jenis kendaraan (Motor/Mobil/Truk): ")
8     fmt.Scan(&kendaraan)
9     fmt.Print("Masukkan durasi parkir (dalam jam): ")
10    fmt.Scan(&durasi)
11    switch {
12    case kendaraan == "Motor" && durasi >= 1 && durasi <= 2:
13        |   tarif = 7000
14    case kendaraan == "Motor" && durasi > 2:
15        |   tarif = 9000
16    case kendaraan == "Mobil" && durasi >= 1 && durasi <= 2:
17        |   tarif = 15000
18    case kendaraan == "Mobil" && durasi > 2:
19        |   tarif = 20000
20    case kendaraan == "Truk" && durasi >= 1 && durasi <= 2:
21        |   tarif = 25000
22    case kendaraan == "Truk" && durasi > 2:
23        |   tarif = 35000
24    default:
25        |   fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")
26    }
27    fmt.Printf("Tarif Parkir: Rp %d\n", tarif)
28 }

```

```

PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 2
Tarif Parkir: Rp 7000
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Mobil
Masukkan durasi parkir (dalam jam): 4
Tarif Parkir: Rp 20000
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 3
Tarif Parkir: Rp 9000
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir (dalam jam): 1
Tarif parkir: Rp 25000
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Contoh Soal Modul 11.3.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\Acer Aspire 3>

```

Deskripsi Program

Program ini digunakan untuk **menerima inputan jenis kendaraan dan durasi parkir (dalam jam), lalu menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parkirnya.** Berikut adalah penjelasannya; **package main** menandakan bahwa program ini merupakan program utama (bisa langsung dijalankan). **import "fmt"** mengimpor package fmt yang digunakan untuk input dan output, seperti membaca data dan menampilkan hasil. **func main() {...}** fungsi utama tempat eksekusi program dimulai. **var kendaraan string** variabel kendaraan menyimpan nama jenis kendaraan. **var durasi int** variabel durasi untuk menyimpan lama parkir dalam jam dengan tipe integer. **var tarif int** variabel tarif akan diisi sesuai dengan aturan tarif (bertipe integer). **fmt.Print("Masukkan jenis kendaraan (Motor/Mobil/Truk): ")** **fmt.Scan(&kendaraan)** **fmt.Print("Masukkan durasi parkir (dalam jam): ")** **fmt.Scan(&durasi)** pada baris **fmt.Print**, program akan menampilkan teks: "Masukkan durasi parkir (dalam jam): " dan "Masukkan durasi parkir (dalam jam): " sekaligus meminta dua input berurutan, yaitu kendaraan dan durasi. Sedangkan pada **fmt.Scan**, program akan membaca input, dan menyimpan nilainya ke dalam variabel kendaraan dan 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")** } pada program ini tidak memakai ekspresi setelah **switch**, sehingga setiap **case** adalah kondisi Boolean. Lalu, kondisi memeriksa dua hal sekaligus: jenis kendaraan dan rentang durasi. Kemudian aturan tarif yang diterapkan, yaitu; untuk Motor: 1–2 jam → Rp 7.000 ; >2 jam → Rp 9.000, Mobil: 1–2 jam → Rp 15.000 ; >2 jam → Rp 20.000, dan Truk: 1–2 jam → Rp 25.000 ; >2 jam → Rp 35.000. Jika input tidak cocok dengan salah satu **case** (misalnya durasi 0 atau negatif dan kendaraan tidak salah satu dari tiga pilihan), maka akan diarahkan ke **default** yang nantinya akan menampilkan teks: “Jenis kendaraan atau durasi parkir tidak valid”. **fmt.Printf("Tarif Parkir: Rp %d\n", tarif)** menampilkan tarif yang telah ditentukan atau jika default terjadi (input tidak valid), tarif tetap bernilai 0 sehingga program akan tetap mencetak Tarif Parkir: Rp 0.

UNGUIDED

1. Unguided 1 Source Code

```
package main

import "fmt"

func main() {

    var pH float64

    fmt.Scan(&pH)

    switch {

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

        fmt.Print("Air layak minum")

    case (pH >= 0 && pH < 6.5) || (pH > 8.6 && pH <=
14):

        fmt.Print("Air tidak layak minum")

    default:

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

    }

}
```

Screenshoot Program

The screenshot shows a Go program in a code editor and its execution output in a terminal. The program is a simple pH checker. It prompts the user to enter a pH value. If the value is between 6.5 and 8.6, it prints 'Air layak minum'. If the value is outside this range but within 0 to 14, it prints 'Air tidak layak minum'. If the value is outside 0 to 14, it prints 'Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.'.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var pH float64
7     fmt.Scan(&pH)
8     switch {
9     case pH >= 6.5 && pH <= 8.6:
10        fmt.Print("Air layak minum")
11     case (pH >= 0 && pH < 6.5) || (pH > 8.6 && pH <= 14):
12        fmt.Print("Air tidak layak minum")
13     default:
14        fmt.Print("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
15     }
16 }
```

Terminal Output:

```
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Latihan Soal Modul 11.1.go"
8.6
Air layak minum
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Latihan Soal Modul 11.1.go"
9
Air tidak layak minum
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\
Latihan Soal Modul 11.1.go"
16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.
PS C:\Users\Acer Aspire 3>
```

Deskripsi Program

Program ini digunakan untuk **menentukan apakah kadar pH pada air yang diinputkan termasuk air yang layak untuk diminum atau tidak**. Berikut adalah penjelasannya; **package main** menandakan bahwa program ini merupakan program utama (bisa langsung dijalankan). **import "fmt"** mengimpor package fmt yang digunakan untuk input dan output, seperti membaca data dan menampilkan hasil. **func main() {...}** fungsi utama tempat eksekusi program dimulai. **var pH float64** Mendeklarasikan variabel pH bertipe float64. Variabel ini akan menyimpan nilai pH yang dimasukkan. **fmt.Scan(&pH)** membaca input, dan menyimpan nilainya ke dalam variabel pH. **switch {...}** switch disini tanpa ekspresi, sehingga setiap **case** berisi kondisi boolean. **case pH >= 6.5 && pH <= 8.6:** Kondisi pertama menggunakan operator && (AND), jadi kedua syarat harus terpenuhi. Jika kondisi ini benar, maka baris di bawahnya yaitu **fmt.Print("Air layak minum")** akan dieksekusi. Jika tidak maka akan memeriksa case selanjutnya, **case (pH >= 0 && pH < 6.5) || (pH > 8.6 && pH <= 14):** Kondisi kedua: dua sub-kondisi digabung dengan operator || (OR): Jika salah satu true maka akan mengeksekusi baris **fmt.Print("Air tidak layak minum")**. **default:** Dieksekusi jika tidak ada case yang terpenuhi. Karena case1 dan case2 terdiri dari rentang 0-14, maka **default** menangani nilai di luar 0-14 dan akan mengeksekusi **baris fmt.Print("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")**.

2. Unguided 2

Source Code

```
package main

import "fmt"

func main() {

    var jenis_kendaraan string

    var jumlah_jam, total_biaya int

    fmt.Scan(&jenis_kendaraan, &jumlah_jam)

    switch jenis_kendaraan {

    case "motor":

        total_biaya = jumlah_jam * 2000

    case "mobil":

        total_biaya = jumlah_jam * 5000

    case "truk":

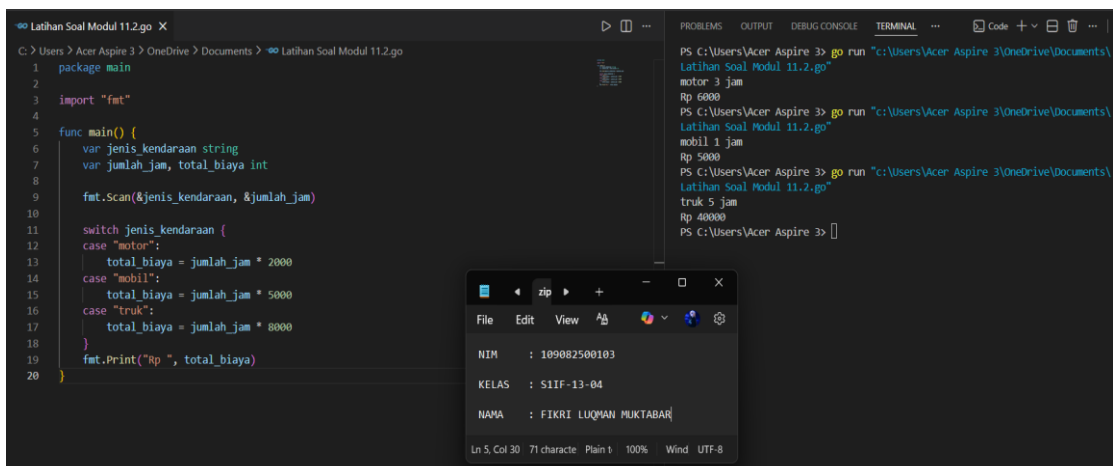
        total_biaya = jumlah_jam * 8000

    }

    fmt.Print("Rp ", total_biaya)

}
```

Screenshoot Program



```
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\Latihan Soal Modul 11.2.go"
motor 3 jam
Rp 6000
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\Latihan Soal Modul 11.2.go"
mobil 1 jam
Rp 5000
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\Latihan Soal Modul 11.2.go"
truk 5 jam
Rp 40000
PS C:\Users\Acer Aspire 3>
```

NIM : 109082500103
KELAS : SIIF-13-04
NAMA : FIKRI LUQMAN MUKTABAR

Ln 5, Col 30 71 character Plain text 100% Wind UTF-8

Deskripsi Program

Program ini digunakan untuk **menghitung tarif parkir berdasarkan 3 jenis kendaraan (motor, mobil, dan truk) dan durasi parkir (dalam jam) yang diinputkan**. Berikut adalah penjelasannya; **package main** menandakan bahwa program ini merupakan program utama (bisa langsung dijalankan). **import "fmt"** mengimpor package fmt yang digunakan untuk input dan output, seperti membaca data dan menampilkan hasil. **func main() {...}** fungsi utama tempat eksekusi program dimulai. **var jenis_kendaraan string** mendeklarasikan variabel **jenis_kendaraan** bertipe **string** dan juga menyimpan kata seperti "motor", "mobil", atau "truk" dari input. **var jumlah_jam, total_biaya int** mendeklarasikan dua variabel bertipe integer; yaitu variabel **jumlah_jam** untuk menyimpan durasi parkir dalam jam dan variabel **total_biaya** untuk menyimpan hasil perhitungan biaya. **fmt.Scan(&jenis_kendaraan, &jumlah_jam)** membaca input, dan menyimpan nilainya ke dalam variabel **jenis_kendaraan** dan **jumlah_jam**. **switch jenis_kendaraan {...}** switch ini membandingkan nilai **jenis_kendaraan** dengan setiap **case**. **case "motor": total_biaya = jumlah_jam * 2000** untuk menentukan total_biaya jika yang diinputkan "motor", maka tarif per jam yaitu 2000 dikali dengan input jumlah_jam-nya. **case "mobil": total_biaya = jumlah_jam * 5000** sama seperti **case** sebelumnya, untuk menentukan total_biaya jika diinputkan "mobil", maka tarif per jam yaitu 5000 dikali dengan input jumlah_jam-nya. **case "truk": total_biaya = jumlah_jam * 8000** sama seperti 2 **case** sebelumnya, untuk menentukan total_biaya jika diinputkan "truk", maka tarif per jam yaitu 8000 dikali dengan input jumlah_jam-nya. **fmt.Print("Rp ", total_biaya)** menampilkan teks: Rp beserta hasil perhitungannya(total biaya-nya).

3. Unguided 3

Source Code

```
package main

import "fmt"

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

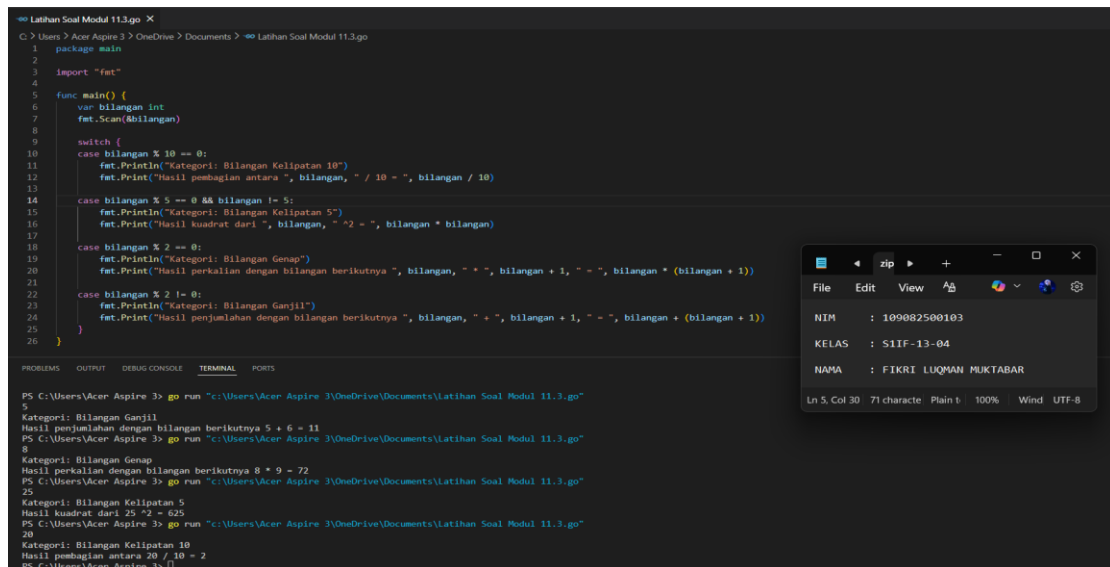
    switch {
    case bilangan % 10 == 0:
        fmt.Println("Kategori: Bilangan Kelipatan 10")
        fmt.Print("Hasil pembagian antara ", bilangan,
            " / 10 = ", bilangan / 10)

        case bilangan % 5 == 0 && bilangan != 5:
            fmt.Println("Kategori: Bilangan Kelipatan 5")
            fmt.Print("Hasil kuadrat dari ", bilangan, " ^2
                = ", bilangan * bilangan)

        case bilangan % 2 == 0:
            fmt.Println("Kategori: Bilangan Genap")
            fmt.Print("Hasil perkalian dengan bilangan
                berikutnya ", bilangan, " * ", bilangan + 1, " = ",
                bilangan * (bilangan + 1))

        case bilangan % 2 != 0:
            fmt.Println("Kategori: Bilangan Ganjil")
            fmt.Print("Hasil penjumlahan dengan bilangan
                berikutnya ", bilangan, " + ", bilangan + 1, " = ",
                bilangan + (bilangan + 1))
    }
}
```

Screenshoot Program



The screenshot shows a Go program in a text editor and its execution in a terminal. The program is a simple calculator that takes an integer input and categorizes it based on its divisibility by 10, 5, or 2. It also performs arithmetic operations based on the category.

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

The terminal output shows the results of running the program for different inputs:

```
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\Latihan Soal Modul 11.3.go"
5
Kategori: Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\Latihan Soal Modul 11.3.go"
8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\Latihan Soal Modul 11.3.go"
25
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 25 * 2 = 625
PS C:\Users\Acer Aspire 3> go run "c:\Users\Acer Aspire 3\OneDrive\Documents\Latihan Soal Modul 11.3.go"
20
Kategori: Bilangan Kelipatan 10
Hasil pembagian antara 20 / 10 = 2
PS C:\Users\Acer Aspire 3>
```

Deskripsi Program

Program ini digunakan untuk **menentukan kategori suatu bilangan yang diinputkan dan melakukan operasi aritmatika sesuai kategorinya**. Bilangan ganjil akan dijumlahkan dengan bilangan berikutnya, bilangan genap dikalikan dengan bilangan berikutnya, bilangan kelipatan 5 dihitung nilai kuadratnya, dan bilangan kelipatan 10 dibagi dengan 10. Berikut adalah penjelasannya; **package main** menandakan bahwa program ini merupakan program utama (bisa langsung dijalankan). **import "fmt"** mengimpor package fmt yang digunakan untuk input dan output, seperti membaca data dan menampilkan hasil. **func main() {...}** fungsi utama tempat eksekusi program dimulai. **var bilangan int** mendeklarasikan variabel bilangan dalam bentuk integer untuk menyimpan nilai input. **fmt.Scan(&bilangan)** membaca input, dan menyimpan nilainya ke dalam variabel bilangan. **switch {...}** switch disini tanpa ekspresi, sehingga setiap **case** berisi kondisi boolean. Karena **switch** hanya menjalankan **case** pertama yang benar, kita letakkan pemeriksaan kelipatan khusus (10 kemudian 5) **sebelum** pengecekan genap/ganjil agar kelipatan 10/5 terdeteksi lebih dulu. Jika genap/ganjil diletakkan dulu, nilai seperti 20 (genap dan kelipatan 10) akan dikategorikan sebagai genap sebelum sempat dicek kelipatan 10. **case bilangan % 10 == 0:** Jika case ini benar, maka 2 baris di bawahnya yaitu **fmt.Println("Kategori: Bilangan Kelipatan 10")** dan **fmt.Print("Hasil pembagian antara ", bilangan, " / 10 = ", bilangan / 10)** akan dieksekusi. Jika tidak maka akan

memeriksa **case** selanjutnya, **case bilangan % 5 == 0 && bilangan != 5**: Sama seperti case sebelumnya, jika case ini benar maka 2 baris di bawahnya yaitu **fmt.Println("Kategori: Bilangan Kelipatan 5")** dan **fmt.Print("Hasil kuadrat dari ", bilangan, " ^2 = ", bilangan * bilangan)** akan dieksekusi. Jika tidak maka akan memeriksa **case** selanjutnya, **case bilangan % 2 == 0**: jika case ini benar maka 2 baris di bawahnya yaitu **fmt.Println("Kategori: Bilangan Genap")** dan **fmt.Print("Hasil perkalian dengan bilangan berikutnya ", bilangan, " * ", bilangan + 1, " = ", bilangan * (bilangan + 1))** akan dieksekusi. Jika tidak maka akan memeriksa **case** selanjutnya, **case bilangan % 2 != 0**: jika case ini benar maka 2 baris di bawahnya yaitu **fmt.Println("Kategori: Bilangan Ganjil")** dan **fmt.Print("Hasil penjumlahan dengan bilangan berikutnya ", bilangan, " + ", bilangan + 1, " = ", bilangan + (bilangan + 1))** akan dieksekusi.