

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 11

SWITCH-CASE



**Telkom
University
PURWOKERTO**

Disusun oleh:

NAMA : DAYANA RISTA NUR FAUZIAH

NIM : 109082500195

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 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)
}
```

Screenshot program

The screenshot shows a Go code editor interface. On the left, the code for `guided1.go` is displayed:

```
∞ guided1.go > main
4     "fmt"
5 )
6
7 func main() {
8     var jam12, jam24 int
9     var label string
10    fmt.Scan(&jam24)
11    switch {
12        case jam24 == 0:
13            jam12 = 12
14            label = "AM"
15        case jam24 < 12:
16            jam12 = jam24
17            label = "AM"
18        case jam24 == 12:
19            jam12 = 12
20            label = "PM"
21        case jam24 > 12:
22            jam12 = jam24 - 12
23            label = "PM"
24    }
25    fmt.Println(jam12, label)
26 }
```

On the right, a terminal window shows the output of running the program:

```
nama : dayana rista nur fauziah
nim : 109082500195

Ln 2, Col 19 50 character Plain t 100% Wind UTF-8
```

Below the terminal, the command line shows the execution:

```
oograman\modul_11\guided1.go
13
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul_11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul_11\guided1.go"
0
12 AM
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul_11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul_11\guided1.go"
12
12 PM
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul_11>
```

Deskripsi program

Program meminta untuk ngubah jam dari format 24 jam ke format 12 jam. Pas program mulai, dia siapin tiga variabel: jam12 buat nyimpen hasil konversi, jam24 buat nyimpen jam yang akan di masukan, dan label buat nandain AM atau PM.

Setelah kamu masukin angka jam 0–23, program langsung ngecek kondisinya satu-satu lewat switch.

- Kalau jamnya 0, itu dianggap jam 12 AM, jadi jam12 diset 12 dan labelnya “AM”.
 - Kalau angka yang kamu masukin kurang dari 12, misalnya 5 atau 9, itu artinya masih pagi, jadi jamnya tetap sama dan labelnya AM.
 - Kalau jam yang kamu masukin 12, itu langsung berubah jadi 12 PM karena jam 12 siang.
 - Tapi kalau angka yang kamu masukin lebih dari 12, misalnya 13, 15, atau 22, program bakal ngurangin angka itu dengan 12 biar balik ke format jam 1–11 dan labelnya jadi PM.
- Di akhir, program tinggal nampilin hasilnya berupa jam versi 12 jam plus keterangan AM/PM. Jadi keseluruhan kodennya cuma ngecek jam yang kamu kasih, nyesuaiin aturnanya, terus nge-print hasilnya.

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

Screenshoot program

The screenshot shows a terminal window and a code editor side-by-side. The terminal window displays the output of a Go program. The code editor shows a Go file named 'guided02.go' with code that reads a name from input and checks if it's a carnivorous plant from Indonesia.

```
∞ guided02.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var nama_tanaman string
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 }
```

Terminal Output:

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\guided02.go"
nepenthes
Terkait dengan Tanaman Karnivora.
Asli Indonesia.
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\guided02.go"
venus
Terkait dengan Tanaman Karnivora.
Tidak Asli Indonesia.
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\guided02.go"
karedok
Tidak termasuk Tanaman Karnivora.
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11>
```

Deskripsi program

Program ini mulai dengan menyiapkan satu variabel bernama `nama_tanaman` untuk menampung tulisan yang nantinya dimasukkan. Setelah itu, program menunggu input—misalnya seseorang mengetik “`nepenthes`”. Begitu input diterima, program langsung masuk ke bagian `switch`, yaitu bagian yang dipakai untuk membandingkan nilai input dengan beberapa pilihan.

Di dalam `switch`, program mengecek apakah nama tanaman yang dimasukkan sama dengan “`nepenthes`” atau “`drosera`”.

- Jika cocok, program menampilkan dua baris teks yang menjelaskan bahwa tanaman tersebut termasuk tanaman karnivora dan berasal dari Indonesia.

- Jika tidak cocok, pengecekan lanjut ke pilihan berikutnya: apakah nama tersebut “`venus`” atau “`sarracenia`”.

- Jika cocok, program menampilkan bahwa tanaman itu karnivora tetapi tidak berasal dari Indonesia.

- Jika input tidak cocok dengan semua pilihan yang disebutkan, bagian `default` akan dijalankan, yaitu pernyataan bahwa tanaman tersebut bukan tanaman karnivora.

Setelah salah satu kondisi terpenuhi, program menampilkan hasilnya dan pekerjaan selesai.

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 shows a Go code editor interface. On the left, the code for `guided3.go` is displayed:

```
go guided3.go > main
5 func main() {
10     fmt.Scan(&kendaraan)
11     fmt.Print("Masukkan durasi parkir (dalam jam): ")
12     fmt.Scan(&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.Printf("Tarif Parkir: Rp %d\n", tarif)
30 }
```

On the right, a terminal window shows the execution of the program and its output:

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\guided3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Motor
Masukkan durasi parkir (dalam jam): 3
Tarif Parkir: Rp 9000
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\guided3.go"
Masukkan jenis kendaraan (Motor/Mobil/Truk): Truk
Masukkan durasi parkir (dalam jam): 1
Tarif Parkir: Rp 25000
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\guided3.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
```

Deskripsi program

Program ini bekerja dengan cara menyiapkan tiga variabel kendaraan untuk menampung jenis kendaraan, durasi untuk lama parkir, dan tarif untuk hasil akhir. Setelah itu, program meminta input jenis kendaraan dan durasi parkir dalam jam. Begitu dua data itu diterima, program masuk ke bagian switch yang berisi beberapa kondisi untuk menentukan tarif parkir. Kondisi-kondisi ini membandingkan apakah kendaraan yang dimasukkan adalah Motor, Mobil, atau Truk, lalu melihat apakah durasinya 1–2 jam atau lebih dari 2 jam. Jika cocok dengan salah satu kondisi, tarif diisi sesuai aturan: Motor 7000 atau 9000, Mobil 15000 atau 20000, dan Truk 25000 atau 35000. Jika data tidak sesuai aturan, bagian default akan muncul dan memberi tahu bahwa input tidak valid. Setelah tarif ditentukan, program langsung menampilkan jumlah tarif parkirnya.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var ph float64

    fmt.Print("Masukkan nilai pH: ")
    fmt.Scan(&ph)

    if ph < 0 || ph > 14 {
        fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0
dan 14.")
    } else if ph >= 6.5 && ph <= 8.6 {
        fmt.Println("Air layak minum")
    } else {
        fmt.Println("Air tidak layak minum")
    }
}
```

Screenshot program

The screenshot shows a code editor with a Go file named 'soal1.go'. The code prompts the user for a pH value, checks if it's between 0 and 14, and prints whether the water is drinkable or not. Below the editor is a terminal window showing the execution of the program and its output.

```
soal1.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var ph float64
7
8     fmt.Print("Masukkan nilai pH: ")
9     fmt.Scan(&ph)
10
11    if ph < 0 || ph > 14 {
12        fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")
13    } else if ph >= 6.5 && ph <= 8.6 {
14        fmt.Println("Air layak minum")
15    } else {
16        fmt.Println("Air tidak layak minum")
17    }
18 }

```

Terminal Output:

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal1.go"
Masukkan nilai pH: 8.6
Air layak minum
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal1.go"
Masukkan nilai pH: 9
Air tidak layak minum
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal1.go"
Masukkan nilai pH: 16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11>
```

Deskripsi program

Berikut penjelasan proses kerja kodennya dalam satu paragraf, bahasa santai, jelas, dan mudah dipahami:

Program ini mulai dengan menyiapkan variabel `ph` untuk menampung angka pH yang dimasukkan. Setelah itu, program meminta input nilai pH dan langsung memeriksanya. Jika angka yang dimasukkan kurang dari 0 atau lebih dari 14, program menampilkan bahwa nilai pH tidak valid karena pH hanya berada di rentang 0–14. Jika angkanya berada dalam rentang normal dan berada antara 6.5 sampai 8.6, program menyimpulkan bahwa air tersebut layak minum. Jika nilainya berada di luar rentang layak minum tetapi masih dalam batas 0–14, program menyatakan bahwa air tidak layak minum. Setelah pengecekan selesai, program menampilkan hasilnya dan proses pun selesai.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var jenis string
    var jam int

    fmt.Print("Jenis kendaraan (motor/mobil/truk) : ")
    fmt.Scan(&jenis)

    fmt.Print("Durasi parkir (jam) : ")
    fmt.Scan(&jam)

    if jam < 1 {
        jam = 1
    }

    tarif := 0

    switch jenis {
    case "motor":
        tarif = 2000 * jam
    case "mobil":
        tarif = 5000 * jam
    case "truk":
        tarif = 8000 * jam
    default:
```

```

        fmt.Println("Jenis kendaraan tidak valid.")

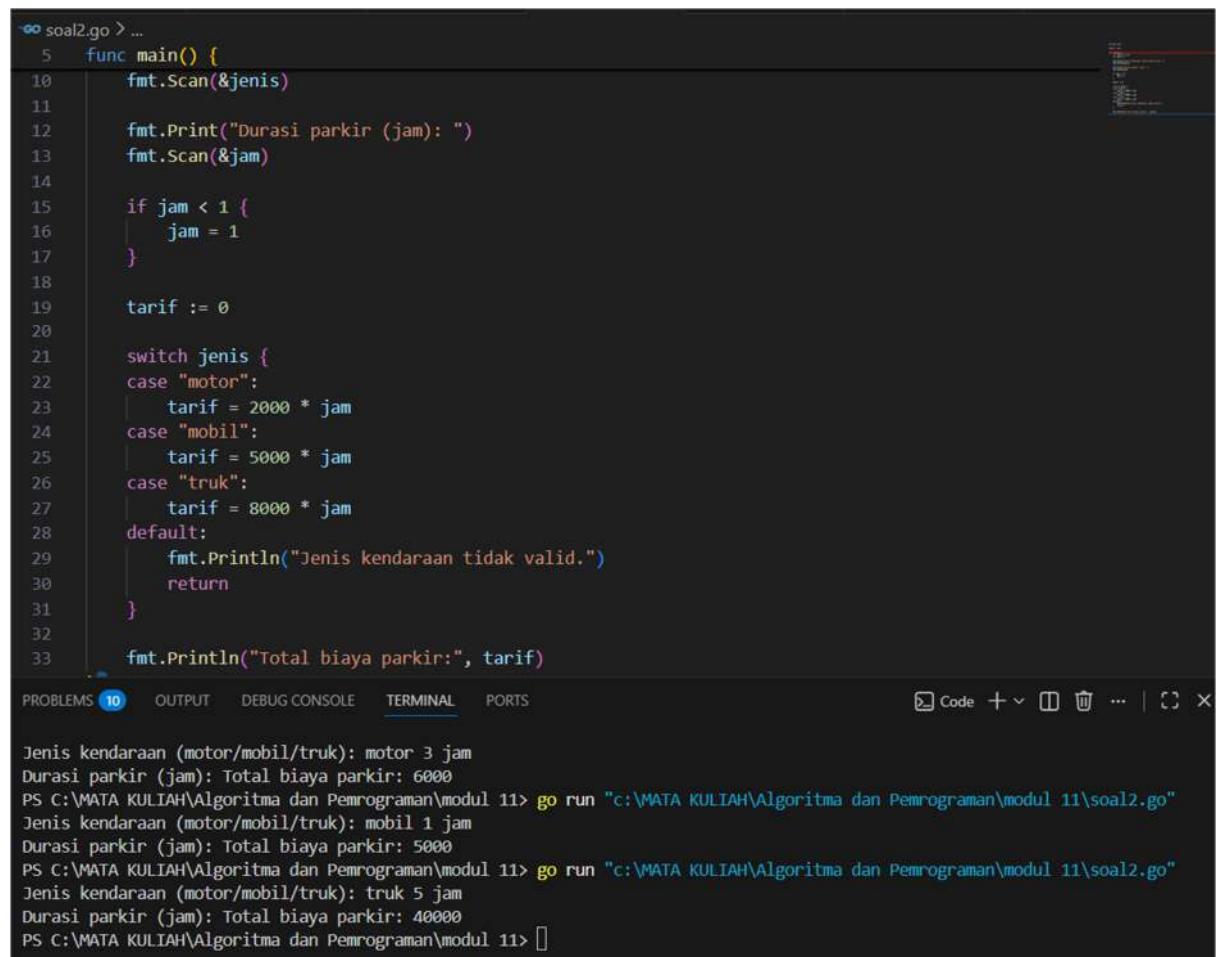
    return

}

fmt.Println("Total biaya parkir:", tarif)
}

```

Screenshot program



The screenshot shows a code editor interface with a terminal window below it. The code editor displays a Go program named 'soal2.go' with line numbers from 5 to 33. The terminal window shows the execution of the program and its output.

```

soal2.go > ...
5 func main() {
10     fmt.Scan(&jenis)
11
12     fmt.Print("Durasi parkir (jam): ")
13     fmt.Scan(&jam)
14
15     if jam < 1 {
16         jam = 1
17     }
18
19     tarif := 0
20
21     switch jenis {
22     case "motor":
23         tarif = 2000 * jam
24     case "mobil":
25         tarif = 5000 * jam
26     case "truk":
27         tarif = 8000 * jam
28     default:
29         fmt.Println("Jenis kendaraan tidak valid.")
30         return
31     }
32
33     fmt.Println("Total biaya parkir:", tarif)

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

Jenis kendaraan (motor/mobil/truk): motor 3 jam
Durasi parkir (jam): Total biaya parkir: 6000
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal2.go"
Jenis kendaraan (motor/mobil/truk): mobil 1 jam
Durasi parkir (jam): Total biaya parkir: 5000
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal2.go"
Jenis kendaraan (motor/mobil/truk): truk 5 jam
Durasi parkir (jam): Total biaya parkir: 40000
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11>

Deskripsi program

Program tersebut bekerja dengan cara meminta pengguna memasukkan jenis kendaraan (motor, mobil, atau truk) dan durasi parkir dalam jam. Setelah itu program memastikan bahwa durasi parkir minimal dihitung sebagai 1 jam jika input kurang dari 1. Selanjutnya program menentukan tarif per jam berdasarkan jenis kendaraan: motor dikenakan biaya 2000 per jam, mobil 5000 per jam, dan truk 8000 per jam. Jika jenis kendaraan tidak sesuai ketiga pilihan tersebut, program langsung menampilkan pesan bahwa jenis kendaraan tidak valid dan proses dihentikan. Setelah tarif per jam ditentukan, program mengalikan tarif tersebut dengan jumlah jam parkir untuk mendapatkan total biaya, lalu menampilkannya kepada pengguna.

3. Tugas 3

Source code

```
package main

import "fmt"

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

    if x%10 == 0 {
        fmt.Println("Kategori: Bilangan Kelipatan 10")
        fmt.Println("Hasil pembagian antara", x, "/ 10 =", x/10)

    } else if x == 5 {
        fmt.Println("Kategori: Bilangan Ganjil")
        fmt.Println("Hasil penjumlahan dengan bilangan berikutnya", x, "+", (x + 1), "=", x+(x+1))

    } else if x%5 == 0 {
        fmt.Println("Kategori: Bilangan Kelipatan 5")
        fmt.Println("Hasil kuadrat dari", x, "^ 2 =", x*x)

    } else if x%2 == 0 {
        fmt.Println("Kategori: Bilangan Genap")
        fmt.Println("Hasil perkalian dengan bilangan berikutnya", x, "*", (x + 1), "=", x*(x+1))

    } else {
        fmt.Println("Kategori: Bilangan Ganjil")
        fmt.Println("Hasil penjumlahan dengan bilangan berikutnya", x, "+", (x + 1), "=", x+(x+1))
    }
}
```

```
    }  
}
```

Screenshoot program

```
-∞ soal3.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x int
7     fmt.Println("Masukkan bilangan: ")
8     fmt.Scan(&x)
9
10    if x%10 == 0 {
11        fmt.Println("Kategori: Bilangan Kelipatan 10")
12        fmt.Println("Hasil pembagian antara", x, "/ 10 =", x/10)
13    } else if x == 5 {
14        fmt.Println("Kategori: Bilangan Ganjil")
15        fmt.Println("Hasil penjumlahan dengan bilangan berikutnya", x, "+", (x + 1), "=", x+(x+1))
16    } else if x%5 == 0 {
17        fmt.Println("Kategori: Bilangan Kelipatan 5")
18        fmt.Println("Hasil kuadrat dari", x, "^ 2 =", x*x)
19    } else if x%2 == 0 {
20        fmt.Println("Kategori: Bilangan Genap")
21        fmt.Println("Hasil perkalian dengan bilangan berikutnya", x, "*", (x + 1), "=", x*(x+1))
22    } else {
23        fmt.Println("Kategori: Bilangan Ganjil")
24        fmt.Println("Hasil penjumlahan dengan bilangan berikutnya", x, "+", (x + 1), "=", x+(x+1))
25    }
26}
27
28
29
30
31 }
```



```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal3.go"
Masukkan bilangan: 5
Kategori: Bilangan Ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal3.go"
Masukkan bilangan: 8
Kategori: Bilangan Genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal3.go"
Masukkan bilangan: 25
Kategori: Bilangan Kelipatan 5
Hasil kuadrat dari 25 ^ 2 = 625
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul 11\soal3.go"
Masukkan bilangan: 20
Kategori: Bilangan Kelipatan 10
Hasil pembagian antara 20 / 10 = 2
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul 11> █
```

Deskripsi program

Program ini meminta pengguna memasukkan sebuah bilangan lalu mengecek kategori bilangan tersebut menggunakan beberapa kondisi berurutan. Jika bilangan habis dibagi 10 maka dianggap sebagai kelipatan 10 dan program menampilkan hasil pembagian bilangan itu dengan 10. Jika bilangan sama dengan 5 maka dianggap sebagai bilangan ganjil khusus, dan program menjumlahkan bilangan tersebut dengan bilangan berikutnya. Jika bilangan tidak sama dengan 5 tetapi habis dibagi 5, maka bilangan tersebut adalah kelipatan 5 dan program menampilkan hasil kuadratnya. Jika bilangan tidak memenuhi kondisi sebelumnya tetapi

genap, maka program mengalikannya dengan bilangan berikutnya. Jika tidak termasuk semua kategori tersebut, bilangan dianggap ganjil dan program menampilkan hasil penjumlahahannya dengan bilangan setelahnya.