

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 11
Switch Case



Disusun oleh:

ULIL FAHMI NURWIDYA

103112430267

S1IF-12-07

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

DASAR TEORI

Switch case merupakan mekanisme kontrol dengan pemilihan dalam bahasa pemrograman computer yang mungkin nilai variabel / ekspresi mengubah aliran kontrol program, switch case juga merupakan pernyataan pilihan ganda, struktur switch case nilai yang di evaluasi hanya di tentuin sekali, switch di pemrograman Go mempunyai perbedaan dengan bahasa lain. Di Go, jika sebuah case terpenuhi, tidak akan dilanjutkan ke pengecekan case selanjutnya, meskipun tidak ada keyword break di situ. Konsep ini berkebalikan dengan switch pada umumnya pemrograman lain (yang ketika sebuah case terpenuhi, maka akan tetap dilanjut mengecek case selanjutnya kecuali ada keyword break). Sebuah case dapat menampung banyak kondisi

TUGAS PENDAHULUAN

1. Tugas 1

Pseudo code

```
Program Umur Kamus umur: Int

Algoritma
Mulai program.
Input nilai umur.
SWITCH

Case 1: Jika umur == 0: => "Tidak terdefinisi".
Case 2: Jika umur < 13: => "Anak-anak".
Case 3: Jika umur < 20 => "Remaja".
Case 4: Jika umur < 6 => "Dewasa".
Case 6: Jika umur >= 60 => "Lansia".

Akhir program.
```

Source code

```
package main

import "fmt"

func kategoriUmur(umur int) string {
```

```
switch {  
    case umur == 0:  
        return "Tidak Terdefinisi"  
    case umur < 13:  
        return "Anak-Anak"  
    case umur < 20:  
        return "Remaja"  
    case umur < 60:  
        return "Dewasa"  
    case umur >= 60:  
        return "Lansia"  
    default:  
        return "Data tidak valid"  
}  
  
}  
  
func main() {  
    var umur int  
  
    fmt.Print("Masukkan umur: ")  
    fmt.Scan(&umur)  
  
    fmt.Println("Termasuk dalam kategori:", kategoriUmur(umur))  
}
```

Screenshoot program

The screenshot shows a Go IDE with a file named `tp1.go` in the `modul 11` directory. The code defines a function `kategoriUmur` that categorizes age into "Tidak Terdefinisi", "Anak-Anak", "Remaja", "Dewasa", "Lansia", or "Data tidak valid" based on the input age. The `main` function prompts the user for an age and prints the category.

```
3 import "fmt"
4
5 func kategoriUmur(umur int) string {
6     switch {
7     case umur == 0:
8         return "Tidak Terdefinisi"
9     case umur < 13:
10        return "Anak-Anak"
11    case umur < 20:
12        return "Remaja"
13    case umur < 60:
14        return "Dewasa"
15    case umur >= 60:
16        return "Lansia"
17    default:
18        return "Data tidak valid"
19    }
20 }
21
22 func main() {
23     var umur int
24
25     fmt.Print("Masukkan umur: ")
26     fmt.Scan(&umur)
27
28     fmt.Println("Termasuk dalam kategori:", kategoriUmur(umur))
29 }
```

The terminal output shows the program being run twice. In the first run, the user enters 12, and the output is "Termasuk dalam kategori: Anak-Anak". In the second run, the user enters 21, and the output is "Termasuk dalam kategori: Dewasa".

```
PS D:\praktik> go run "d:\praktik\modul 11\tp1.go"
Masukkan umur: 12
Termasuk dalam kategori: Anak-Anak
PS D:\praktik> go run "d:\praktik\modul 11\tp1.go"
Masukkan umur: 21
Termasuk dalam kategori: Dewasa
PS D:\praktik>
```

Deskripsi program

Program di atas user meminta program untuk mengkategorikan umur user berdasarkan rentang usia tertentu: "tidak terdefinisi dalam umur" untuk input umur 0. "anak-anak" untuk umur di bawah 13 tahun. "remaja" untuk umur antara 13 hingga kurang dari 20 tahun. "dewasa" untuk umur antara 20 hingga kurang dari 60 tahun. "lansia" untuk umur 60 tahun ke atas.

2. Tugas 2

Pseudo code

Program Hari

rumus var hari

int var hasil

string input

kode

algoritma

switch

case 1 : jika kode bernilai 1 maka kondisi : hari := senin case 2 : jika kode bernilai 2 maka kondisi 2 : hari := selasa case 3 : jika kode bernilai 3 maka kondisi 3 : hari := rabu case 4 : jika kode bernilai 4 maka kondisi 4 : hari := kamis case 5 : jika kode bernilai 5 maka kondisi 5 : hari := jumat case 6 : jika kode bernilai 6 maka kondisi 6 : hari := sabtu case 7 : jika kode bernilai 7 maka kondisi 7 : hari

:= minggu default tidak ada yang terpenuhi maka kode tidak ditemukan output print hari end switch
end program

Source code

```
package main

import "fmt"

func getHari(kode int) string {
    switch kode {
    case 1:
        return "Senin"
    case 2:
        return "Selasa"
    case 3:
        return "Rabu"
    case 4:
        return "Kamis"
    case 5:
        return "Jumat"
    case 6:
        return "Sabtu"
    case 7:
        return "Minggu"
    default:
        return "Kode Tidak Ditemukan"
    }
}

func main() {
    var kode int

    fmt.Print("Masukkan kode hari: ")
    fmt.Scan(&kode)
```

```

    hari := getHari(kode)

    fmt.Println("Hari:", hari)
}

```

Screenshoot program

The screenshot shows a Go IDE with the following source code in `tp2.go`:

```

4  func getHari(kode int) string {
14     case 5:
15         return "Jumat"
16     case 6:
17         return "Sabtu"
18     case 7:
19         return "Minggu"
20     default:
21         return "Kode Tidak Ditemukan"
22     }
23 }
24
25 func main() {
26     var kode int
27
28     fmt.Print("Masukkan kode hari: ")
29     fmt.Scan(&kode)
30
31     hari := getHari(kode)
32     fmt.Println("Hari:", hari)
33 }
34

```

The terminal output shows the program being run twice. In the first run, the user enters '2' and the output is 'Hari: Selasa'. In the second run, the user enters '20' and the output is 'Hari: Kode Tidak Ditemukan'.

Deskripsi program

Kode program di atas berfungsi untuk menampilkan nama hari berdasarkan kode hari yang dimasukkan pengguna.

3. Tugas 3

Pseudo code

```

Program Film rumus
var kode film string
input kendaraan dan waktu
algoritma
switch Kode
"G": untuk semua umur
"PG": untuk anak-anak diatas 7 tahun
"PG-13": untuk remaja diatas 13 tahun
"R": untuk dewasa
Default: Jika kode tidak ditemukan, mencetak pesan: kode tidak ditemukan
end switch

```

```
end program
```

Source code

```
package main

import "fmt"

func deskripsiFilm(kode string) string {
    switch kode {
        case "G":
            return "untuk semua umur"
        case "PG":
            return "untuk anak-anak di atas 7 tahun"
        case "PG-13":
            return "untuk remaja di atas 13 tahun"
        case "R":
            return "untuk dewasa"
        default:
            return "kode tidak ditemukan"
    }
}

func main() {
    var kode string
    fmt.Print("Masukkan kode film: ")
    fmt.Scan(&kode)
    fmt.Println("Kategori:", deskripsiFilm(kode))
}
```

Screenshoot program

```
modul 11 > tp 3 > tp3.go > ...
1 package main
2
3 import "fmt"
4
5 func deskripsiFilm(kode string) string {
6     switch kode {
7         case "G":
8             return "untuk semua umur"
9         case "PG":
10            return "untuk anak-anak di atas 7 tahun"
11        case "PG-13":
12            return "untuk remaja di atas 13 tahun"
13        case "R":
14            return "untuk dewasa"
15        default:
16            return "kode tidak ditemukan"
17        }
18    }
19
20 func main() {
21     var kode string
22     fmt.Print("Masukkan kode film: ")
23     fmt.Scan(&kode)
24     fmt.Println("Kategori:", deskripsiFilm(kode))
25 }
26
```

PROBLEMS OUTPUT **TERMINAL** PORTS DEBUG CONSOLE

```
Hari: Selasa
PS D:\praktik> go run "d:\praktik\modul 11\tp2\tp2.go"
Masukkan kode hari: 20
Hari: Kode Tidak Ditemukan
PS D:\praktik> go run "d:\praktik\modul 11\tp 3\tp3.go"
Masukkan kode film: g
Kategori: kode tidak ditemukan
PS D:\praktik> go run "d:\praktik\modul 11\tp 3\tp3.go"
Masukkan kode film: G
Kategori: untuk semua umur
PS D:\praktik>
```

*Untitled - Notepad

File Edit Format View Help

nim : 103112430178

nama : ULIL FAHMI NURWIDYA

kelas : 12 IF - 07

Ln 3, 100% Windows (CRLF) UTF-8

Deskripsi program

Program di atas bertujuan untuk memberikan informasi tentang kategori penonton berdasarkan kode film.

LATIHAN MODUL

1. Tugas 1

Pseudo code

Kamus

ph: Floa

Algoritma

Mulai program.

Input nilai ph.

Switch Kondisi 1: Jika $ph \geq 6.5$ dan $ph \leq 8.6$: \Rightarrow "Air layak diminum".

Kondisi 2: Jika $ph < 6.5$ dan $ph > 0$ atau $ph > 8.6$ dan $ph \leq 14$: \Rightarrow "Air tidak layak diminum".

Kondisi 3: Jika $ph > 14$ atau $ph < 0$: \Rightarrow "Inputan tidak valid, harus antara 0-14".

End program

Source code

```
package main

import "fmt"

func cekKelayakanAir(ph float64) string {
    switch {
    case ph >= 6.5 && ph <= 8.6:
        return "Air layak diminum"
    case (ph < 6.5 && ph > 0) || (ph > 8.6 && ph <= 14):
        return "Air tidak layak diminum"
    case ph > 14 || ph < 0:
        return "Inputan tidak valid, harus antara 0-14"
    default:
        return "Inputan tidak valid"
    }
}

func main() {
    var ph float64
    fmt.Print("Masukkan kadar pH: ")
    fmt.Scanln(&ph)
    fmt.Println(cekKelayakanAir(ph))
}
```

Screenshoot program

```
modul 11 > unguiden1 > unguided1.go > ...
1 package main
2
3 import "fmt"
4 func cekKelayakanAir(ph float64) string {
5     switch {
6     case ph >= 6.5 && ph <= 8.6:
7         return "Air layak diminum"
8     case (ph < 6.5 && ph > 0) || (ph > 8.6 && ph <= 14):
9         return "Air tidak layak diminum"
10    case ph > 14 || ph < 0:
11        return "Inputan tidak valid, harus antara 0-14"
12    default:
13        return "Inputan tidak valid"
14    }
15 }
16
17 func main() {
18     var ph float64
19     fmt.Print("Masukkan kadar pH: ")
20     fmt.Scanln(&ph)
21     fmt.Println(cekKelayakanAir(ph))
22 }
23
```

*Untitled - Notepad

File Edit Format View Help

nim : 103112430178

nama : ULIL FAHMI NURWIDYA

kelas : 12 IF - 07

Ln 3, 100% Windows (CRLF) UTF-8

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

Kategori: kode tidak ditemukan

- PS D:\praktik> go run "d:\praktik\modul 11\tp 3\tp3.go"

Masukkan kode film: G

Kategori: untuk semua umur

- PS D:\praktik> go run "d:\praktik\modul 11\unguiden1\unguided1.go"

Masukkan kadar pH: 6

Air tidak layak diminum

- PS D:\praktik> go run "d:\praktik\modul 11\unguiden1\unguided1.go"

Masukkan kadar pH: 6.7

Air layak diminum

PS D:\praktik>

Deskripsi program

Program tersebut untuk membuat sebuah inputan berupa ph dari suatu air atau larutan dan kemudian mengeceknya apakah layak untuk diminum atau tidak.

2. Tugas 2

Pseudo code

Program Parkir

Kamus

Var jenisKendaraan:String var durasiParkir int Var tarifPerJam: Int totalBiaya: Int

Algoritma mulai program

Masukkan jenis kendaraan (motor, mobil, truk):

Baca input jenisKendaraan

Masukkan durasi parkir (dalam jam):

Jika durasiParkir kurang dari 1, set durasiParkir = 1 (durasi minimum adalah 1 jam).

Gunakan pernyataan switch untuk menentukan tarifPerJam berdasarkan jenisKendaraan:

Jika jenisKendaraan = "motor", set tarifPerJam = 2000.

Jika jenisKendaraan = "mobil", set tarifPerJam = 5000.

Jika jenisKendaraan = "truk", set tarifPerJam = 8000. Jika jenisKendaraan = jenis kendaraan tidak valid

totalBiaya = tarifPerJam * durasiParkir.

end.

Source code

```
package main

import "fmt"

func getTarifPerJam(jenisKendaraan string) (int, error) {
    switch jenisKendaraan {
        case "motor":
            return 2000, nil
        case "mobil":
            return 5000, nil
        case "truk":
            return 8000, nil
        default:
            return 0, fmt.Errorf("jenis kendaraan tidak valid")
    }
}

func hitungBiayaParkir(tarifPerJam, durasiParkir int) int {
    if durasiParkir < 1 {
        durasiParkir = 1
    }
    return tarifPerJam * durasiParkir
}

func main() {
    var jenisKendaraan string
    var durasiParkir int
    fmt.Print("Masukkan jenis kendaraan (motor, mobil, truk): ")
}
```

```

    fmt.Scanln(&jenisKendaraan)

    fmt.Print("Masukkan durasi parkir (dalam jam): ")

    fmt.Scanln(&durasiParkir)

    tarifPerJam, err := getTarifPerJam(jenisKendaraan)

    if err != nil {

        fmt.Println(err)

        return

    }

    totalBiaya := hitungBiayaParkir(tarifPerJam, durasiParkir)

    fmt.Printf("Total biaya parkir: Rp %d\n", totalBiaya)

}

```

Screenshoot program

The screenshot shows a Go IDE with the following source code in `modul 11\unguided2.go`:

```

1 package main
2
3 import "fmt"
4
5 func getTarifPerJam(jenisKendaraan string) (int, error) {
6     switch jenisKendaraan {
7     case "motor":
8         return 2000, nil
9     case "mobil":
10        return 5000, nil
11    case "truk":
12        return 8000, nil
13    default:
14        return 0, fmt.Errorf("jenis kendaraan tidak valid")
15    }
16 }
17
18 func hitungBiayaParkir(tarifPerJam, durasiParkir int) int {
19     if durasiParkir < 1 {
20         durasiParkir = 1
21     }
22     return tarifPerJam * durasiParkir
23 }
24
25 func main() {
26     var jenisKendaraan string
27     var durasiParkir int

```

The terminal output shows the program's execution:

```

PS D:\praktik> go run "d:\praktik\modul 11\unguiden1\unguided1.go"
Masukkan kadar pH: 6
Air tidak layak diminum
PS D:\praktik> go run "d:\praktik\modul 11\unguiden1\unguided1.go"
Masukkan kadar pH: 6.7
Air layak diminum
PS D:\praktik> go run "d:\praktik\modul 11\unguided2\unguided2.go"
Masukkan jenis kendaraan (motor, mobil, truk): motor
Masukkan durasi parkir (dalam jam): 3
Total biaya parkir: Rp 6000
PS D:\praktik>

```

A Notepad window titled "Untitled - Notepad" is open in the foreground, displaying the following text:

```

File Edit Format View Help
nim : 103112430178
nama : ULIL FAHMI NURWIDYA
kelas : 12 IF - 07

```

Deskripsi program

Program meminta pengguna memasukkan jenis kendaraan: "motor", "mobil", atau "truk". Jika jenis kendaraan tidak sesuai dengan pilihan yang tersedia, program mencetak pesan tidak ditemukan. Jika durasi yang dimasukkan kurang dari 1, program secara otomatis menetapkan durasi menjadi 1 jam (durasi minimum parkir).

3. Tugas 3

Pseudo code

```
Program operasi matematika

Kamus

Variabel: angka:
Int bagi:
Boolean.
kuadrat: Boolean
ganjil: Boolean
genap: Boolean
angka2: Integer
hasil: Int


Algoritma

Mulai program.

Baca input angka.

Hitung kondisi:

bagi: Apakah angka kelipatan 10. kuadrat:
Apakah angka kelipatan 5. ganjil: Apakah
angka adalah bilangan ganjil. genap: Apakah
angka adalah bilangan genap.

Switch

Kategori: Bilangan Kelipatan 10.
Hitung hasil = angka / 10.

Kategori: Bilangan Kelipatan 5.
Hitung hasil = angka * angka.

Kategori: Bilangan Ganjil.
Hitung angka2 = angka + 1, hasil = angka + angka2.

Kategori: Bilangan Genap.
Hitung angka2 = angka + 1, hasil = angka * angka2.

Akhir program.end program
```

Source code

```
package main

import "fmt"

func cekKelipatan10(angka int) (bool, int) {
    if angka%10 == 0 && angka > 10 {
        return true, angka / 10
    }
    return false, 0
}

func cekKelipatan5(angka int) (bool, int) {
    if angka%5 == 0 && angka > 5 {
        return true, angka * angka
    }
    return false, 0
}

func cekGanjil(angka int) (bool, int, int) {
    if angka%2 != 0 {
        angka2 := angka + 1
        return true, angka, angka + angka2
    }
    return false, 0, 0
}

func cekGenap(angka int) (bool, int, int) {
    if angka%2 == 0 {
        angka2 := angka + 1
        return true, angka, angka * angka2
    }
    return false, 0, 0
}
```

```
func main() {  
    var angka int  
    fmt.Print("Masukkan angka: ")  
    fmt.Scan(&angka)  
  
    if ok, hasil := cekKelipatan10(angka); ok {  
        fmt.Println("Kategori = Bilangan kelipatan 10")  
        fmt.Printf("Hasil pembagian berikutnya %d / 10 = %d\n",  
angka, hasil)  
    } else if ok, hasil := cekKelipatan5(angka); ok {  
        fmt.Println("Kategori = Bilangan kelipatan 5")  
        fmt.Printf("Hasil kuadrat berikutnya %d^2 = %d\n", angka,  
hasil)  
    } else if ok, angka1, hasil := cekGanjil(angka); ok {  
        fmt.Println("Kategori = Bilangan ganjil")  
        fmt.Printf("Hasil penjumlahan bilangan berikutnya %d + %d =  
%d\n", angka1, angka1+1, hasil)  
    } else if ok, angka1, hasil := cekGenap(angka); ok {  
        fmt.Println("Kategori = Bilangan genap")  
        fmt.Printf("Hasil perkalian bilangan berikutnya %d * %d =  
%d\n", angka1, angka1+1, hasil)  
    }  
}
```

Screenshoot program

```
modul 11 > unguided3 > unguided3.go > cekKelipatan10
1 package main
2
3 import "fmt"
4
5 func cekKelipatan10(angka int) (bool, int) {
6     if angka%10 == 0 && angka > 10 {
7         return true, angka / 10
8     }
9     return false, 0
10 }
11
12 func cekKelipatan5(angka int) (bool, int) {
13     if angka%5 == 0 && angka > 5 {
14         return true, angka * angka
15     }
16     return false, 0
17 }
18
19 func cekGanjil(angka int) (bool, int, int) {
20     if angka%2 != 0 {
21         angka2 := angka + 1
22         return true, angka, angka + angka2
23     }
24     return false, 0, 0
25 }
26 func cekGenap(angka int)
27     if angka%2 == 0 {
```

Debug Console (Ctrl+Shift+Y)

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

Masukkan kadar pH: 6.7
Air layak diminum
PS D:\praktik> go run "d:\praktik\modul 11\unguided2\unguided2.go"
Masukkan jenis kendaraan (motor, mobil, truk): motor
Masukkan durasi parkir (dalam jam): 3
Total biaya parkir: Rp 6000
PS D:\praktik> go run "d:\praktik\modul 11\unguided3\unguided3.go"
Masukkan angka: 5
Kategori = Bilangan ganjil
Hasil penjumlahan bilangan berikutnya 5 + 6 = 11
PS D:\praktik>

*Untitled - Notepad
File Edit Format View Help
nim : 103112430178
nama : ULIL FAHMI NURWIDYA
kelas : 12 IF - 07
Ln 3, 100% Windows (CRLF) UTF-8

Deskripsi program

Program di atas ditulis dalam bahasa Go dan berfungsi untuk mengidentifikasi kategori suatu bilangan bulat berdasarkan sifat atau pola tertentu.