

**LAPORAN PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN 1**  
**MODUL 12**  
**WHILE-LOOP**



**DISUSUN OLEH:**  
**SETYO NUGROHO**

**103112400024**

**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. 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 perulangan 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.

### 2. 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.

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 1

Source Code:

```
coso1 > go coso1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n, j int
7     fmt.Scan(&n)
8     j = n
9     for j > 1 {
10        fmt.Print(j, "x")
11        j = j - 1
12    }
13    fmt.Println(1)
14 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\F
0
1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\F
5
5x4x3x2x1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\F
10
10x9x8x7x6x5x4x3x2x1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\F
1
1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> 
```

Deskripsi Program:

program yang digunakan untuk menampilkan deret bilangan Faktorial dari satu bilangan. Masukan terdiri dari sebuah bilangan bulat non negatif n. dan Keluaran berupa deret bilangan dari Faktorial n. Perhatikan contoh masukan dan keluaran yang diberikan.

## 2. Contoh 2

Source Code:

```
coso2 > go coso2.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var token string
7      fmt.Scan(&token)
8      for token != "12345abcde" {
9          fmt.Scan(&token)
10     }
11     fmt.Println("Selamat Anda Berhasil Login")
12 }
13
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRA
qwe12312
231234
13213
123lijwe
12345abcde
Selamat Anda Berhasil Login
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRA
12345abcde
Selamat Anda Berhasil Login
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> |
```

Deksripsi Program:

Program ini berfungsi untuk memvalidasi token login pengguna, di mana token yang valid adalah "12345abcde". Program akan meminta pengguna untuk memasukkan token dan memeriksa validitasnya. Jika token yang dimasukkan salah, program akan terus meminta pengguna untuk memasukkan ulang token hingga token yang benar diberikan. Setelah pengguna berhasil memasukkan token yang valid, program akan menampilkan pesan: "Selamat Anda berhasil login".

### 3. Contoh 3

Source Code:

```
coso3 > go coso3.go > main
2
3 import "fmt"
4
5 func main() {
6     var N, s1, s2, j, temp int
7     fmt.Scan(&N)
8     s1 = 0
9     s2 = 1
10    j = 0
11    for j < N {
12        fmt.Print(s1, " ")
13        temp = s1 + s2
14        s1 = s2
15        s2 = temp
16        j = j + 1
17    }
18 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12.go"
5
0 1 1 2 3
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12.go"
2
0 1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12.go"
10
0 1 1 2 3 5 8 13 21 34
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> 
```

Deksripsi Program:

Program ini mencetak deret Fibonacci sebanyak **N bilangan pertama**, di mana **N** adalah bilangan bulat positif yang bernilai minimal 2. Prosesnya dimulai dengan meminta input **N** dari pengguna. Program kemudian menghitung dan mencetak **N** bilangan pertama dalam deret Fibonacci, dimulai dari 0 dan 1. Deret Fibonacci dihitung dengan menjumlahkan dua bilangan sebelumnya untuk mendapatkan bilangan berikutnya. Hasil akhirnya adalah deret Fibonacci sebanyak **N bilangan** yang ditampilkan sebagai keluaran.

## SOAL LATIHAN

1.

Source Code:

```
latsol1 > go latsol1.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var username, password string
7      fmt.Scan(&username, &password)
8      percobaan := 0
9      for username != "Admin" || password != "Admin" {
10         fmt.Scan(&username, &password)
11         percobaan++
12     }
13     fmt.Println(percobaan, "Percobaan gagal login")
14 }
15
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING
User123 user123
User admin
Admin admin
Admin Admin123
Admin Admin
4 Percobaan gagal login
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING
Admin Admin
0 Percobaan gagal login
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12>
```

Deskripsi Program:

Program ini meminta pengguna memasukkan username dan password. Jika kombinasi yang dimasukkan tidak sama dengan "Admin", program akan meminta pengguna untuk mencoba lagi. Program menggunakan variabel percobaan untuk menghitung jumlah upaya login yang gagal. Proses ini berlangsung dalam loop hingga pengguna memberikan kombinasi username dan password yang benar. Setelah berhasil login, program mencetak jumlah percobaan gagal disertai pesan "Percobaan gagal login".

2.

Source Code:

```
latssol2 > go latssol2.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     fmt.Scan(&n)
8     for n > 0 {
9         digit := n % 10
10        fmt.Println(digit)
11        n = n / 10
12    }
13 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM
2
2
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM
2544
4
4
5
2
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM
3423554654
4
5
6
4
5
5
3
2
4
3
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> |
```

Deksripsi Program:

Program ini memecah angka yang dimasukkan pengguna menjadi digit-digitnya. Pengguna diminta memasukkan sebuah bilangan integer  $n$ . Dalam loop, digit terakhir diambil menggunakan operasi modulus ( $n \% 10$ ) dan dicetak. Kemudian digit terakhir dihapus dengan pembagian bilangan bulat ( $n = n // 10$ ). Proses ini berulang hingga nilai  $n$  menjadi nol. Digit-digit dicetak dalam urutan terbalik dari angka aslinya karena pemrosesan dimulai dari digit paling kanan.

3.

Source Coding:

```
latsol3 > go latsol3.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x, y int
7     fmt.Scan(&x, &y)
8     hasil := 0
9     for x >= y {
10         x = x - y
11         hasil++
12     }
13     fmt.Println(hasil)
14 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\F
5 2
2
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\F
10 7
1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> go run "d:\ALGORITMA PROGRAMING\F
120 4
30
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum_Modul_12> |
```

Deksripsi Program:

Program ini menghitung hasil pembagian bilangan bulat menggunakan metode pengurangan berulang. Pengguna diminta memasukkan dua bilangan bulat, x dan y. Program mengurangi y dari x secara berulang hingga x kurang dari y, sambil menambahkan 1 ke variabel hasil pada setiap pengurangan. Setelah loop selesai, nilai hasil yang mencerminkan hasil pembagian bilangan bulat tanpa menggunakan operator pembagian (/) akan dicetak.



## **DAFTAR PUSTAKA**

**Prasti Eko Yunanto, S.T., M.Kom.**

**MODUL PRAKTIKUM 12-WHILW-LOOP ALGORITMA DAN PEMOGRAMAN 1  
S1 INFORMATIKA**