

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 12
“WHILE-LOOP”



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.

14. 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. Sebelumnya pada modul ke-5 dan 6 telah dipelajari instruksi perulangan dengan for-loop. Instruksi for-loop memungkinkan kita melakukan berulangan sebanyak n iterasi, akan tetapi pada banyak kasus yang melibatkan perulangan, tidak semua perulangan diketahui jumlah iterasinya di awal. Perulangan seperti ini disebut juga dengan istilah perulangan dengan kondisi.

13. Karakteristik While-Loop

Struktur kontrol perulangan menggunakan while-loop memiliki bentuk yang hampir serupa dengan penulisan if-then pada percabangan, yaitu memiliki kondisi dan aksi. Hal yang membedakan adalah aksi akan dilakukan secara berulang-ulang selama kondisi bernilai true.

- 1) Kondisi, merupakan nilai atau operasi tipe data yang menghasilkan tipe data boolean.

Kondisi ini merupakan syarat terjadinya perulangan. Artinya perulangan terjadi apabila kondisi bernilai true.

- 2) Aksi, merupakan kumpulan instruksi yang akan dieksekusi secara berulang-ulang selama kondisi bernilai true. *Salah satu instruksi dari aksi harus bisa membuat kondisi yang awalnya bernilai true menjadi false, tujuannya adalah untuk membuat perulangan*

berhenti. Pada penulisan notasinya secara umum bahasa pemrograman menggunakan kata kunci while, tetapi khusus di bahasa pemrograman Go, kata kunci yang digunakan adalah for. Walaupun berbeda dari kata kunci yang digunakan, secara struktur penulisannya tetap sama, sehingga tetap mudah untuk membedakan instruksi for yang digunakan adalah for-loop atau while-loop

CONTOH SOAL

1. Contoh Latihan Soal 1

Source Code:

```
package main

import "fmt"

func main() {
    var n, j int
    fmt.Scan(&n)
    j = n

    for j > 1 {
        fmt.Print(j, " x ")
        j = j - 1
    }
    fmt.Println(1)
}
```

Output:

```
PS D:\Semester1\Semester 1 APro\Go\MODUL12> go run "d:\Semester1\Semester 1 APro\Go\MODUL12\1coso.go"
5
5 x 4 x 3 x 2 x 1
PS D:\Semester1\Semester 1 APro\Go\MODUL12> go run "d:\Semester1\Semester 1 APro\Go\MODUL12\1coso.go"
10
10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1
PS D:\Semester1\Semester 1 APro\Go\MODUL12> go run "d:\Semester1\Semester 1 APro\Go\MODUL12\1coso.go"
1
1
```

Deskripsi Program:

Program 1coso.go digunakan untuk menampilkan deret bilangan Faktorial dari suatu bilangan.

Masukan terdiri dari sebuah bilangan bulat non negatif n.

Keluaran berupa deret bilangan dari Faktorial n. Perhatikan contoh masukan dan keluaran 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	0	1
2	5	5 x 4 x 3 x 2 x 1
3	10	10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1
4	1	1

2. Contoh Latihan Soal 2

Source Code:

```
package main

import "fmt"

func main() {
    var token string
    fmt.Scan(&token)

    for token != "12345abcde" {
        fmt.Scan(&token)
    }
    fmt.Println("Selamat Anda Berhasil Login")
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\MODUL12> go run "d:\Semester1\Semester 1 AlPro\Go\MODUL12\2coso.go"
Qwe12312
231234
13213
123lijwe
12345abcde
Selamat Anda Berhasil Login
PS D:\Semester1\Semester 1 AlPro\Go\MODUL12> go run "d:\Semester1\Semester 1 AlPro\Go\MODUL12\2coso.go"
12345abcde
Selamat Anda Berhasil Login
```

Deskripsi Program:

Program 2coso.go digunakan untuk login ke dalam suatu aplikasi. Asumsi token untuk yang valid adalah "12345abcde".

Masukan terdiri dari suatu token. Selama token yang diberikan salah, maka program akan meminta token secara terus menerus hingga token yang diberikan benar.

Keluaran adalah teks yang menyatakan "Selamat Anda berhasil login".

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

No	Masukan	Keluaran
1	Qwe12312 231234 13213 123lijwe 12345abcde	Selamat Anda berhasil login
2	12345abcde	Selamat Anda berhasil login

3. Contoh Latihan Soal 3

Source Code:

```
package main

import "fmt"

func main() {
    var N, s1, s2, j, temp int
    fmt.Scan(&N)

    s1 = 0
    s2 = 1
    j = 0
    for j < N {
        fmt.Print(s1, " ")
        temp = s1 + s2
        s1 = s2
        s2 = temp
        j = j + 1
    }
}
```

Output:

```
PS D:\Semester1\Semester 1 A1Pro\Go\MODUL12> go run "d:\Semester1\Semester 1 A1Pro\Go\MODUL12\3coso.go"
5
0 1 1 2 3
PS D:\Semester1\Semester 1 A1Pro\Go\MODUL12> go run "d:\Semester1\Semester 1 A1Pro\Go\MODUL12\3coso.go"
2
0 1
PS D:\Semester1\Semester 1 A1Pro\Go\MODUL12> go run "d:\Semester1\Semester 1 A1Pro\Go\MODUL12\3coso.go"
10
0 1 1 2 3 5 8 13 21 34
```

Deskripsi Program:

Program 3coso.go digunakan untuk mencetak N bilangan pertama dalam deret Fibonacci..

Masukan terdiri dari bilangan bulat positif N dengan nilai besar atau sama dengan 2.

Keluaran terdiri dari sejumlah N bilangan yang menyatakan N deret bilangan Fibonacci yang pertama.

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

No	Masukan	Keluaran
1	5	0 1 1 2 3
2	2	0 1
3	10	0 1 1 2 3 5 8 13 21 34

SOAL LATIHAN

1. Latihan Soal 1

Source Code:

```
package main

import "fmt"

func main() {

    const username = "Admin"
    const password = "Admin"
    var ulang int

    for {
        var inputUsername, inputPassword string
        fmt.Print("Masukkan Username : ")
        fmt.Scanln(&inputUsername)
        fmt.Print("Masukan Password: ")
        fmt.Scanln(&inputPassword)

        if inputUsername == username && inputPassword == password {
            fmt.Printf("%d percobaan gagal login\n", ulang)
            fmt.Println("Login Berhasil")
            break
        } else {
            fmt.Println("Username atau Password salah")
            ulang++
        }
    }
}
```


Output:

```
PS D:\Semester1\Semester 1 A1Pro\Go\MODUL12> go run "d:\Semester1\Semester 1 A1Pro\Go\MODUL12\Latsol1.go"
Masukkan Username : User123
Masukkan Password: user123
Username atau Password salah
Masukkan Username : User
Masukkan Password: admin
Username atau Password salah
Masukkan Username : Admin
Masukkan Password: admin
Username atau Password salah
Masukkan Username : Admin
Masukkan Password: Admin123
Username atau Password salah
Masukkan Username : Admin
Masukkan Password: Admin
4 percobaan gagal login
Login Berhasil
PS D:\Semester1\Semester 1 A1Pro\Go\MODUL12> go run "d:\Semester1\Semester 1 A1Pro\Go\MODUL12\Latsol1.go"
Masukkan Username : Admin
Masukkan Password: Admin
0 percobaan gagal login
Login Berhasil
```

Deskripsi Program:

Program Soal1.go dibuat dengan tujuan untuk menghitung berapa banyak seseorang pengguna gagal melakukan login, karena kesalahan memberikan username dan password.

Masukan terdiri dari dua teks yang berisi username dan password, apabila username dan password salah, maka program akan meminta masukan ulang. Apabila username dan password sudah benar maka program akan menampilkan informasi berapa kali percobaan login yang gagal dilakukan. Asumsi username dan password yang benar adalah "Admin" dan "Admin" tanpa tanda petik.

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

No	Masukan	Keluaran
1	User123 user123 User admin Admin admin Admin Admin123 Admin Admin	4 percobaan gagal login
2	Admin Admin	0 percobaab gagal login

2. Latihan Soal 2

Source Code:

```
package main

import "fmt"

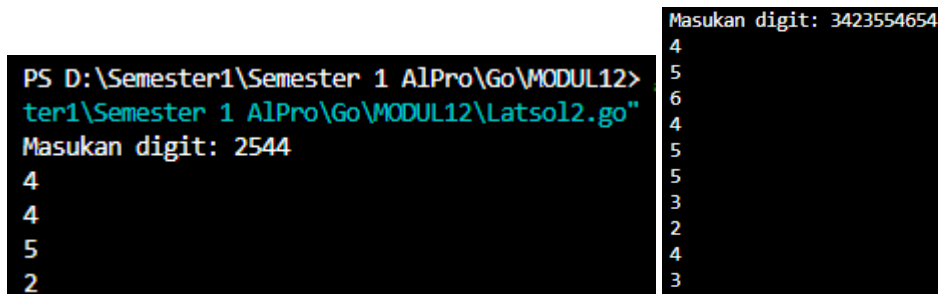
func main() {

    var digit int
    fmt.Print("Masukan digit: ")
    fmt.Scan(&digit)

    if digit <= 0 {
        fmt.Print("harap masukan bilangan bulat positif")
        return
    }

    for digit > 0 {
        angka := digit % 10
        fmt.Println(angka)
        digit /= 10
    }
}
```

Output:



Deskripsi Program:

Program 2Soal.go dibuat untuk mencacah setiap digit yang terdapat di dalam suatu bilangan bulat positif.

Masukan terdiri dari suatu bilangan bulat positif.

Keluaran terdiri dari nilai digit pada bilangan tersebut. Tampilkan dari digit terakhir (paling kanan) sampai dengan digit pertama (paling kiri).

No	Masukan	Keluaran
1	2	2
2	2544	4 4 5 2

3	3423554654	4 5 6 4 5 5 3 2 4 3
---	------------	--

3. Latihan Soal 3

Source Code:

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Print("Masukan Nilai x : ")
    fmt.Scan(&x)
    fmt.Print("Masukan Nilai y : ")
    fmt.Scan(&y)

    hasil := 0
    for x >= y {
        x = x - y
        hasil++
    }
    fmt.Println(hasil)
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\MODUL12> go run "d:\Semester1\Semester 1 AlPro\Go\MODUL12\Latsol3.go"
Masukan Nilai x : 5
Masukan Nilai y : 2
2
PS D:\Semester1\Semester 1 AlPro\Go\MODUL12> go run "d:\Semester1\Semester 1 AlPro\Go\MODUL12\Latsol3.go"
Masukan Nilai x : 10
Masukan Nilai y : 7
1
PS D:\Semester1\Semester 1 AlPro\Go\MODUL12> go run "d:\Semester1\Semester 1 AlPro\Go\MODUL12\Latsol3.go"
Masukan Nilai x : 120
Masukan Nilai y : 4
30
```

Deskripsi Program:

Program 3Soal.go untuk mencari hasil integer division dari dua bilangan. Gunakan perulangan dan tidak diperbolehkan menggunakan operator pembagian.

Masukan terdiri dari dua bilangan bulat positif x dan y, yang mana $x \geq y$.

Keluaran terdiri dari hasil dari operasi $x \div y$.

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

No	Masukan	Keluaran
1	5 2	2
2	10 7	1
3	120 4	30

DAFTAR PUSTAKA

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

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

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

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

<file:///C:/Users/Laptopku/Downloads/MODUL%2012.pdf>