

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



DISUSUN OLEH:
SHEILA STEPHANIE ANINDYA
103112400086
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

I. Paradigma Perulangan

Perulangan adalah struktur kontrol yang memungkinkan eksekusi instruksi yang sama dilakukan secara berulang, sehingga menghindari penulisan instruksi berulang secara manual. For-loop, yang dipelajari pada modul sebelumnya, digunakan untuk perulangan dengan jumlah iterasi yang sudah diketahui. Namun, pada kasus tertentu, jumlah iterasi tidak diketahui di awal. Perulangan semacam ini disebut perulangan dengan kondisi, di mana pengulangan bergantung pada pemenuhan kondisi tertentu.

II. While-Loop

While Loop adalah metode perulangan dimana ada kondisi yang harus dipenuhi supaya looping bisa berjalan terus. While Loop mengulangi eksekusi sub diagram didalamnya sampai terminal kondisi menerima nilai Boolean tertentu. Nilai Boolean tergantung dari sifat dari While Loop. Dengan cara klik kanan pada terminal kondisi dapat dilakukan pemilihan Stop if True atau Continue if True. While Loop akan selalu dieksekusi minimal satu kali. Jumlah pengulangan eksekusi dapat diambil dari terminal iterasi (i).

- 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.*

CONTOH SOAL

1. Latihan1

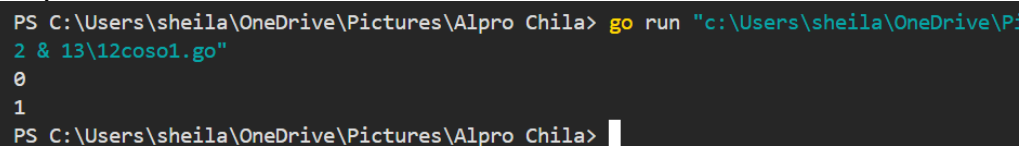
Source Code:

```
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 C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\P
2 & 13\12coso1.go"
0
1
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> |
```

Deskripsi Program:

Program ini membuat deret bilangan Faktorial dari suatu bilangan dengan menggunakan perulangan for, yang akan berjalan selama $j > 1$.

2. Latihan2

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 C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chila\o"
12345abcde
Selamat Anda berhasil login
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> 
```

Deskripsi Program:

Program ini memvalidasi login menggunakan token dengan memasukkan token berulang kali hingga input token sesuai dengan string yang diharapkan (12345abcde).

3. Latihan3

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 C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chila\o"
10
0 1 1 2 3 5 8 13 21 34
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> 
```

Deskripsi Program:

Program ini menghasilkan deret Fibonacci sebanyak N yang diinput, dengan s1 (bilangan awal 0) dan s2 (bilangan awal 1) sebagai bilangan Fibonacci pertama. Serta j sebagai iterasi.

SOAL LATIHAN

Statement perulangan

1. Buatlah sebuah program dalam bahasa Go yang digunakan untuk menghitung berapa banyak seseorang pengguna gagal melakukan login, karena kesalahan memberikan username dan password.

Source Code:

```
package main

import "fmt"

func main() {
    var username, password string

    for {
        correctUsername := "Admin"
        correctPassword := "Admin"
        failedAttempts := 0

        fmt.Print("username: ")
        fmt.Scanln(&username)

        fmt.Print("password: ")
        fmt.Scanln(&password)

        if username == correctUsername && password == correctPassword {
            fmt.Printf("%d percobaan gagal login.\n", failedAttempts)
            break
        } else {
            fmt.Println("gagal login.")
            failedAttempts++
        }
    }
}
```

Output:

```
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chila\Lapra 1
username: Admin
password: Admin
0 percobaan gagal login.
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila>
```

Deskripsi Program:

Program ini menghitung jumlah percobaan login yang gagal dengan memasukkan username dan password secara berulang hingga keduanya benar sebagai perulangan.

Statement perulangan

2. Buatlah program yang digunakan untuk mencacah setiap digit yang terdapat di dalam suatu bilangan bulat positif.

Source Code:

```
package main

import "fmt"

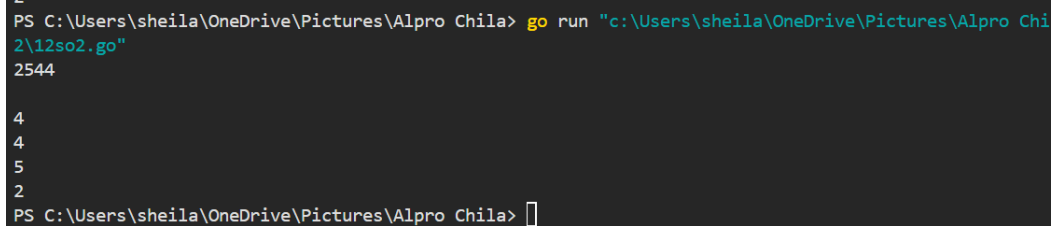
func main() {
    var bilangan int
    fmt.Scan(&bilangan)

    if bilangan <= 0 {
        fmt.Println("Tidak valid")
        return
    }

    fmt.Println()

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

Output:



```
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chi
2\12so2.go"
2544

4
4
5
2
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> █
```

Deskripsi Program:

Program ini mencacah dan menampilkan setiap digit dalam sebuah bilangan bulat positif, dimulai dari digit terakhir (paling kanan) hingga digit pertama (paling kiri).

Statement perulangan

3. Buatlah program untuk mencari hasil integer division dari dua bilangan. Gunakan perulangan dan tidak diperbolehkan menggunakan operator pembagian.

Source Code:

```
package main

import "fmt"

func main() {
    var x, y int

    fmt.Print("x = ")
    fmt.Scanln(&x)
    fmt.Print("y = ")
    fmt.Scanln(&y)

    if x <= 0 || y <= 0 {
        fmt.Println("tidak valid")
        return
    }
    result := 0
    for x >= y {
        x = x - y
        result++
    }

    fmt.Println(result)
}
```

Output:

```
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chila\main.go"
x = 5
y = 2
2
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> █
```

Deskripsi Program:

Program ini menghitung hasil integer division ($x \div y$) dari dua bilangan bulat positif x dan y tanpa menggunakan operator pembagian.

DAFTAR PUSTAKA

1. Telkom University. *Modul 12 Praktikum Algoritma dan Pemrograman*. Fakultas Informatika, Telkom University, 2024.
2. Unikom. (n.d.). *Materi Pertemuan 4: Labview 1 + While Loop + Shift Register*.
https://repository.unikom.ac.id/62967/1/Materi%20Pertemuan%204_Labview%201%20While%20Loop%20%2B%20Shift%20Register.pdf

