

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 11
SWITCH-CASE



Disusun oleh:

Panji Fauzan Habibullah Galang Sokya Whijaya

103112430162

S1IF-12-07

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

DASAR TEORI

Pseudocode

1. Pseudocode adalah notasi untuk menulis algoritma agar mudah dibaca dan dipahami, tetapi tidak mengikuti sintaksis dari bahasa pemrograman tertentu.
2. Program berguna memberikan identitas pada algoritma, membantu programmer mengorganisasi fungsionalitasnya.
3. Kamus berguna untuk mendeklarasikan data yang akan digunakan dan juga memudahkan programmer memahami tipe apa yang akan diproses. Contoh tipe data seperti integer, real, string, dan boolean. Di sini kita dapat menggunakan type, constant atau var.
4. Algoritma bertujuan untuk menjelaskan langkah-langkah yang harus dilakukan untuk mencapai tujuan algoritma. Ini mencakup input dan output.
5. Input berguna untuk memasukan data.
6. Structure/record digunakan untuk mengelompokkan beberapa variabel agar efisien
7. For digunakan untuk menjalankan kode secara berulang sesuai kondisi tertentu.
8. If digunakan untuk menentukan apakah algoritma layak untuk dilanjutkan jika syaratnya bersifat true. Else If digunakan jika algoritma if bersifat false tetapi memiliki syarat yang cukup untuk menjalankan Else If. Else digunakan ketika algoritma bersifat false pada If dan Else If ataupun ketika tidak ada Else If.
9. Switch-case adalah alternatif dari else-if, tetapi pemilihan tindakan didasarkan pada nilai tertentu atau operasi tipe data dasar yang tidak menghasilkan nilai boolean, berdasarkan kondisi.
10. Output berguna menyajikan hasil akhir dari algoritma akan disajikan kepada pengguna.
11. Endprogram bertujuan menghentikan algoritma.

Go Lang

1. Go (sering disebut sebagai Golang) adalah bahasa pemrograman yang dibuat oleh Google.
2. Setiap program Go dimulai dengan mendeklarasikan paket. Ini adalah cara untuk mengorganisir kode ke dalam modul-modul.
3. Import digunakan untuk mengimpor paket-paket ke dalam program.
4. Type, Constanta, dan Variabel memiliki tipe data seperti int, float32/64, string, dan bool.
5. Fmt.Scanln berguna untuk memasukan data.
6. Structure/record digunakan untuk mengelompokkan beberapa variabel agar efisien

7. For digunakan untuk menjalankan kode secara berulang sesuai kondisi tertentu.
8. If digunakan untuk menentukan apakah program layak untuk dilanjutkan jika syaratnya bersifat true. Else If digunakan jika program if bersifat false tetapi memiliki syarat yang cukup untuk menjalankan Else If. Else digunakan ketika program bersifat false pada If dan Else If ataupun ketika tidak ada Else If.
9. Switch-case adalah alternatif dari else-if, tetapi pemilihan tindakan didasarkan pada nilai tertentu atau operasi tipe data dasar yang tidak menghasilkan nilai boolean, berdasarkan kondisi.
10. Fmt.println berguna menyajikan hasil akhir dari kode program akan disajikan kepada pengguna.

TUGAS PENDAHULUAN

1. Tugas 1

Pseudo code

```
program KategoriUsia
kamus
    usia : integer
algoritma
    input(usia)
    depend on
        usia >= 0 and usia < 13:
            output("Anak-anak")
        usia >= 0 and usia < 20:
            output("Remaja")
        usia >= 0 and usia < 60:
            output("Dewasa")
        usia >= 60:
            output("Lansia")
        default:
            output("Umur tidak terdaftar dalam kategori")
    end
endprogram
```

Source code

```
package main

import "fmt"

func main() {
    var usia int
    fmt.Scan(&usia)
    switch {
```

```

case usia >= 0 && usia < 13:

    fmt.Print("Anak-anak")

    break

case usia >= 0 && usia < 20:

    fmt.Print("Remaja")

    break

case usia >= 0 && usia < 60:

    fmt.Print("Dewasa")

    break

case usia >= 60:

    fmt.Print("Lansia")

default:

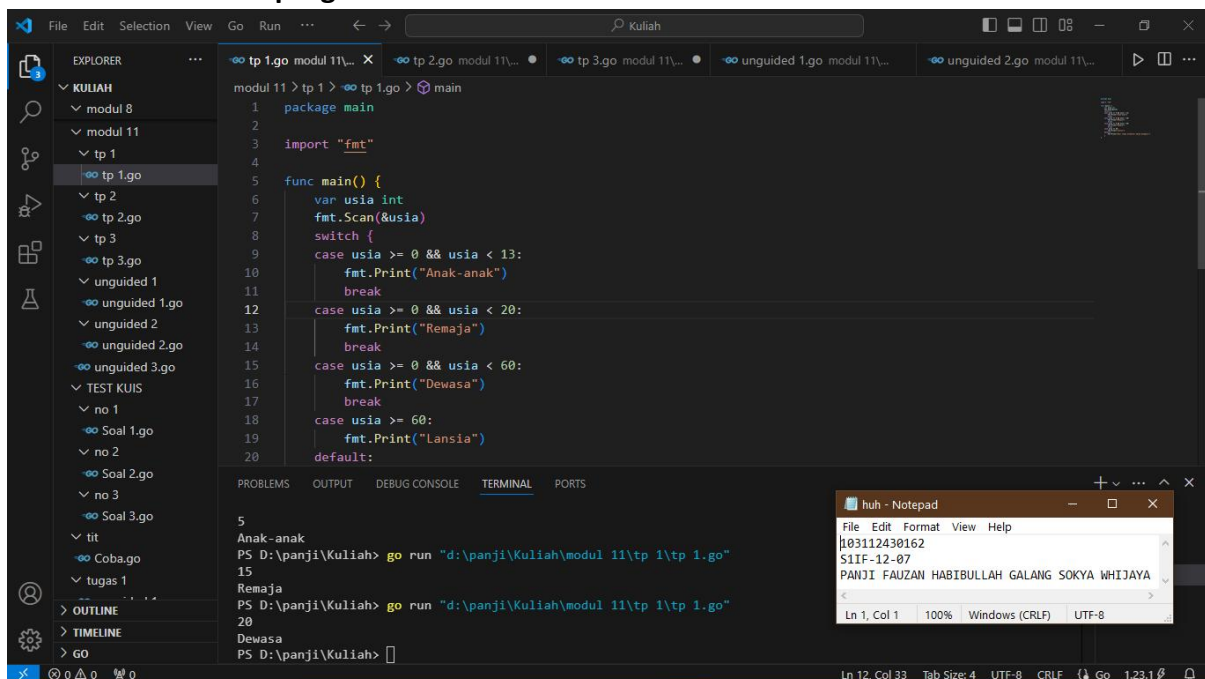
    fmt.Print("Umur tidak terdaftar dalam kategori")

}

}

```

Screenshoot program



Deskripsi program

- ◆ package main digunakan dalam Go Lang untuk menandai bahwa ini adalah program yang dapat dieksekusi.
- ◆ import "fmt" digunakan untuk operasi input/output seperti mencetak pesan ke layar dan mengambil input dari pengguna.
- ◆ func main() adalah titik masuk dari program. Semua perintah dalam program akan dieksekusi di dalam fungsi ini.
- ◆ Mendeklarasikan variabel usia dengan tipe integer untuk menyimpan nilai usia yang dimasukkan oleh pengguna.
- ◆ fmt.Scan(&usia) digunakan untuk membaca input usia dari pengguna dan menyimpannya dalam variabel usia.
- ◆ switch adalah struktur kontrol yang digunakan untuk memilih blok kode berdasarkan kondisi yang sesuai dengan nilai usia.
- ◆ case usia >= 0 && usia < 13 digunakan untuk mengevaluasi apakah nilai usia berada dalam rentang 0 hingga 12 tahun. Jika kondisi terpenuhi, program mencetak "Anak-anak".
- ◆ case usia >= 13 && usia < 20 digunakan untuk mengevaluasi apakah nilai usia berada dalam rentang 13 hingga 19 tahun. Jika kondisi terpenuhi, program mencetak "Remaja".
- ◆ case usia >= 20 && usia < 60 digunakan untuk mengevaluasi apakah nilai usia berada dalam rentang 20 hingga 59 tahun. Jika kondisi terpenuhi, program mencetak "Dewasa".
- ◆ case usia >= 60 digunakan untuk mengevaluasi apakah nilai usia adalah 60 tahun atau lebih. Jika kondisi terpenuhi, program mencetak "Lansia".
- ◆ default adalah kondisi fallback yang akan dijalankan jika tidak ada kondisi case yang terpenuhi. Dalam kasus ini, program mencetak "Umur tidak terdaftar dalam kategori".
- ◆ break digunakan untuk menghentikan eksekusi lebih lanjut dalam blok switch setelah satu kondisi terpenuhi.

2. Tugas 2

Pseudo code

```
program NamaHari
kamus
    hari : integer
algoritma
    input(hari)
    depend on
        hari = 1:
            output("Senin")
        hari = 2:
            output("Selasa")
        hari = 3:
```

```
        output("Rabu")

    hari = 4:
        output("Kamis")

    hari = 5:
        output("Jumat")

    hari = 6:
        output("Sabtu")

    hari = 7:
        output("Minggu")

    default:
        output("Bukan Hari")

end
endprogram
```

Source code

```
package main

import "fmt"

func main() {

    var hari int

    fmt.Scan(&hari)

    switch hari {

    case 1:

        fmt.Println("Senin")

    case 2:

        fmt.Println("Selasa")

    case 3:

        fmt.Println("Rabu")

    case 4:
```

```

        fmt.Println("Kamis")

    case 5:

        fmt.Println("Jumat")

    case 6:

        fmt.Println("Sabtu")

    case 7:

        fmt.Println("Minggu")

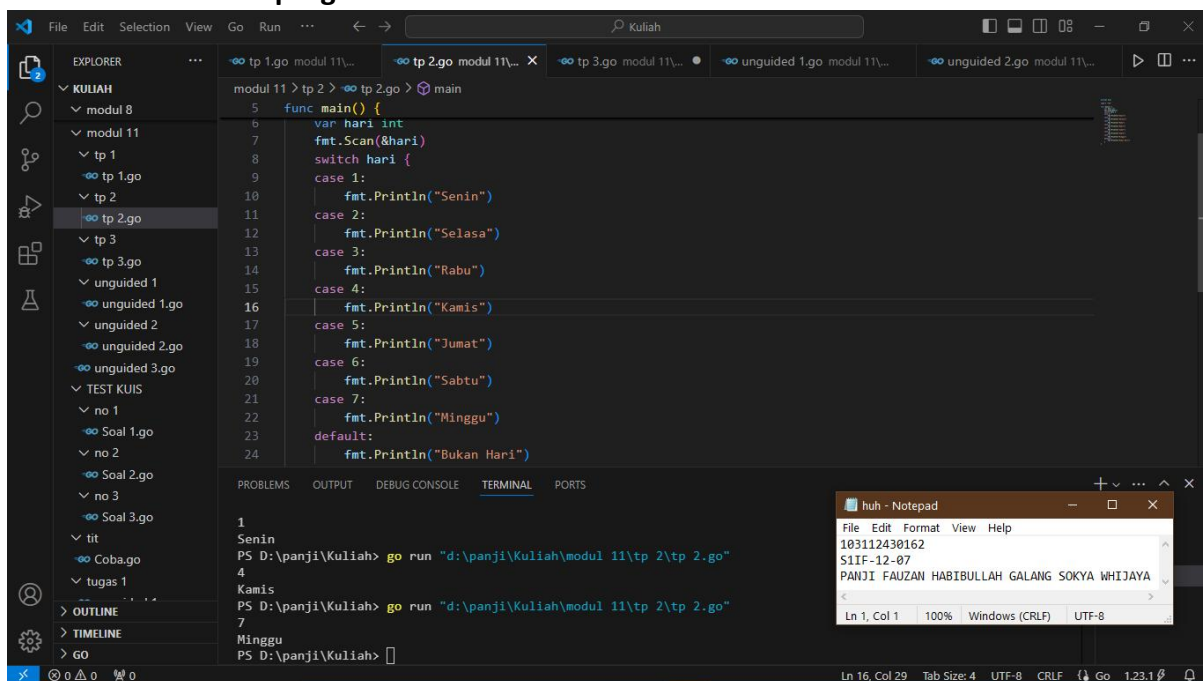
    default:

        fmt.Println("Bukan Hari")

}
}

```

Screenshoot program



Deskripsi program

- ◆ package main digunakan dalam Go Lang untuk menandai bahwa ini adalah program yang dapat dieksekusi.
- ◆ import "fmt" digunakan untuk operasi input/output seperti mencetak pesan ke layar dan mengambil input dari pengguna.
- ◆ func main() adalah titik masuk dari program. Semua perintah dalam program akan dieksekusi di dalam fungsi ini.

- ◆ Mendeklarasikan variabel hari dengan tipe integer untuk menyimpan nilai angka hari yang dimasukkan oleh pengguna.
- ◆ `fmt.Scan(&hari)` membaca input angka dari pengguna dan menyimpannya dalam variabel hari.
- ◆ switch hari adalah struktur kontrol untuk memilih satu dari beberapa blok kode berdasarkan nilai hari.
- ◆ case 1 mencetak "Senin" jika nilai hari adalah 1.
- ◆ case 2 mencetak "Selasa" jika nilai hari adalah 2.
- ◆ case 3 mencetak "Rabu" jika nilai hari adalah 3.
- ◆ case 4 mencetak "Kamis" jika nilai hari adalah 4.
- ◆ case 5 mencetak "Jumat" jika nilai hari adalah 5.
- ◆ case 6 mencetak "Sabtu" jika nilai hari adalah 6.
- ◆ case 7 mencetak "Minggu" jika nilai hari adalah 7.
- ◆ default digunakan sebagai blok fallback untuk menangani kasus ketika nilai hari tidak sesuai dengan angka 1 hingga 7. Program mencetak "Bukan Hari".

3. Tugas 3

Pseudo code

```

program KategoriRating
kamus
    rating, kpi : string
algoritma
    input(rating)
    depend on rating
        "G", "g":
            kpi = "Semua usia"
        "PG", "pg":
            kpi = "Anak-anak di atas 7 tahun"
        "PG-13", "pg-13":
            kpi = "Remaja di atas 13 tahun"
        "R", "r":
            kpi = "Dewasa"
    default:
        kpi = "Kode Tidak Valid"
    end
    output(kpi)
endprogram

```

Source code

```
package main

import "fmt"

func main() {

    var rating, kpi string

    fmt.Scan(&rating)

    switch rating {

    case "G", "g":

        kpi = "Semua usia"

    case "PG", "pg":

        kpi = "Anak-anak di atas 7 tahun"

    case "PG-13", "pg-13":

        kpi = "Remaja di atas 13 tahun"

    case "R", "r":

        kpi = "Dewasa"

    default:

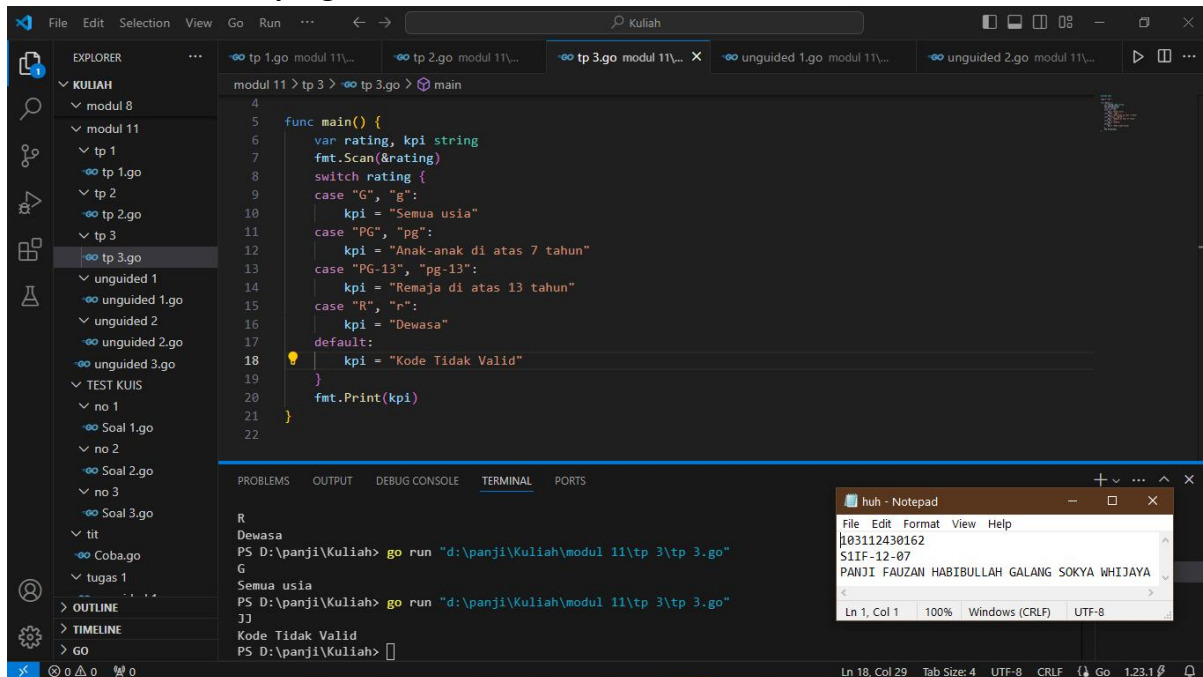
        kpi = "Kode Tidak Valid"

    }

    fmt.Print(kpi)

}
```

Screenshoot program



```
4
5 func main() {
6     var rating, kpi string
7     fmt.Scan(&rating)
8     switch rating {
9     case "G", "g":
10        kpi = "Semua usia"
11    case "PG", "pg":
12        kpi = "Anak-anak di atas 7 tahun"
13    case "PG-13", "pg-13":
14        kpi = "Remaja di atas 13 tahun"
15    case "R", "r":
16        kpi = "Dewasa"
17    default:
18        kpi = "Kode Tidak Valid"
19    }
20    fmt.Print(kpi)
21 }
22
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
R
Dewasa
PS D:\panji\Kuliah> go run "d:\panji\Kuliah\modul 11\tp 3\tp 3.go"
G
Semua usia
PS D:\panji\Kuliah> go run "d:\panji\Kuliah\modul 11\tp 3\tp 3.go"
JJ
Kode Tidak Valid
PS D:\panji\Kuliah>
```

Deskripsi program

- ◆ package main digunakan dalam Go Lang untuk menandai bahwa ini adalah program yang dapat dieksekusi.
- ◆ import "fmt" digunakan untuk operasi input/output seperti mencetak pesan ke layar dan mengambil input dari pengguna.
- ◆ func main() adalah titik masuk dari program. Semua perintah dalam program akan dieksekusi di dalam fungsi ini.
- ◆ Mendeklarasikan variabel rating dengan tipe string untuk menyimpan nilai kode rating yang dimasukkan oleh pengguna, dan variabel kpi dengan tipe string untuk menyimpan kategori usia yang sesuai berdasarkan rating.
- ◆ fmt.Scan(&rating) digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel rating.
- ◆ switch rating adalah struktur kontrol untuk memeriksa nilai yang dimasukkan dalam rating dan memilih tindakan berdasarkan nilai tersebut.
- ◆ case "G", "g" mencetak "Semua usia" jika rating adalah "G" atau "g".
- ◆ case "PG", "pg" mencetak "Anak-anak di atas 7 tahun" jika rating adalah "PG" atau "pg".
- ◆ case "PG-13", "pg-13" mencetak "Remaja di atas 13 tahun" jika rating adalah "PG-13" atau "pg-13".
- ◆ case "R", "r" mencetak "Dewasa" jika rating adalah "R" atau "r".
- ◆ default digunakan sebagai blok fallback yang mencetak "Kode Tidak Valid" jika rating yang dimasukkan tidak sesuai dengan salah satu pilihan yang valid.
- ◆ fmt.Print(kpi) digunakan untuk mencetak kategori usia yang sesuai dengan rating yang dimasukkan oleh pengguna, atau pesan "Kode Tidak Valid" jika input tidak valid.

LATIHAN MODUL

1. Tugas 1

Pseudo code

```
program KategoriPHAir
kamus
    pH : real
algoritma
    input(pH)
    depend on
        pH >= 6.5 and pH <= 8.6:
            output("Air layak minum")
        pH < 6.5 or (pH > 8.6 and pH <= 14):
            output("Air tidak layak minum")
        pH > 14:
            output("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14")
    end
endprogram
```

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

```

        fmt.Print("Air tidak layak minum")

    case pH > 14:

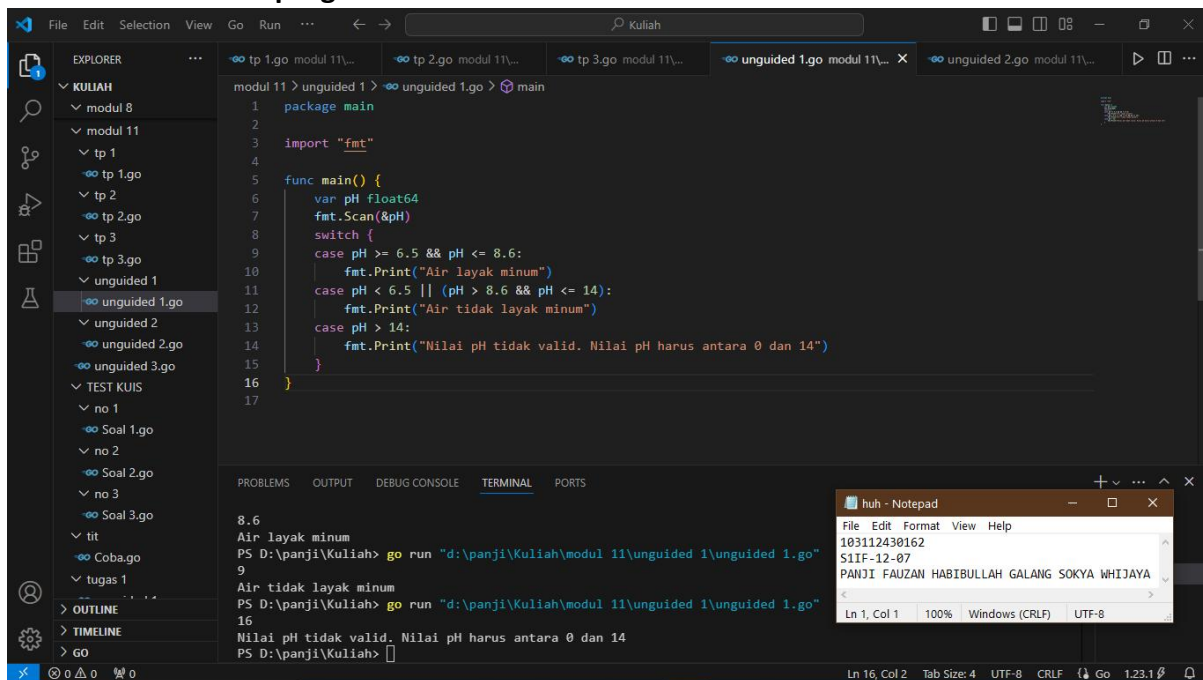
        fmt.Print("Nilai pH tidak valid. Nilai pH harus
antara 0 dan 14")

    }

}

```

Screenshoot program



Deskripsi program

- ◆ package main digunakan dalam Go Lang untuk menandai bahwa ini adalah program yang dapat dieksekusi.
- ◆ import "fmt" digunakan untuk operasi input/output seperti mencetak pesan ke layar dan mengambil input dari pengguna.
- ◆ func main() adalah titik masuk dari program. Semua perintah dalam program akan dieksekusi di dalam fungsi ini.
- ◆ Mendeklarasikan variabel pH dengan tipe float64 untuk menyimpan nilai pH yang dimasukkan oleh pengguna.
- ◆ fmt.Scan(&pH) digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel pH.
- ◆ switch digunakan untuk memeriksa nilai pH dan menentukan kategori kualitas air berdasarkan rentang nilai pH yang diberikan.
- ◆ case pH >= 6.5 && pH <= 8.6 akan mencetak "Air layak minum" jika nilai pH berada dalam rentang yang dianggap layak minum, yaitu antara 6.5 dan 8.6.

- ◆ case $pH < 6.5 \vee (pH > 8.6 \wedge pH \leq 14)$ mencetak "Air tidak layak minum" jika nilai pH lebih rendah dari 6.5 atau lebih besar dari 8.6 namun tidak melebihi 14, yang menunjukkan kondisi yang tidak layak untuk diminum.
- ◆ case $pH > 14$ mencetak "Nilai pH tidak valid. Nilai pH harus antara 0 dan 14" jika nilai pH lebih besar dari 14, yang berada di luar rentang nilai pH yang valid (0 hingga 14).

2. Tugas 2

Pseudo code

```

program BiayaParkir
kamus
    kendaraan : string
    durasi : real
    TBiayaParkir : integer
algoritma
    input(kendaraan, durasi)
    output(kendaraan, durasi, "jam")
    if durasi < 1 then
        durasi = 1
    else
        durasi = round(durasi)
    depend on kendaraan
        "Motor" or "motor":
            TBiayaParkir = 2000 * durasi
        "Mobil" or "mobil":
            TBiayaParkir = 5000 * durasi
        "Truk" or "truk" or "Truck" or "truck":
            TBiayaParkir = 8000 * durasi
    end
    output("Rp ", TBiayaParkir)
endprogram

```

Source code

```

package main

import (

```

```
    "fmt"

    "math"

)

func main() {

    var kendaraan string

    var durasi float64

    var TBiayaParkir int

    fmt.Scan(&kendaraan, &durasi)

    fmt.Println(kendaraan, durasi, "jam")

    if durasi < 1 {

        durasi = 1

    } else {

        durasi = math.Round(durasi)

    }

    switch kendaraan {

    case "Motor", "motor":

        TBiayaParkir = 2000 * int(durasi)

    case "Mobil", "mobil":

        TBiayaParkir = 5000 * int(durasi)

    case "Truk", "truk", "Truck", "truck":

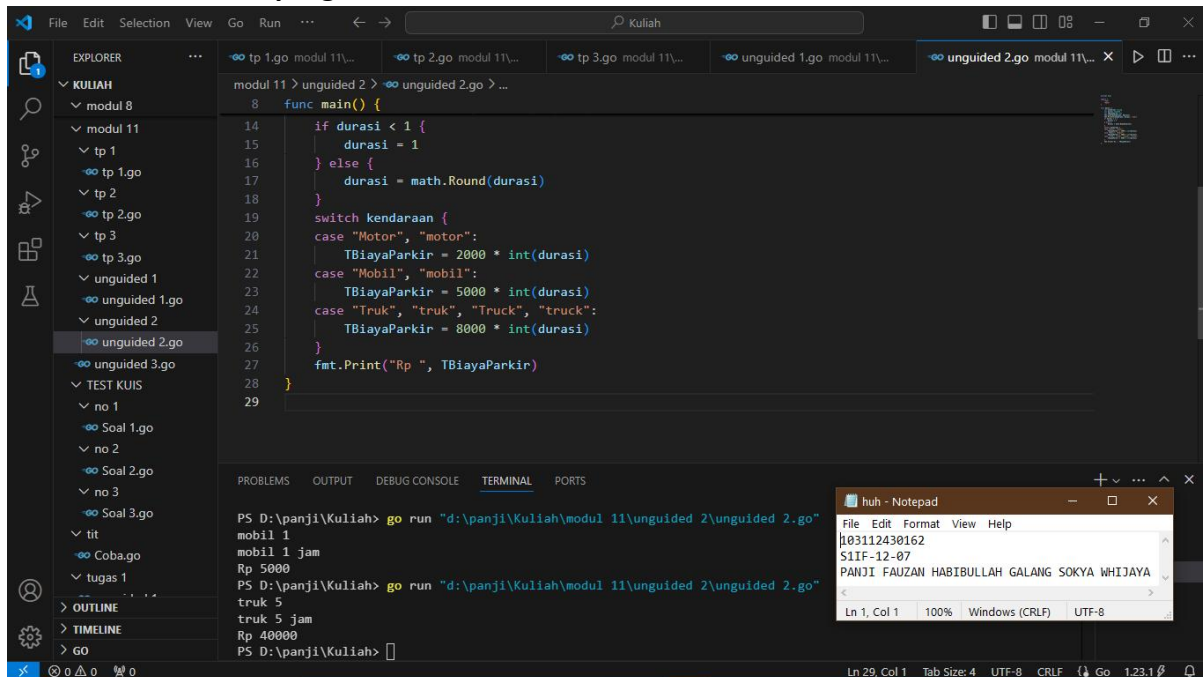
        TBiayaParkir = 8000 * int(durasi)

    }

    fmt.Print("Rp ", TBiayaParkir)

}
```

Screenshoot program



```
8 func main() {  
14     if durasi < 1 {  
15         durasi = 1  
16     } else {  
17         durasi = math.Round(durasi)  
18     }  
19     switch kendaraan {  
20     case "Motor", "motor":  
21         TBiayaParkir = 2000 * int(durasi)  
22     case "Mobil", "mobil":  
23         TBiayaParkir = 5000 * int(durasi)  
24     case "Truk", "truk", "Truck", "truck":  
25         TBiayaParkir = 8000 * int(durasi)  
26     }  
27     fmt.Print("Rp ", TBiayaParkir)  
28 }  
29
```

```
PS D:\panji\Kuliah> go run "d:\panji\Kuliah\modul 11\unguided 2\unguided 2.go"  
mobil 1  
Rp 5000  
PS D:\panji\Kuliah> go run "d:\panji\Kuliah\modul 11\unguided 2\unguided 2.go"  
truk 5  
Rp 40000  
PS D:\panji\Kuliah>
```

Deskripsi program

- ◆ package main digunakan dalam Go Lang untuk menandai bahwa ini adalah program yang dapat dieksekusi.
- ◆ import "fmt" digunakan untuk operasi input/output seperti mencetak pesan ke layar dan mengambil input dari pengguna.
- ◆ import "math" digunakan untuk melakukan operasi matematika, khususnya fungsi pembulatan (rounding) pada durasi parkir.
- ◆ func main() adalah titik masuk dari program. Semua perintah dalam program akan dieksekusi di dalam fungsi ini.
- ◆ Mendeklarasikan variabel kendaraan dengan tipe string untuk menyimpan jenis kendaraan (Motor, Mobil, atau Truk) yang dimasukkan oleh pengguna.
- ◆ Mendeklarasikan variabel durasi dengan tipe float64 untuk menyimpan durasi parkir yang dimasukkan oleh pengguna dalam jam.
- ◆ Mendeklarasikan variabel TBiayaParkir dengan tipe integer untuk menyimpan total biaya parkir yang dihitung berdasarkan jenis kendaraan dan durasi parkir.
- ◆ fmt.Scan(&kendaraan, &durasi) digunakan untuk mengambil input jenis kendaraan dan durasi parkir dari pengguna.
- ◆ fmt.Println(kendaraan, durasi, "jam") digunakan untuk mencetak jenis kendaraan dan durasi parkir yang dimasukkan oleh pengguna.
- ◆ if durasi < 1 memeriksa apakah durasi parkir kurang dari 1 jam. Jika ya, maka durasi diubah menjadi 1 jam, karena parkir dihitung minimal 1 jam.
- ◆ else jika durasi lebih dari atau sama dengan 1 jam, durasi dibulatkan ke angka terdekat menggunakan math.Round(durasi).
- ◆ switch kendaraan digunakan untuk memeriksa jenis kendaraan yang dipilih dan menghitung biaya parkir berdasarkan tarif yang berbeda:

- ◆ case "Motor", "motor" menghitung biaya parkir untuk motor dengan tarif Rp 2000 per jam.
- ◆ case "Mobil", "mobil" menghitung biaya parkir untuk mobil dengan tarif Rp 5000 per jam.
- ◆ case "Truk", "truk", "Truck", "truck" menghitung biaya parkir untuk truk dengan tarif Rp 8000 per jam.
- ◆ `fmt.Print("Rp ", TBiayaParkir)` mencetak total biaya parkir yang dihitung dalam format Rupiah (Rp).

3. Tugas 3

Pseudo code

```

program KategoriBilangan
kamus
    bil : integer
    hasil : integer
algoritma
    input(bil)
    depend on
        bil % 10 = 0 dan bil > 10:
            hasil = bil / 10
            output("Kategori: Bilangan Kelipatan 10")
            output("Hasil pembagian antara ", bil, " / ", 10, " = ", hasil)
        bil % 2 = 0:
            hasil = (bil + 1) * bil
            output("Kategori: Bilangan Genap")
            output("Hasil perkalian dengan bilangan berikutnya ", bil, " * ", bil + 1, " = ", hasil)
        bil % 5 = 0 dan bil > 5:
            hasil = bil * bil
            output("Kategori: Bilangan Kelipatan 5")
            output("Hasil kuadrat dari ", bil, " ^2 = ", hasil)
        default:
            hasil = (bil + 1) + bil
            output("Kategori: Bilangan Ganjil")
            output("Hasil penjumlahan dengan bilangan berikutnya ", bil, " + ", bil + 1, " = ",
hasil)

```

```
end  
endprogram
```

Source code

```
package main  
  
import "fmt"  
  
func main() {  
    var bil, hasil int  
    fmt.Scan(&bil)  
    switch {  
    case bil%10 == 0 && bil > 10:  
        hasil = bil / 10  
        fmt.Println("Kategori: Bilangan Kelipatan 10")  
        fmt.Print("Hasil pembagian antara ", bil, " / ",  
10, " = ", hasil)  
    case bil%2 == 0:  
        hasil = (bil + 1) * bil  
        fmt.Println("Kategori: Bilangan Genap")  
        fmt.Print("Hasil perkalian dengan bilangan  
berikutnya ", bil, " * ", bil+1, " = ", hasil)  
    case bil%5 == 0 && bil > 5:  
        hasil = bil * bil  
        fmt.Println("Kategori: Bilangan Kelipatan 5")  
        fmt.Print("Hasil kuadrat dari ", bil, " ^2 = ",  
hasil)  
    default:
```

```

        hasil = (bil + 1) + bil

        fmt.Println("Kategori: Bilangan Ganjil")

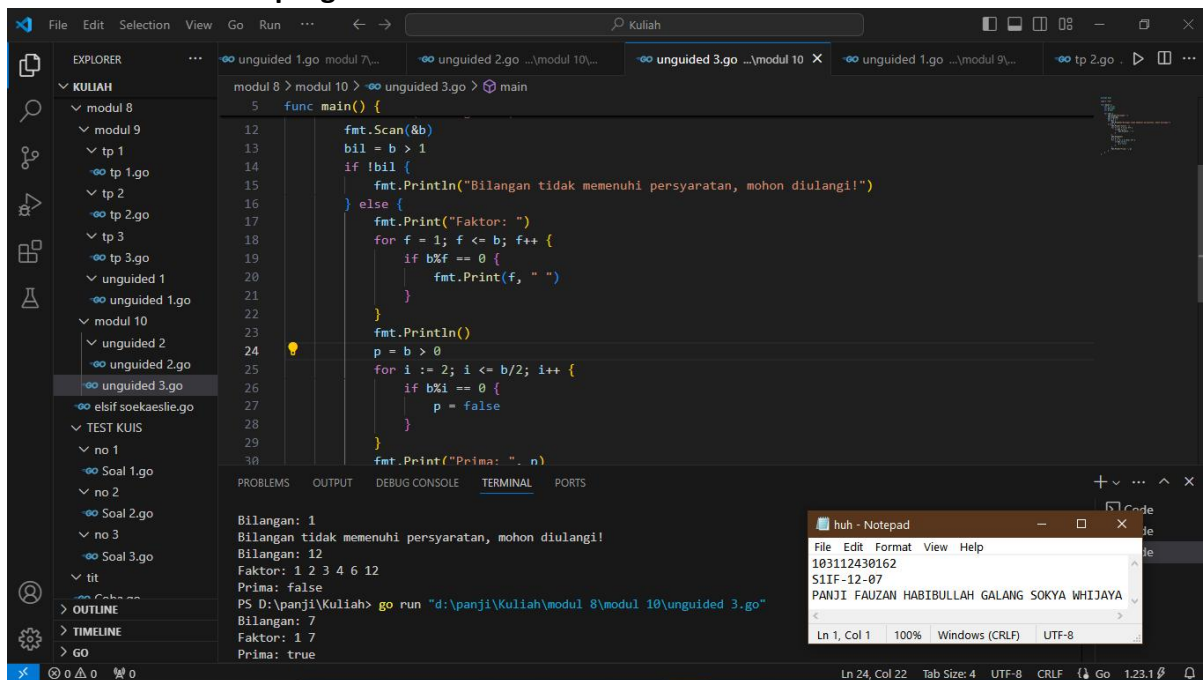
        fmt.Print("Hasil penjumlahan dengan bilangan
berikutnya ", bil, " + ", bil+1, " = ", hasil)

    }

}

```

Screenshoot program



Deskripsi program

- ◆ package main digunakan dalam Go Lang untuk menandai bahwa ini adalah program yang dapat dieksekusi.
- ◆ import "fmt" digunakan untuk operasi input/output seperti mencetak pesan ke layar dan mengambil input dari pengguna.
- ◆ func main() adalah titik masuk dari program. Semua perintah dalam program akan dieksekusi di dalam fungsi ini.
- ◆ Mendeklarasikan variabel bil dan hasil dengan tipe integer untuk menyimpan bilangan yang dimasukkan oleh pengguna dan hasil dari operasi yang dihitung.
- ◆ fmt.Scan(&bil) digunakan untuk mengambil input bilangan dari pengguna.
- ◆ switch digunakan untuk memeriksa kondisi dari bilangan yang dimasukkan dan mengelompokkan ke dalam kategori yang sesuai. Ada empat kemungkinan kategori:
- ◆ case bil%10 == 0 && bil > 10 memeriksa apakah bilangan adalah kelipatan 10 dan lebih besar dari 10. Jika iya, maka program akan membagi bilangan tersebut dengan 10 dan mencetak kategori "Bilangan Kelipatan 10", serta hasil pembagian.

- ◆ `case bil%2 == 0` memeriksa apakah bilangan tersebut genap. Jika genap, maka program akan mengalikan bilangan dengan bilangan berikutnya (`bilangan + 1`) dan mencetak kategori "Bilangan Genap", serta hasil perkalian tersebut.
- ◆ `case bil%5 == 0 && bil > 5` memeriksa apakah bilangan tersebut kelipatan 5 dan lebih besar dari 5. Jika iya, maka program akan menghitung kuadrat dari bilangan tersebut dan mencetak kategori "Bilangan Kelipatan 5", serta hasil kuadrat.
- ◆ default jika bilangan tidak memenuhi kondisi sebelumnya, maka dianggap sebagai bilangan ganjil. Program akan menjumlahkan bilangan tersebut dengan bilangan berikutnya (`bilangan + 1`) dan mencetak kategori "Bilangan Ganjil", serta hasil penjumlahan tersebut.