

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 11

SWITCH-CASE



Disusun oleh:

Nabyla Zahirah Ramadhani

109082500104

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

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 jam int

    fmt.Println("Masukkan jam : ")

    fmt.Scan(&jam)

    switch jam {

        case 0:

            fmt.Println("12 AM")

        case 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11:

            fmt.Println(jam, "AM")

        case 12:

            fmt.Println("12 PM")

        case 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23:

            fmt.Println(jam-12, "PM")

        default:

            fmt.Println("input tidak valid")
    }
}
```

Screenshot program

```
week 11
guided1.go
package main
import "fmt"
func main () {
    var jam int
    fmt.Print("Masukkan jam : ")
    fmt.Scan(&jam)

    switch jam {
        case 0:
            fmt.Println("12 AM")
        case 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11:
            fmt.Println(jam, "AM")
        case 12:
            fmt.Println("12 PM")
        case 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23:
            fmt.Println(jam-12, "PM")
        default:
            fmt.Println("input tidak valid")
    }
}

Nama: Nabyla Zahirah Ramadhan
NIM: 109082500104
Kelas: S1IF-13-07

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS EXPLORER
go run "/Users/nabylaaa/Documents/go/week 11/guided1.go"
nabylaaa@Nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/guided1.go"
Masukkan jam : 13
1 PM
nabylaaa@Nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/guided1.go"
Masukkan jam : 25
input tidak valid
nabylaaa@Nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/guided1.go"
Masukkan jam : 21
9 PM
nabylaaa@Nabylas-MacBook-Air week 11 %
```

Deskripsi program

Program ini melakukan konversi waktu dari bentuk 24 jam ke bentuk 12 jam, variabelnya jam dan tipenya int. Pertama program menampilkan teks `fmt.Println("Masukkan jam : ")` untuk meminta input jam dalam format 24 jam. Nilai yang di input user akan dibaca menggunakan `fmt.Scan(&jam)`. Setelah mendapatkan input, program masuk ke bagian `switch jam { ... }`.

Pada case 0, program mencetak `fmt.Println("12 AM")` karena dalam format 12 jam, pukul 0 (00:00) dianggap sebagai jam 12 AM.

Pada case 1, 2, 3, ..., 11, program menuliskan `fmt.Println(jam, "AM")`.

Pada case 12, program mencetak `fmt.Println("12 PM")`

Pada case 13 sampai 23, program menjalankan `fmt.Println(jam-12, "PM")`. Bagian `jam - 12` digunakan untuk mengubah jam 24-jam menjadi format 12-jam. Misalnya, kalau user memasukkan 15, maka program menghitung $15 - 12 = 3$, lalu mencetak 3 PM.

Bagian default akan dieksekusi jika angka jam tidak berada antara 0–23. Pada kondisi ini program menampilkan "input tidak valid" untuk memberitahu bahwa pengguna memasukkan jam yang tidak ada di case.

2. Guided 2

Source Code

```
package main

import "fmt"

func main () {

    var nama_tanaman string

    fmt.Print("Masukkan nama tanaman: ")

    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.")


    }
}
```

Screenshot program

```

package main
import "fmt"
func main () {
    var nama_tanaman string
    fmt.Println("Masukkan nama tanaman: ")
    fmt.Scan(&nama_tanaman)
    switch nama_tanaman {
        case "nepenthes", "drosera":
            fmt.Println("Termasuk Tanaman Karnivora.")
        case "venus", "sarracenia":
            fmt.Println("Termasuk Tanaman Karnivora.")
            fmt.Println("Tidak Asli Indonesia.")
        default:
            fmt.Println("Tidak termasuk tanaman karnivora.")
    }
}

```

Nama: Nabyla Zahirah Ramadhan
NIM: 109082500104
Kelas: S1IF-13-07

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS EXPLORER

```

go run "/Users/nabylaaa/Documents/go/week 11/guided2.go"
nabylaaa@labylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/guide
d2.go"
Masukkan nama tanaman: nepenthes
Termasuk Tanaman Karnivora.
Asli Indonesia.
nabylaaa@labylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/guide
d2.go"
Masukkan nama tanaman: venus
Termasuk Tanaman Karnivora.
Tidak Asli Indonesia.
nabylaaa@labylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/guide
d2.go"
Masukkan nama tanaman: karedok
Tidak termasuk tanaman karnivora.
nabylaaa@labylas-MacBook-Air week 11 %

```

Ln 14, Col 17 Tab Size: 4 UTF-8 LF ↵ Go ⌂ 1.25.1 ⌂ ⌂ ⌂

Deskripsi program

Program ini untuk menentukan apakah tanaman tersebut termasuk tanaman karnivora atau tidak. Jika ya, apakah tanaman tersebut asli indonesia atau tidak. Variabelnya `nama_tanaman` tipenya string. Pertama program meminta user input nama tanaman menggunakan `fmt.Print("Masukkan nama tanaman: ")` dan akan dibaca menggunakan `fmt.Scan(&nama_tanaman)`. Berikutnya, program menggunakan switch untuk mengecek nama tanaman yang dimasukkan. Bagian switch `nama_tanaman` berarti program akan mencocokkan isi variabel tersebut dengan beberapa kemungkinan nilai.

Pada case "nepenthes", "drosera", program mencetak:

"Termasuk Tanaman Karnivora."

"Asli Indonesia."

Artinya, jika pengguna memasukkan salah satu dari dua nama itu, program akan menyimpulkan bahwa tanaman tersebut adalah tanaman karnivora dan berasal dari Indonesia.

Pada case "venus", "sarracenia", program menampilkan:

"Termasuk Tanaman Karnivora."

"Tidak Asli Indonesia."

Ini berarti kedua nama tanaman itu juga termasuk karnivora, tetapi bukan tanaman asli Indonesia.

Pada bagian default, program mencetak:

"Tidak termasuk tanaman karnivora."

Bagian default akan dijalankan kalau nama tanaman yang dimasukkan pengguna tidak cocok dengan pilihan yang ada di case mana pun.

3. Guided 3

Source Code

```
package main

import "fmt"

func main () {

    var jenis string

    var durasi, tarif int


        fmt.Print("Masukkan jenis kendaraan
(motor/mobil/truk) : ")

        fmt.Scan(&jenis)

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

        fmt.Scan(&durasi)


    switch jenis {

        case "motor":

            switch {

                case durasi >= 1 && durasi <= 2:

                    tarif = 7000

                case durasi > 2:

                    tarif = 9000

            }

        case "mobil":

            switch {

                case durasi >= 1 && durasi <= 2:
```

```
        tarif = 15000

    case durasi > 2:

        tarif = 20000

    }

case "truk":

    switch {

    case durasi >= 1 && durasi <= 2:

        tarif = 25000

    case durasi > 2:

        tarif = 35000

    }

default:

    fmt.Println("Jenis kendaraan atau durasi
parkir tidak valid")

}

fmt.Printf("Trif parkir: Rp%d\n", tarif)

}
```

Screenshot program

```

package main
import "fmt"
func main () {
    var jenis string
    var durasi, tarif int

    fmt.Println("Masukkan jenis kendaraan (motor/mobil/truk): ")
    fmt.Scan(&jenis)
    fmt.Println("Masukkan durasi parkir (dalam jam): ")
    fmt.Scan(&durasi)

    switch jenis {
    case "motor":
        switch {
        case durasi >= 1 && durasi <= 2:
            tarif = 7000
        case durasi > 2:
            tarif = 9000
        }
    case "mobil":
        switch {
        case durasi >= 1 && durasi <= 2:
            tarif = 15000
        case durasi > 2:
            tarif = 20000
        }
    }
}


```

Nama: Nabyla Zahirah Ramadhan
NIM: 109082500104
Kelas: S1IF-13-07

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS EXPLORER

Ln 37, Col 45 Tab Size: 4 UTF-8 LF ↵ Go ⌂ 1.25.1 ⌂ ⌂

Deskripsi program

Program menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parker, variabelnya jenis tipenya string dan durasi, tarif yang tipenya int. Pertama program meminta input jenis kendaraan menggunakan `fmt.Print("Masukkan jenis kendaraan (motor/mobil/truk): ")`, `fmt.Print("Masukkan durasi parkir (dalam jam): ")` dan akan dibaca menggunakan `fmt.Scan(&jenis)`, `fmt.Scan(&durasi)`. Bagian utama program terdapat pada switch jenis. Jika user input "motor", maka program masuk ke bagian case "motor". Di dalamnya terdapat switch kedua yaitu `switch { ... }` yang digunakan untuk mengecek kondisi durasi. Jika kondisi `durasi >= 1 && durasi <= 2` terpenuhi, program akan memberikan tarif dengan `tarif = 7000`. Namun jika kondisi `durasi > 2` yang terpenuhi, maka program mengubah tarif menjadi `tarif = 9000`.

Jika jenis kendaraan yang diinput adalah "mobil", maka program masuk ke bagian case "mobil" yang juga berisi pengecekan durasi. Jika durasi rentang 1 hingga 2 jam, tarif ditetapkan dengan `tarif = 15000`. Jika lebih dari 2 jam, perintah `tarif = 20000` akan dijalankan.

Jika user input "truk", program menuju ke case "truk". Pada bagian ini juga digunakan pengecekan durasi yang sama. Jika durasinya 1 sampai 2 jam, program menetapkan `tarif = 25000`, dan jika lebih lama dari itu, tarif berubah menjadi `tarif = 35000`.

Jika tipe kendaraan tidak sesuai dengan ketiga pilihan tersebut, bagian default akan dijalankan, yaitu perintah `fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")` yang menandakan adanya input salah. Program mencetak hasil akhirnya melalui `fmt.Printf("Trif parkir: Rp%d\n", tarif)` yang menampilkan tarif parkir sesuai hasil perhitungan berdasarkan jenis kendaraan dan durasi parkir.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main () {

    var ph float64

    fmt.Print("Masukkan kadar ph air: ")

    fmt.Scan(&ph)

    switch {

    case ph < 0 || ph > 14:

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

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

        fmt.Println("Air layak minum")

    case ph < 6.5 || ph > 8.6:

        fmt.Println("Air tidak layak minum")

    }

}
```

Screenshot program

```
week 11
```

```
soal1.go > main
```

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

```
Nama: Nabyla Zahirah Ramadhan  
NIM: 109082500104  
Kelas: S1IF-13-07
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS EXPLORER
```

```
go run "/Users/nabylaaa/Documents/go/week 11/soal1.go"
nabylaaa@Nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/soal1
.go"
Masukkan kadar ph air: 8.6
Air layak minum
nabylaaa@Nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/soal1
.go"
Masukkan kadar ph air: 9
Air tidak layak minum
nabylaaa@Nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/soal1
.go"
Masukkan kadar ph air: 16
Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.
nabylaaa@Nabylas-MacBook-Air week 11 %
```

```
Ln 15, Col 6 Tab Size: 4 UTF-8 LF ↵ Go ⌂ 1.25.1 ⌂ ⌂
```

Deskripsi program

Program ini menentukan kadar ph air, variabelnya ph yang bertipe float64. Pertama program akan meminta input ph menggunakan `fmt.Print("Masukkan kadar ph air: ")` dan akan dibaca menggunakan `fmt.Scan(&ph)`. Setelah nilai pH diperoleh, program masuk ke bagian pemilihan kondisi menggunakan switch tanpa nilai, yaitu `switch { ... }`, switch seperti ini digunakan untuk memeriksa kondisi-kondisi logika secara langsung. Kondisi pertama ditulis pada `case ph < 0 || ph > 14:`. Maksudnya, jika nilai pH yang dimasukkan kurang dari 0 atau lebih dari 14, program akan menjalankan `fmt.Println("Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.")`. Bagian ini dibuat karena skala pH sebenarnya hanya valid dari 0 sampai 14. Jika nilai pH masih berada pada rentang yang benar, program menuju kondisi berikutnya, yaitu `case ph >= 6.5 && ph <= 8.6:`. Jika kondisi ini benar, sehingga program mencetak `fmt.Println("Air layak minum")`. Jika kedua kondisi sebelumnya tidak terpenuhi, program jatuh ke kondisi terakhir yaitu `case ph < 6.5 || ph > 8.6:`. Jika pH terlalu rendah atau terlalu tinggi, program menampilkan `fmt.Println("Air tidak layak minum")`.

2. Tugas 2

Source code

Screenshot program

The screenshot shows a code editor with several tabs open, including 'guided1.go', 'guided2.go', 'guided3.go', 'soal1.go', and 'soal2.go'. The 'soal2.go' tab is active, displaying the following Go code:

```
package main
import "fmt"
func main() {
    var jenis string
    var durasi, tarif int
    fmt.Print("Masukkan jenis kendaraan (motor/mobil/truk): ")
    fmt.Scan(&jenis)
    fmt.Print("Masukkan durasi parkir (jam): ")
    fmt.Scan(&durasi)

    if durasi < 1 {
        durasi = 1
    }

    switch jenis {
    case "motor":
        tarif = 2000
    case "mobil":
        tarif = 5000
    case "truk":
        tarif = 8000
    default:
        fmt.Println("Jenis kendaraan tidak valid")
        return
    }

    total := tarif * durasi
    fmt.Println("Total biaya parkir: Rp", total)
}
```

The output window shows the execution of the program:

```
go run "/Users/nabylaaa/Documents/go/week 11/soal2.go"
Masukkan jenis kendaraan (motor/mobil/truk): motor
Masukkan durasi parkir (jam): 3
Total biaya parkir: Rp 6000
nabylaaa@nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/soal2.go"
Masukkan jenis kendaraan (motor/mobil/truk): mobil
Masukkan durasi parkir (jam): 1
Total biaya parkir: Rp 5000
nabylaaa@nabylas-MacBook-Air week 11 % go run "/Users/nabylaaa/Documents/go/week 11/soal2.go"
Masukkan jenis kendaraan (motor/mobil/truk): truk
Masukkan durasi parkir (jam): 5
Total biaya parkir: Rp 40000
nabylaaa@nabylas-MacBook-Air week 11 %
```

A tooltip box appears over the code editor, containing the student's information:

Nama: Nabyla Zahirah Ramadhan
NIM: 109082500104
Kelas: STIF-13-07

Deskripsi program

Program ini enghitung tarif parker berdasar jenis kendaraan dan durasi parker, variabelnya jenis yang bertipe string dan durasi, tarif yang bertipe int. Pertama program meminta input jenis kendaraan dan durasi menggunakan `fmt.Print("Masukkan jenis kendaraan (motor/mobil/truk): ")`, `fmt.Print("Masukkan durasi parkir (jam): ")` dan akan dibaca menggunakan `fmt.Scan(&jenis)`, `fmt.Scan(&durasi)`. Selanjutnya ada kondisi `if durasi < 1 { durasi = 1 }`, yang berfungsi jika durasi parkir tidak mungkin kurang dari satu jam. Lalu program akan masuk ke `switch jenis {}`, terdapat tiga kemungkinan: jika jenis kendaraan adalah "motor", maka program menetapkan tarif dengan kode `tarif = 2000`; jika jenisnya "mobil", maka program menetapkan tarif `tarif = 5000`; dan jika jenisnya "truk", tarifnya menjadi `8000`. Ketiga bagian ini masing-masing berada di dalam `case "motor"`, `case "mobil"`, dan `case "truk"`. Namun jika user input jenis kendaraan selain tiga itu, bagian `default` pada `switch` akan dijalankan, program menampilkan pesan `fmt.Println("Jenis kendaraan tidak valid")` lalu langsung menghentikan program dengan `return`, sehingga tidak dilanjutkan. Jika jenis kendaraan valid, program kemudian menghitung biaya total parkir menggunakan perintah `total := tarif * durasi`, yaitu mengalikan tarif per jam dengan jumlah jam parkir yang sudah dimasukkan sebelumnya. Setelah total biaya dihitung, program menampilkan hasilnya lewat

kalimat `fmt.Println("Total biaya parkir: Rp", total)`, sehingga user bisa melihat berapa biaya parkir yang harus dibayar.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var bilangan int

    fmt.Print("Masukan satu bilangan bulat : ")
    fmt.Scan(&bilangan)

    switch{
        case bilangan % 10 == 0:
            hasil := bilangan / 10
            fmt.Println("Kategori: Bilangan kelipatan
10")
            fmt.Printf("Hasil pembagian antara %d / 10 =
%d\n", bilangan, hasil)
        case bilangan % 5 == 0 && bilangan != 5:
            hasil := bilangan * bilangan
            fmt.Println("Kategori: Bilangan kelipatan
5")
            fmt.Printf("Hasil kuadrat dari %d ^2 =
%d\n", bilangan, hasil)
        case bilangan % 2 ==0:
            a := bilangan + 1
            hasil := bilangan * a
            fmt.Println("Kategori: Bilangan genap")
            fmt.Printf("Hasil perkalian dengan bilangan
berikutnya %d * %d = %d\n", bilangan, a, hasil)
        default:
            a := bilangan + 1
            hasil := bilangan + a
            fmt.Println("Kategori: Bilangan ganjil")
```

```
    fmt.Printf("Hasil penjumlahan dengan  
bilangan berikutnya %d + %d = %d\n", bilangan, a,  
hasil)  
  
}  
  
}
```

Screenshot program

```
go guided1.go 5 go guided2.go 1 go guided3.go 1 go soal1.go 1 go soal2.go 1 go soal3.go 1 x

soal3.go > main
1 package main
2 import "fmt"
3 func main() {
4     var bilangan int
5
6     fmt.Println("Masukan satu bilangan bulat : ")
7     fmt.Scan(&bilangan)
8
9     switch{
10     case bilangan % 10 == 0:
11         hasil := bilangan / 10
12         fmt.Println("Kategori: Bilangan kelipatan 10")
13         fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", bilangan, hasil)
14     case bilangan % 5 == 0 && bilangan != 5:
15         hasil := bilangan * bilangan
16         fmt.Println("Kategori: Bilangan kelipatan 5")
17         fmt.Printf("Hasil kuadrat dari %d ^ 2 = %d\n", bilangan, hasil)
18     case bilangan % 2 == 0:
19         a := bilangan + 1
20         hasil := bilangan * a
21         fmt.Println("Kategori: Bilangan genap")
22         fmt.Printf("Hasil perkalian dengan bilangan berikunya %d * %d = %d\n", bilangan, a, hasil)
23     default:
24         a := bilangan + 1
25         hasil := bilangan + a
26         fmt.Println("Kategori: Bilangan ganjil")
27         fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n", bilangan, a, hasil)
}

```

Nama: Nabyla Zahirah Ramadhan
NIM: 109082500104
Kelas: S1IF-13-07

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS EXPLORER

nabylaaaa@nabylas-MacBook-Air week 11 % go run "/Users/nabylaaaa/Documents/go/week 11/soal3.go"
Kategori: Bilangan ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
nabylaaaa@nabylas-MacBook-Air week 11 % go run "/Users/nabylaaaa/Documents/go/week 11/soal3.go"
Masukan satu bilangan bulat : 8
Kategori: Bilangan genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
nabylaaaa@nabylas-MacBook-Air week 11 % go run "/Users/nabylaaaa/Documents/go/week 11/soal3.go"
Masukan satu bilangan bulat : 25
Kategori: Bilangan kelipatan 5
Hasil kuadrat dari 25 ^2 = 625
nabylaaaa@nabylas-MacBook-Air week 11 % go run "/Users/nabylaaaa/Documents/go/week 11/soal3.go"
Masukan satu bilangan bulat : 20
Kategori: Bilangan kelipatan 10
Hasil pembagian antara 20 / 10 = 2
nabylaaaa@nabylas-MacBook-Air week 11 %

Deskripsi program

Program untuk mengidentifikasi pola arimatika, variabelnya bilangan yang bertipe int. Pertama program meminta user menginput bilangan menggunakan `fmt.Print("Masukan satu bilangan bulat : ")`, dan akan dibaca menggunakan `fmt.Scan(&bilangan)`. Begitu angkanya sudah masuk, program memakai struktur switch tanpa kondisi utama. Ini artinya setiap case di dalamnya adalah syarat yang langsung dicek satu per satu dari atas sampai salah satunya cocok. Kondisi pertama yang diperiksa adalah `bilangan % 10 == 0`, yang artinya angka tersebut habis dibagi 10. Jika benar, program langsung menghitung hasil pembagiannya menggunakan `hasil := bilangan / 10`, lalu menampilkan kategori “Bilangan kelipatan 10” dan hasil pembagian tersebut lewat `fmt.Printf()`. Kalau syarat pertama tidak terpenuhi, program melihat apakah bilangan itu kelipatan 5 tapi bukan angka 5 lewat `bilangan % 5 == 0 && bilangan != 5`. Pada bagian ini, program menghitung kuadrat dari bilangan tersebut (`bilangan * bilangan`) dan menampilkan kategori serta hasil perhitungannya.

Jika bukan kelipatan 10 maupun kelipatan 5, case berikutnya mengecek apakah bilangan tersebut genap dengan $\text{bilangan \% } 2 == 0$. Bila iya, program menentukan angka setelahnya ($\text{bilangan} + 1$) lalu mengalikan keduanya. Hasil perkalian itu kemudian dicetak bersama informasi bahwa angka tersebut termasuk bilangan genap. Kalau tidak masuk semua kategori di atas, case terakhir yaitu default yang akan dijalankan. Ini berarti bilangan pasti ganjil. Di bagian ini program juga mengambil angka berikutnya lalu menjumlahkan keduanya, kemudian menuliskan kategori “Bilangan ganjil” beserta hasil penjumlahannya.