

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 11
Switch Case



Disusun oleh:
M.Afriza marantika
103112430271
S1IF-12-07

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

DASAR TEORI

Switch Case adalah sebuah struktur kontrol yang digunakan dalam pemrograman untuk memilih salah satu dari beberapa jalur eksekusi berdasarkan nilai dari sebuah ekspresi. Struktur ini lebih efisien dibandingkan dengan menggunakan banyak pernyataan if-else yang berulang ketika memeriksa banyak kondisi.

TUGAS PENDAHULUAN

1. Tugas 1

Pseudo code

Function kategoriUmur(umur):

```
IF umur == 0 THEN
    RETURN "Tidak Terdefinisi"
ELSE IF umur < 13 THEN
    RETURN "Anak-Anak"
ELSE IF umur < 20 THEN
    RETURN "Remaja"
ELSE IF umur < 60 THEN
    RETURN "Dewasa"
ELSE IF umur >= 60 THEN
    RETURN "Lansia"
ELSE
    RETURN "Data tidak valid"
```

Main Program:

```
DECLARE umur AS Integer

PRINT "Masukkan umur: "
READ umur FROM input

CALL kategoriUmur(umur) AND PRINT "Termasuk dalam kategori: " + result of kategoriUmur
```

Source code

```
package main
```

```
import "fmt"

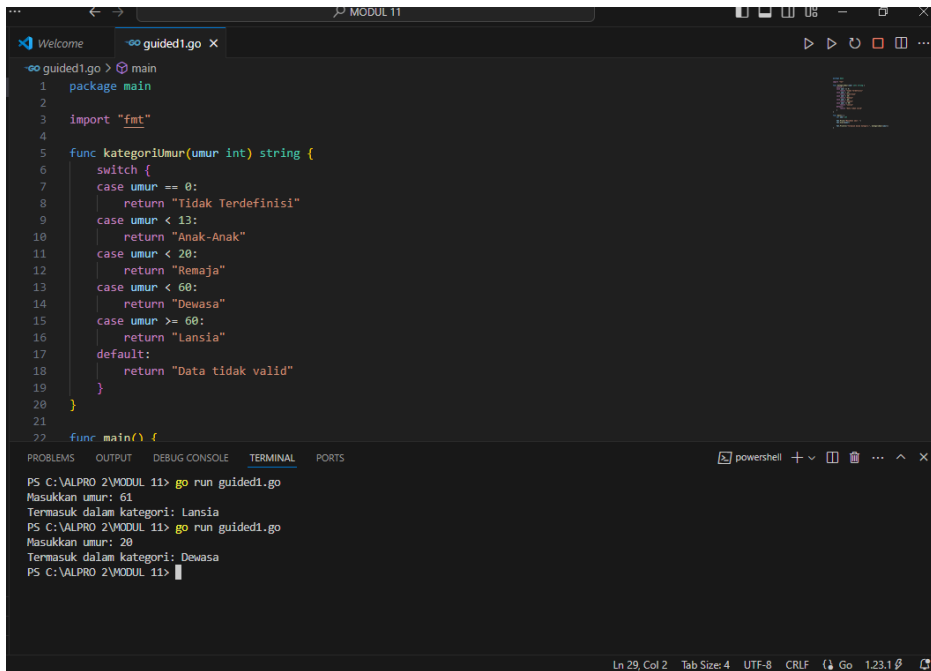
func kategoriUmur(umur int) string {
    switch {
    case umur == 0:
        return "Tidak Terdefinisi"
    case umur < 13:
        return "Anak-Anak"
    case umur < 20:
        return "Remaja"
    case umur < 60:
        return "Dewasa"
    case umur >= 60:
        return "Lansia"
    default:
        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



```
1 package main
2
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     // ...
24 }
```

```
PS C:\VALPRO\2\MODUL 11> go run guided1.go
Masukkan umur: 61
Termasuk dalam kategori: Lansia
PS C:\VALPRO\2\MODUL 11> go run guided1.go
Masukkan umur: 20
Termasuk dalam kategori: Dewasa
PS C:\VALPRO\2\MODUL 11>
```

Deskripsi program

Program ini bertujuan untuk mengklasifikasikan seseorang ke dalam kategori umur berdasarkan input yang diberikan oleh pengguna. Program menggunakan fungsi `kategoriUmur` yang akan menerima input berupa umur (dalam bentuk angka) dan mengembalikan kategori umur yang sesuai. Kategori yang tersedia antara lain: "Tidak Terdefinisi", "Anak-Anak", "Remaja", "Dewasa", dan "Lansia".

2. Tugas 2

Pseudo code

Function `getHari(kode)`:

```
IF kode == 1 THEN
    RETURN "Senin"
ELSE IF kode == 2 THEN
    RETURN "Selasa"
ELSE IF kode == 3 THEN
    RETURN "Rabu"
ELSE IF kode == 4 THEN
    RETURN "Kamis"
ELSE IF kode == 5 THEN
    RETURN "Jumat"
ELSE IF kode == 6 THEN
    RETURN "Sabtu"
ELSE IF kode == 7 THEN
```

```
    RETURN "Minggu"
```

```
ELSE
```

```
    RETURN "Kode Tidak Ditemukan"
```

Main Program:

```
    DECLARE kode AS Integer
```

```
    PRINT "Masukkan kode hari: "
```

```
    READ kode FROM input
```

```
    CALL getHari(kode) AND STORE RESULT IN hari
```

```
    PRINT "Hari: " + hari
```

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

```

1 package main
2
3 import "fmt"
4 func getHari(kode int) string {
5     switch kode {
6     case 1:
7         return "Senin"
8     case 2:
9         return "Selasa"
10    case 3:
11        return "Rabu"
12    case 4:
13        return "Kamis"
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 }

```

```

PS C:\VALPRO 2\MODUL 11> go run guided2.go
Masukkan kode hari: 5
Hari: Jumat
PS C:\VALPRO 2\MODUL 11> go run guided2.go
Masukkan kode hari: 6
Hari: Sabtu
PS C:\VALPRO 2\MODUL 11>

```

Deskripsi program

Program ini bertujuan untuk mengonversi kode angka (1 - 7) menjadi nama hari dalam seminggu. Program diatas meminta pengguna untuk memasukkan kode angka yang mewakili salah satu hari dalam seminggu, lalu menampilkan nama hari yang sesuai dengan kode tersebut.

3. Tugas 3

Pseudo code

Function deskripsiFilm(kode):

```
IF kode == "G" THEN
    RETURN "untuk semua umur"
ELSE IF kode == "PG" THEN
    RETURN "untuk anak-anak di atas 7 tahun"
ELSE IF kode == "PG-13" THEN
    RETURN "untuk remaja di atas 13 tahun"
ELSE IF kode == "R" THEN
    RETURN "untuk dewasa"
ELSE
    RETURN "kode tidak ditemukan"
```

Main Program:

```
DECLARE kode AS String

PRINT "Masukkan kode film: "
READ kode FROM input

CALL deskripsiFilm(kode) AND STORE RESULT IN kategori

PRINT "Kategori: " + kategori
```

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

```

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 }

```

```

PS C:\VALPRO 2\MODUL 11> go run guided3.go
Masukkan kode film: G
Kategori: untuk semua umur
PS C:\VALPRO 2\MODUL 11> go run guided3.go
Masukkan kode film: R
Kategori: untuk dewasa
PS C:\VALPRO 2\MODUL 11>

```

Deskripsi program

Program diatas untuk memberikan deskripsi kategori film berdasarkan kode rating film yang dimasukkan oleh pengguna. Program meminta input kode rating film berupa string (misalnya "G", "PG", "PG-13", atau "R"), dan kemudian menampilkan kategori usia yang sesuai berdasarkan kode tersebut.

LATIHAN MODUL

1. Tugas 1

Pseudo code

```
Function cekKelayakanAir(ph):  
    IF ph >= 6.5 AND ph <= 8.6 THEN  
        RETURN "Air layak diminum"  
    ELSE IF (ph < 6.5 AND ph > 0) OR (ph > 8.6 AND ph <= 14) THEN  
        RETURN "Air tidak layak diminum"  
    ELSE IF ph > 14 OR ph < 0 THEN  
        RETURN "Inputan tidak valid, harus antara 0-14"  
    ELSE  
        RETURN "Inputan tidak valid"
```

Main Program:

```
    DECLARE ph AS Float  
  
    PRINT "Masukkan kadar pH: "  
    READ ph FROM input  
  
    CALL cekKelayakanAir(ph) AND PRINT RESULT
```

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

```

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 }

```

```

PS C:\VALPRO 2\MODUL 11> go run unguided1.go
Masukkan kadar pH: 8.6
Air layak diminum
PS C:\VALPRO 2\MODUL 11> go run unguided1.go
CreateFile unguided1.go: The system cannot find the file specified.
PS C:\VALPRO 2\MODUL 11> go run unguided1.go
Masukkan kadar pH: 9
Air tidak layak diminum
PS C:\VALPRO 2\MODUL 11>

```

Deskripsi program

Program diatas untuk mengevaluasi kelayakan air berdasarkan kadar pH yang dimasukkan oleh pengguna. Program ini akan mengklasifikasikan air apakah layak diminum atau tidak berdasarkan nilai pH yang diberikan. Selain itu, program juga akan memverifikasi apakah input pH yang dimasukkan berada dalam rentang yang valid (antara 0 hingga 14).

2. Tugas 2

Pseudo code

Function getTarifPerJam(jenisKendaraan):

```
IF jenisKendaraan == "motor" THEN
    RETURN 2000, NULL
ELSE IF jenisKendaraan == "mobil" THEN
    RETURN 5000, NULL
ELSE IF jenisKendaraan == "truk" THEN
    RETURN 8000, NULL
ELSE
    RETURN 0, ERROR "jenis kendaraan tidak valid"
```

Function hitungBiayaParkir(tarifPerJam, durasiParkir):

```
IF durasiParkir < 1 THEN
    SET durasiParkir = 1
END IF
RETURN tarifPerJam * durasiParkir
```

Main Program:

```
DECLARE jenisKendaraan AS String
DECLARE durasiParkir AS Integer
DECLARE tarifPerJam AS Integer
DECLARE totalBiaya AS Integer
DECLARE err AS Error

PRINT "Masukkan jenis kendaraan (motor, mobil, truk): "
READ jenisKendaraan FROM input

PRINT "Masukkan durasi parkir (dalam jam): "
READ durasiParkir FROM input

CALL getTarifPerJam(jenisKendaraan) AND STORE RESULT IN tarifPerJam, err

IF err IS NOT NULL THEN
    PRINT err MESSAGE
    EXIT PROGRAM
```

END IF

CALL hitungBiayaParkir(tarifPerJam, durasiParkir) AND STORE RESULT IN totalBiaya

PRINT "Total biaya parkir: Rp " + totalBiaya

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

    }

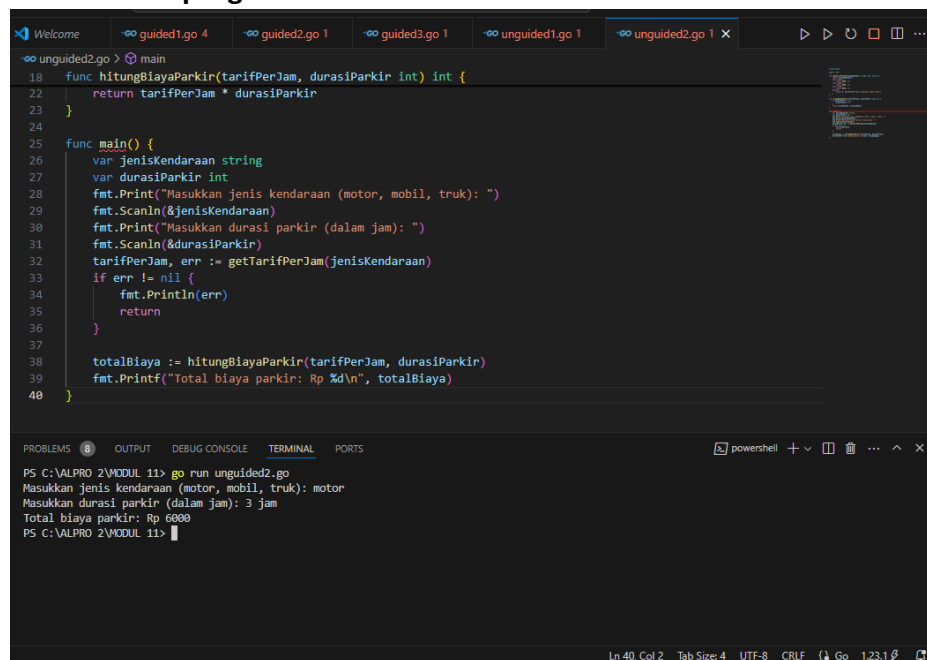
    totalBiaya := hitungBiayaParkir(tarifPerJam, durasiParkir)

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

}

```

Screenshoot program



```

Welcome | unguided2.go 4 | unguided2.go 1 | unguided3.go 1 | unguided1.go 1 | unguided2.go 1 X
- unguided2.go > main
18 func hitungBiayaParkir(tarifPerJam, durasiParkir int) int {
22     return tarifPerJam * durasiParkir
23 }
24
25 func main() {
26     var jenisKendaraan string
27     var durasiParkir int
28     fmt.Print("Masukkan jenis kendaraan (motor, mobil, truk): ")
29     fmt.Scanln(&jenisKendaraan)
30     fmt.Print("Masukkan durasi parkir (dalam jam): ")
31     fmt.Scanln(&durasiParkir)
32     tarifPerJam, err := getTarifPerJam(jenisKendaraan)
33     if err != nil {
34         fmt.Println(err)
35         return
36     }
37
38     totalBiaya := hitungBiayaParkir(tarifPerJam, durasiParkir)
39     fmt.Printf("Total biaya parkir: Rp %d\n", totalBiaya)
40 }

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS
PS C:\VALPRO 2\MODUL 11> go run unguided2.go
Masukkan jenis kendaraan (motor, mobil, truk): motor
Masukkan durasi parkir (dalam jam): 3 jam
Total biaya parkir: Rp 6000
PS C:\VALPRO 2\MODUL 11>

```

Deskripsi program

Program diatas untuk menghitung biaya parkir berdasarkan jenis kendaraan dan durasi parkir yang dimasukkan oleh pengguna. Jenis kendaraan yang valid adalah "motor", "mobil", dan "truk", dengan tarif parkir per jam yang berbeda untuk masing-masing. Jika jenis kendaraan yang dimasukkan tidak valid, program akan menampilkan pesan error dan menghentikan eksekusi. Program ini juga memastikan bahwa durasi parkir tidak kurang dari satu jam. Jika durasi parkir yang dimasukkan kurang dari satu jam, program akan otomatis mengubahnya menjadi satu jam.

3. Tugas 3

Pseudo code

Function cekKelipatan10(angka):

IF angka is divisible by 10 AND angka is greater than 10 THEN

RETURN TRUE, angka divided by 10

END IF

RETURN FALSE, 0

Function cekKelipatan5(angka):

IF angka is divisible by 5 AND angka is greater than 5 THEN

RETURN TRUE, angka squared (angka * angka)

END IF

RETURN FALSE, 0

Function cekGanjil(angka):

IF angka is odd (not divisible by 2) THEN

angka2 = angka + 1

RETURN TRUE, angka, angka + angka2

END IF

RETURN FALSE, 0, 0

Function cekGenap(angka):

IF angka is even (divisible by 2) THEN

angka2 = angka + 1

RETURN TRUE, angka, angka * angka2

END IF

RETURN FALSE, 0, 0

Main Program:

DECLARE angka AS Integer

PRINT "Masukkan angka: "

READ angka FROM input

CALL cekKelipatan10(angka) AND STORE result in ok, hasil

```

IF ok is TRUE THEN
    PRINT "Kategori = Bilangan kelipatan 10"
    PRINT "Hasil pembagian berikutnya angka / 10 = hasil"
    EXIT PROGRAM
END IF

CALL cekKelipatan5(angka) AND STORE result in ok, hasil
IF ok is TRUE THEN
    PRINT "Kategori = Bilangan kelipatan 5"
    PRINT "Hasil kuadrat berikutnya angka^2 = hasil"
    EXIT PROGRAM
END IF

CALL cekGanjil(angka) AND STORE result in ok, angka1, hasil
IF ok is TRUE THEN
    PRINT "Kategori = Bilangan ganjil"
    PRINT "Hasil penjumlahan bilangan berikutnya angka1 + angka2 = hasil"
    EXIT PROGRAM
END IF

CALL cekGenap(angka) AND STORE result in ok, angka1, hasil
IF ok is TRUE THEN
    PRINT "Kategori = Bilangan genap"
    PRINT "Hasil perkalian bilangan berikutnya angka1 * angka2 = hasil"
    EXIT PROGRAM
END IF

```

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

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

```

34 func main() {
35     fmt.Scan(&angka)
36
37     if ok, hasil := cekKelipatan10(angka); ok {
38         fmt.Println("Kategori = Bilangan kelipatan 10")
39         fmt.Printf("Hasil pembagian berikutnya %d / 10 = %d\n", angka, hasil)
40     } else if ok, hasil := cekKelipatan5(angka); ok {
41         fmt.Println("Kategori = Bilangan kelipatan 5")
42         fmt.Printf("Hasil kuadrat berikutnya %d^2 = %d\n", angka, hasil)
43     } else if ok, angka1, hasil := cekGanjil(angka); ok {
44         fmt.Println("Kategori = Bilangan ganjil")
45         fmt.Printf("Hasil penjumlahan bilangan berikutnya %d + %d = %d\n", angka1, angka1+1, hasil)
46     } else if ok, angka1, hasil := cekGenap(angka); ok {
47         fmt.Println("Kategori = Bilangan genap")
48         fmt.Printf("Hasil perkalian bilangan berikutnya %d * %d = %d\n", angka1, angka1+1, hasil)
49     }
50 }
51
52

```

The terminal output shows the program's execution:

```

PS C:\VALPRO 2\MODUL 11> go unguided3.go
go unguided3.go: unknown command
Run 'go help' for usage.
PS C:\VALPRO 2\MODUL 11> go run unguided3.go
Masukkan angka: 5
Kategori = Bilangan ganjil
Hasil penjumlahan bilangan berikutnya 5 + 6 = 11
PS C:\VALPRO 2\MODUL 11> go run unguided3.go
Masukkan angka: 8
Kategori = Bilangan genap
Hasil perkalian bilangan berikutnya 8 * 9 = 72
PS C:\VALPRO 2\MODUL 11>

```

Deskripsi program

Program diatas bertujuan untuk mengklasifikasikan angka yang dimasukkan oleh pengguna ke dalam kategori tertentu, yaitu: bilangan kelipatan 10, bilangan kelipatan 5, bilangan ganjil, atau bilangan genap. Berdasarkan kategori yang ditemukan, program akan melakukan perhitungan tambahan dan menampilkan hasilnya.