

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 11
“SWITCH-CASE”



DISUSUN OLEH:
ANASTASIA ADINDA NARENDRA INDRIANTO
103112400085
S1 IF-12-01

DOSEN:
Yohani Setiya Rafika Nur, M. Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024/2025

DASAR TEORI

1. Konsep Input dan Output

Input merupakan intruksi dasar untuk membaca data yang diberikan dari pengguna. Data yang diberikan oleh pengguna akan disimpan ke dalam suatu wadah yang disebut *variable*. Penulisan intruksi input beragam menyesuaikan Bahasa pemrograman yang digunakan. Output merupakan perintah untuk menampilkan data ke layar monitor. Data yang sudah diproses atau diolah oleh program komputer perlu ditampilkan ke layar sehingga pengguna bisa memperoleh informasi dari hasil pengolahan data yang dilakukan program.

2. Konsep Data, Variabel, dan Intruksi Dasar

Variabel adalah nama dari suatu lokasi di memori, yang data dengan tipe tertentu dapat disimpan. Nama variabel dimulai dengan huruf dan dapat diikuti dengan sejumlah huruf, angka atau garisbawah. Tipe data yang umum adalah integer, real, Boolean, karakter dan string. Nilai data yang tersimpan dalam variabel dapat diperoleh dengan menyebutkan langsung nama variabelnya. Informasi Alamat atau Lokasi variabel dapat diperoleh dengan menambahkan prefix dan di depan nama variabel tersebut.

3. Konsep Bahasa Pemrograman Go

Bahasa Go menganut kesesuaian tipe data yang ketat. Tipe data yang berbeda tidak boleh dicampur dalam satu ekspresi, bahkan tipe data masih yang sejenis. Menyesuaikan tipe data ada beberapa cara yaitu *casting tipe (data)* mengubah tipe dari data yang diberikan ke tipe data yang diinginkan, memanfaatkan fungsi *Sprint* dan *Sscan* dari paket *fmt*, dan memanfaatkan fungsi-fungsi dalam paket *strconv*, seperti *Atoi*, *Itoa* dan *ParseBool*. Variabel harus dideklarasikan terlebih dahulu sebelum digunakan. Variabel juga harus diinisialisasi dulu agar nilai yang tersimpan diketahui dengan jelas dan eksekusi algoritma menjadi terprediksi. Dalam Bahasa Go, variabel yang tidak diinisialisasi lebih dahulu otomatis diisi dengan nilai default ekuivalen dengan bit 0.

4. Konsep Tipe Data

Melalui ilmu komputer tentunya kita sering mendengar istilah data. Data terbagi menjadi bermacam-macam tipe data yang terklasifikasi dan memiliki fungsi sendiri. Tentunya, berbagai macam tipe data tersebut sangat bermanfaat bagi kinerja komputer melalui kode-kode dalam bahasa pemrograman. Data types atau tipe data adalah sebuah pengklasifikasian data berdasarkan jenis data tersebut. Tipe data dibutuhkan agar kompiler dapat mengetahui bagaimana sebuah data akan digunakan. Untuk mengembangkan sebuah program ada beberapa tipe data yang dibutuhkan. Tipe data memiliki 4 data type yang sering digunakan yaitu,

- 1) Bilangan Bulat (Integer) Tipe bilangan bulat (Integer) adalah tipe data numerik yang biasa digunakan apabila bertemu dengan bilangan bulat, seperti 1, 27, 100, dll. Bilangan ini juga mengenal nilai positif dan negatif. Tipe data numerik yang termasuk ke dalam bilangan bulat adalah *byte*, *short*, *int*, dan *long*.
- 2) Bilangan Pecahan (Floating Point) Tipe bilangan pecahan atau *floating point* adalah bilangan yang menangani bilangan desimal atau perhitungan secara detail. Karena kemampuannya, float point berbanding terbalik dengan integer. Terdapat dua tipe pada bilangan pecahan ini yaitu *float* dan *double*.
- 3) Karakter (Char)
Tipe data karakter tunggal yang biasa didefinisikan dengan tanda petik (') di awal dan di akhir karakternya. Tipe ini mengikuti aturan “unicode” sehingga bilangan harus

diawali kode “/u”. Tetapi juga biasa menggunakan bilangan heksadesimal dari 0000 sampai FFFF.

4) Boolean

Tipe data *boolean* merupakan tipe yang memiliki dua nilai yaitu benar (*true*) atau salah (*false*). Nilai yang digunakan pada tipe ini sangat penting dalam mengambil keputusan suatu kejadian tertentu.

5. **Deklarasi dan Inisialisasi**

Deklarasi variabel dalam bahasa pemrograman komputer adalah pernyataan yang digunakan untuk menentukan nama variabel dan tipe datanya. Deklarasi memberi tahu kompiler tentang keberadaan entitas dalam program dan lokasinya. Saat Anda mendeklarasikan variabel, Anda juga harus menginisialisasinya. Inisialisasi adalah proses pemberian nilai pada Variabel. Setiap bahasa pemrograman memiliki metode tersendiri untuk menginisialisasi variabel. Jika nilai tidak diberikan pada Variabel, maka proses tersebut hanya disebut Deklarasi.

6. **Konstanta dalam Bahasa Pemrograman**

Konstanta adalah entitas yang merujuk pada nilai data yang tetap dan tidak dapat diubah. Selama eksekusi/perhitungan pemrograman, nilai suatu konstanta tidak dapat diubah, tetap konstan. Jenis Konstanta ada konstanta Bilangan Bulat. Konstanta bilangan bulat adalah barisan bilangan bulat yang nilainya tetap. Mereka tidak boleh mengandung koma desimal atau angka pecahan. Konstanta bilangan bulat dapat berupa bilangan positif atau negatif. Mereka termasuk bilangan bulat sistem desimal, bilangan bulat sistem oktal, bilangan bulat sistem heksadesimal.

7. **Integer Division dan Modulo**

Pembagian pada tipe data integer (**integer division** atau **div**) sedikit berbeda dengan pembagian yang sudah sering kita pelajari dari sekolah dasar. Hasil pembagian akan bertipe data integer, artinya kita akan mengabaikan bilangan yang muncul setelah tanda koma atau floating point. Hasil dari operasi div ini dikenal dengan istilah **quotient**. Modulo atau modulus "mod" adalah operasi untuk mencari nilai integer dari sisa pembagian pada integer division, biasanya dikenal juga dengan istilah remainder. Misalnya untuk operasi 10 dibagi dengan 3, maka sisa pembagian adalah 1. Operator modulo menggunakan "%" di dalam bahasa pemrograman Go.

8. **Casting atau Konversi Tipe Data**

Pada Bahasa pemrograman Go, tipe data bersifat statis, artinya tipe data yang sudah didefinisikan tidak dapat diganti selama program berjalan. Casting merupakan salah satu Teknik konversi antar tipe data di dalam Bahasa pemrograman.

9. **Paradigma Perulangan**

Perulangan merupakan salah satu struktur kontrol yang memungkinkan suatu instruksi yang sama dilakukan berulang kali dalam waktu atau jumlah yang lama. Tanpa instruksi perulangan, maka suatu instruksi akan ditulis dalam jumlah yang sangat banyak. Salah satu instruksi perulangan yang paling mudah adalah **for-loop**, yang mana dengan instruksi ini dapat digunakan untuk mengulangi instruksi sebanyak **n** kali (iterasi). Batasan besar nilai dari **n** menyesuaikan dengan batasan dari tipe data integer yang digunakan.

10. **Paradigma Percabangan if-else**

Percabangan if-else adalah cara untuk menjalankan kode yang berbeda tergantung pada apakah kondisi yang diberikan bernilai benar (*true*) atau salah (*false*). Bentuk paling sederhana dari percabangan ini adalah dengan menggunakan if untuk memeriksa kondisi, artinya kode program dari baris ke-1 hingga baris terakhir akan dieksekusi satu persatu.

11. Pengertian Switch-Case

Switch case adalah pernyataan pilihan ganda dalam bahasa pemrograman yang memungkinkan nilai variabel atau ekspresi untuk mengubah aliran kontrol eksekusi program. Penggunaan if-else dan switch-case bergantung pada kebutuhan Anda. if-else digunakan ketika Anda memiliki beberapa kondisi yang berbeda, sementara switch-case digunakan ketika Anda memiliki banyak pilihan yang mungkin dengan variabel yang sama.

12. Karakteristik Switch-Case

Pada dasarnya switch-case merupakan alternatif dari penggunaan else-if, tetapi pemilihan aksi tidak dilakukan berdasarkan kondisi, tetapi suatu nilai atau operasi tipe data dasar yang tidak menghasilkan boolean.

Penulisan switch case terdiri dari komponen berikut:

- a. Ekspresi, merupakan operasi tipe data yang menghasilkan nilai selain tipe data boolean,
- b. Nilai, pilihan case dari ekspresi yang dideklarasikan pada instruksi switch-case.
- c. Aksi, merupakan kumpulan instruksi yang akan dieksekusi sesuai dengan nilai yang dihasilkan dari ekspresi yang terdapat pada switch-case. Aksi lain hanya akan dieksekusi apabila tidak ada pilihan nilai yang sesuai dengan hasil ekspresi.

CONTOH SOAL

1. Contoh Latihan Soal 1

Source Code:

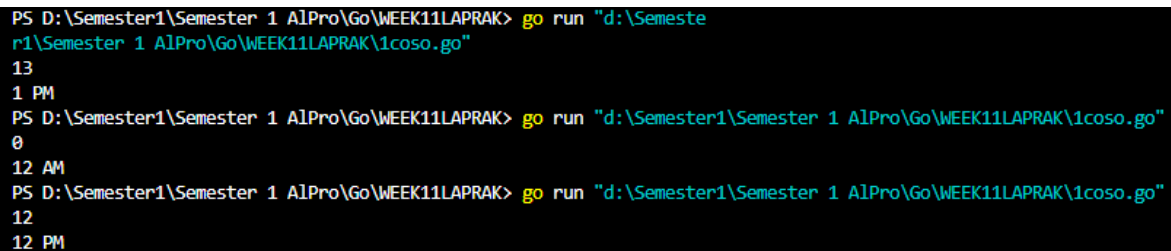
```
package main

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)
}
```

Output:



```
PS D:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK> go run "d:\Semeste
r1\Semester 1 APro\Go\WEEK11LAPRAK\1coso.go"
13
1 PM
PS D:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK\1coso.go"
0
12 AM
PS D:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK\1coso.go"
12
12 PM
```

Deskripsi Program:

Program 1coso.go dibuat untuk membuat sebuah algoritma digunakan untuk melakukan konversi waktu dari bentuk 24 jam ke dalam bentuk 12 jam.

Masukan terdiri dari satu bilangan bulat dengan interval 0 s.d. 23 (inklusif) yang merepresentasikan jam dalam bentuk 24 jam.

Keluaran terdiri dari teks yang menyatakan format jam pada masukan dalam bentuk 12 jam.. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

No	Masukan	Keluaran
1	13	1 PM
2	0	12 AM

3	12	12 PM
---	----	-------

2. Contoh Latihan Soal 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("Bukan Asli Indonesia")
    default:
        fmt.Println("Tidak Termasuk Tanaman Karnivora")
    }
}
```

Output:

```
PS D:\Semester1\Semester 1 ALPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 ALPro\Go\WEEK11LAPRAK\2coso.go"
nepenthes
Termasuk Tanaman Karnivora
Asli Indonesia
PS D:\Semester1\Semester 1 ALPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 ALPro\Go\WEEK11LAPRAK\2coso.go"
venus
Termasuk Tanaman Karnivora
Bukan Asli Indonesia
PS D:\Semester1\Semester 1 ALPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 ALPro\Go\WEEK11LAPRAK\2coso.go"
karedok
Tidak Termasuk Tanaman Karnivora
```

Deskripsi Program:

Program 2coso.go dibuat untuk menentukan apakah tanaman tersebut termasuk tanaman karnivora atau tidak. Jika ya, apakah tanaman tersebut asli indonesia atau tidak.

Masukan terdiri dari satu string, yakni nama tanaman tersebut.

Keluaran berupa teks "Termasuk Tanaman Karnivora" atau "Tidak termasuk Tanaman Karnivora", dan dilanjutkan "Asli Indonesia" atau "Bukan Asli Indonesia". Keluaran sesuai dengan masukan yang diberikan. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan output;

No	Masukan	Keluaran
1	nepenthes	Termasuk Tanaman Karnivora Asli Indonesia
2	venus	Termasuk Tanaman Karnivora Bukan Asli Indonesia
3	karedok	Tidak termasuk Tanaman Karnivora

3. Contoh Latihan Soal 3

Source Code:

```
package main

import (
    "fmt"
)

func main() {
    var tipe_kendaraan string
    var durasi, tarif int

    fmt.Print("Masukan jenis kendaraan (Motor/Mobil/Truk) : ")
    fmt.Scan(&tipe_kendaraan)
    fmt.Print("Masukan durasi parkir (dalam jam) : ")
    fmt.Scan(&durasi)

    switch {
    case tipe_kendaraan == "Motor" && durasi >= 1 && durasi <= 2:
        tarif = 7000
    case tipe_kendaraan == "Motor" && durasi > 2:
        tarif = 9000
    case tipe_kendaraan == "Mobil" && durasi >= 1 && durasi <= 2:
        tarif = 15000
    case tipe_kendaraan == "Mobil" && durasi > 2:
        tarif = 20000
    case tipe_kendaraan == "Truk" && durasi >= 1 && durasi <= 2:
        tarif = 25000
    case tipe_kendaraan == "Truk" && durasi > 2:
        tarif = 35000
    default:
        fmt.Println("Jenis kendaraan atau durasi parkir tidak valid")
    }
    fmt.Printf("Tarif Parkir : Rp %d", tarif)
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\3coso.go"
Masukan jenis kendaraan (Motor/Mobil/Truk) : Motor
Masukan durasi parkir (dalam jam) : 2
Tarif Parkir : Rp 7000
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\3coso.go"
Masukan jenis kendaraan (Motor/Mobil/Truk) : Mobil
Masukan durasi parkir (dalam jam) : 4
Tarif Parkir : Rp 20000
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\3coso.go"
Masukan jenis kendaraan (Motor/Mobil/Truk) : Truk
Masukan durasi parkir (dalam jam) : 1
Tarif Parkir : Rp 25000
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\3coso.go"
Masukan jenis kendaraan (Motor/Mobil/Truk) : Sepeda
Masukan durasi parkir (dalam jam) : 2
Jenis kendaraan atau durasi parkir tidak valid
Tarif Parkir : Rp 0
```

Deskripsi Program:

Program 3coso.go dibuat untuk menentukan tarif parkir berdasarkan jenis kendaraan dan durasi parkir. Terdapat tiga jenis kendaraan yang dapat diparkir dengan tarif berbeda berdasarkan durasi parkir. Untuk motor, tarifnya adalah Rp 7.000 untuk durasi 1-2 jam, dan Rp 9.000 jika durasi lebih dari 2 jam. Mobil dikenakan tarif Rp 15.000 untuk 1-2 jam, dan Rp 20.000 untuk durasi lebih dari 2 jam. Sementara itu, truk dikenakan tarif Rp 25.000 untuk 1-2 jam, dan Rp 35.000 jika lebih dari 2 jam.

Masukan terdiri dari 2 baris, baris pertama adalah string yang menyatakan jenis kendaraan sedangkan baris kedua merupakan bilangan bulat yang menyatakan durasi parkir dalam jam.

Keluaran berupa string yang menampilkan tarif parkir berdasarkan jenis kendaraan dan durasi parkir kendaraan atau "Jenis kendaraan atau durasi parkir tidak valid" jika jenis kendaraan selain dari motor, mobil dan truk. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan output;

No	Masukan	Keluaran
1	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Motor</u> Masukkan durasi parkir (dalam jam): <u>2</u>	Tarif Parkir: Rp 7000
2	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Mobil</u> Masukkan durasi parkir (dalam jam): <u>4</u>	Tarif Parkir: Rp 20000
3	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Motor</u> Masukkan durasi parkir (dalam jam): <u>3</u>	Tarif Parkir: Rp 9000
4	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Truk</u> Masukkan durasi parkir (dalam jam): <u>1</u>	Tarif Parkir: Rp 25000
5	Masukkan jenis kendaraan (Motor/Mobil/Truk): <u>Sepeda</u> Masukkan durasi parkir (dalam jam): <u>2</u>	Jenis kendaraan atau durasi parkir tidak valid Tarif Parkir: Rp 0

SOAL LATIHAN

1. Latihan Soal 1

Source Code:

```
package main

import "fmt"

func main() {
    var pH float64
    fmt.Print("Masukan Nilai pH air : ")
    fmt.Scan(&pH)

    switch {
    case pH < 0 || pH > 14:
        fmt.Println("Nilai ph tidak valid, masukan antara 0-14")
    case pH >= 6.5 && pH <= 8.6:
        fmt.Println("Air layak minum")
    default:
        fmt.Println("Air tidak layak minum")
    }
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\1Latsol.go"
Masukan Nilai pH air : 8.6
Air layak minum
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\1Latsol.go"
Masukan Nilai pH air : 9
Air tidak layak minum
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\1Latsol.go"
Masukan Nilai pH air : 16
Nilai ph tidak valid, masukan antara 0-14
```

Deskripsi Program:

Program Soal1.go dibuat dengan tujuan menentukan apakah kadar pH pada air yang diinput termasuk Air yang layak untuk diminum atau tidak.

Masukan terdiri dari satu float, yakni kadar pH tersebut.

Keluaran berupa teks seperti contoh "Air Layak Minum" (ph >= 6.5 dan ph <= 8.6) atau "Air Tidak Layak Minum" (ph < 6.5 dan ph > 8.6), dan "Input tidak valid, rentang pH 0 - 14" jika nilai float diatas 14. Keluaran sesuai dengan masukan yang diberikan.

User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

No	Masukan	Keluaran
1	8.6	Air layak minum
2	9	Air tidak layak minum
3	16	Nilai pH tidak valid. Nilai pH harus antara 0 dan 14.

2. Latihan Soal 2

Source Code:

```
package main

import "fmt"

func main() {
    var jenisKendaraan string
    var tarif float64
    var durasi int

    fmt.Print("Masukan jenis kendaraan : ")
    fmt.Scan(&jenisKendaraan)

    fmt.Print("Masukan durasi : ")
    fmt.Scan(&durasi)

    switch jenisKendaraan {
    case "motor":
        tarif = float64(durasi) * 2000
    case "mobil":
        tarif = float64(durasi) * 5000
    case "truk":
        tarif = float64(durasi) * 8000
    default:
        fmt.Println("Jenis kendaraan tidak valid")
        return
    }
    fmt.Println("Total biaya Rp", tarif)
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\2Latsol.go"
Masukan jenis kendaraan : motor
Masukan durasi : 3 jam
Total biaya Rp 6000
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\2Latsol.go"
Masukan jenis kendaraan : mobil
Masukan durasi : 1 jam
Total biaya Rp 5000
PS D:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 AlPro\Go\WEEK11LAPRAK\2Latsol.go"
Masukan jenis kendaraan : truk
Masukan durasi : 5 jam
Total biaya Rp 40000
```

Deskripsi Program:

Program 2Soal.go dibuat untuk menghitung tarif parkir berdasarkan jenis kendaraan dan durasi parkir yang dimasukkan oleh pengguna. Ada tiga jenis kendaraan: motor, mobil, dan truk, dengan masing-masing memiliki tarif parkir yang berbeda. Motor dikenakan tarif sebesar Rp 2.000 per jam, mobil sebesar Rp 5.000 per jam, dan truk sebesar Rp 8.000 per jam.

Program harus dapat menentukan tarif per jam berdasarkan jenis kendaraan, lalu mengalikan tarif tersebut dengan jumlah jam parkir untuk menghasilkan total biaya parkir. Jika durasi parkir kurang dari 1 jam, maka durasi tetap dianggap 1 jam.

Masukan berupa jenis kendaraan dan durasi parkir dalam jam.

Keluaran berupa total biaya parkir berdasarkan jenis kendaraan dan durasi parkir.

No	Masukan	Keluaran
1	motor 3 jam	Rp 6000
2	mobil 1 jam	Rp 5000
3	truk 5 jam	Rp 40000

3. Latihan Soal 3

Source Code:

```
package main

import "fmt"

func main() {
    var bilangan int

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

    switch {
    case bilangan%10 == 0:
        fmt.Println("Kategori : Bilangan kelipatan 10")
        fmt.Printf("Hasil pembagian antara %d / 10 = %d\n", bilangan, bilangan/10)

    case bilangan%5 == 0 && bilangan != 5:
        fmt.Println("Kategori : Bilangan kelipatan 5")
        fmt.Printf("Hasil kuadrat dari %d ^2 = %d\n", bilangan, bilangan*bilangan)

    case bilangan%2 == 1:
        fmt.Println("Kategori : Bilangan ganjil")
        fmt.Printf("Hasil penjumlahan dengan bilangan berikutnya %d + %d = %d\n",
            bilangan, bilangan+1, bilangan+(bilangan+1))

    case bilangan%2 == 0 && bilangan%5 != 0 && bilangan%10 != 0:
        fmt.Println("Kategori : Bilangan genap")
        fmt.Printf("Hasil perkalian dengan bilangan berikutnya %d * %d = %d\n",
            bilangan, bilangan+1, bilangan*(bilangan+1))

    default:
        fmt.Println("Tidak ada bilangan yang cocok")
    }
}
```

Output:

```
PS D:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK\3Latsol.go"
Masukan bilangan: 5
Kategori : Bilangan ganjil
Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
PS D:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK\3Latsol.go"
Masukan bilangan: 8
Kategori : Bilangan genap
Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
PS D:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK\3Latsol.go"
Masukan bilangan: 25
Kategori : Bilangan kelipatan 5
Hasil kuadrat dari 25 ^2 = 625
PS D:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK> go run "d:\Semester1\Semester 1 APro\Go\WEEK11LAPRAK\3Latsol.go"
Masukan bilangan: 20
Kategori : Bilangan kelipatan 10
Hasil pembagian antara 20 / 10 = 2
```

Deskripsi Program:

Program 3Soal.go dibuat untuk mengidentifikasi pola aritmatika berdasarkan bilangan yang diinputkan dan melakukan operasi matematika yang sesuai. Beberapa ketentuan kategori diantaranya:

- Bilangan Ganjil : Menghitung penjumlahan antara bilangan yang diinput dengan bilangan berikutnya
- Bilangan Genap : Menghitung perkalian antara bilangan yang diinput dengan bilangan berikutnya
- Bilangan Kelipatan 5 : Menghitung hasil kuadrat dari bilangan yang diinputkan
- Bilangan Kelipatan 10 : Membagi bilangan yang diinputkan dengan bilangan 10

Masukan terdiri dari satu bilangan bulat.

Keluaran berupa teks "Kategori: Bilangan Ganjil" dilanjutkan dengan "Hasil penjumlahan dengan bilangan berikutnya $%(input) + %(input+1) = %(hasil)$ " untuk Bilangan Ganjil. "Kategori: Bilangan Genap" dilanjutkan dengan "Hasil perkalian dengan bilangan berikutnya $%(input) * %(input+1) = %(hasil)$ " untuk Bilangan Genap. "Kategori: Bilangan Kelipatan 5" dilanjutkan dengan "Hasil kuadrat dari $%(input)^2 = %(hasil)$ " untuk Bilangan Kelipatan 5. "Kategori: Bilangan Kelipatan 10" dilanjutkan dengan "Hasil pembagian antara $%(input) / 10 = %(hasil)$ " untuk Bilangan Kelipatan 10. Keluaran sesuai dengan masukan yang diberikan. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan output;

No	Masukan	Keluaran
1	5	Kategori: Bilangan Ganjil Hasil penjumlahan dengan bilangan berikutnya 5 + 6 = 11
2	8	Kategori: Bilangan Genap Hasil perkalian dengan bilangan berikutnya 8 * 9 = 72
3	25	Kategori: Bilangan Kelipatan 5 Hasil kuadrat dari 25 ^2 = 625
4	20	Kategori: Bilangan Kelipatan 10 Hasil pembagian antara 20 / 10 = 2

DAFTAR PUSTAKA

<file:///D:/Semester%201%20AlPro/Modul/MODUL%202.pdf>

<https://www.toppr.com/guides/computer-science/introduction-to-c/data-types-variables-and-constants/declaration-of-variables/>

<https://www.toppr.com/guides/computer-science/introduction-to-c/data-types-variables-and-constants/constants-in-programming-language/>

<https://www.dicoding.com/blog/macam-macam-tipe-data/>

<file:///D:/Semester%201%20AlPro/Modul/MODUL%203.pdf>

<file:///D:/Semester1/Semester%201%20AlPro/Modul/MODUL%205%20dan%206.pdf>

<file:///D:/Semester1/Semester%201%20AlPro/WEEKMATERI/MODUL%209.pdf>

<file:///D:/Semester1/Semester%201%20AlPro/WEEKMATERI/MODUL%2011.pdf>