

**LAPORAN PRAKTIKUM ALGORITMA**

**DAN PEMROGRAMAN 1**

**MODUL 12**

**WHILE-LOOP**



**Disusun oleh:**

**ALIN KARISA HIZANNAH**

**109082500010**

**S1IF-13-07**

**Asisten Praktikum**

Adithana dharma putra

Apri pandu wicaksono

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## LATIHAN KELAS – GUIDED

### 1. Guided 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)
}
```

#### Screenshot program

```
guide1.go 2
guide1.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 }
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrograman\Praktek 12> go run guide1.go
5
5 x 4 x 3 x 2 x 1
PS D:\TEL-U\Semester 1\Algoritma Pemrograman\Praktek 12> go run guide1.go
10
10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1
PS D:\TEL-U\Semester 1\Algoritma Pemrograman\Praktek 12> go run guide1.go
1
PS D:\TEL-U\Semester 1\Algoritma Pemrograman\Praktek 12>
```

```
109082500010
IF-13-07
ALIN KARISA HIZANNAH
```

## Deskripsi program

Program ini dibuat untuk menampilkan deret bilangan faktorial dari suatu bilangan yang diinputkan oleh pengguna. Deret faktorial tersebut ditampilkan dalam bentuk urutan menurun, dimulai dari bilangan awal kemudian diikuti bilangan sebelumnya hingga mencapai angka 1. Bentuk tampilan menggunakan simbol “x” sebagai pemisah, sehingga menghasilkan format seperti  $n \times (n-1) \times \dots \times 1$ . Pola ini membantu pengguna memahami proses perhitungan faktorial sebelum nantinya dikalikan satu per satu dalam konsep matematika.

Selain menghasilkan urutan bilangan faktorial, program juga memperhatikan kondisi khusus seperti ketika nilai input sama dengan 1. Dalam kasus ini, program langsung menampilkan angka 1 karena faktorial dari 1 adalah 1. Dengan logika perulangan yang sederhana dan mudah dipahami, program ini cocok digunakan sebagai dasar pembelajaran pemrograman, khususnya mengenai konsep perulangan dan faktorial dalam bahasa Go.

## 2. Guided 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")
}
```

### Screenshot program

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run guide2.go
Qwe12312
231213
1231ijwe
12345abcde
Selamat Anda berhasil login
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run guide2.go
12345abcde
Selamat Anda berhasil login
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12>
```

109082500010  
IF-13-07  
ALIN KARISA HIZANNAH

## Deskripsi program

Program ini dibuat untuk mengecek apakah pengguna memasukkan token yang benar. Ketika program dijalankan, pengguna diminta mengetikkan sebuah token. Jika token yang dimasukkan belum sesuai dengan token yang benar, yaitu "12345abcde", maka program akan terus meminta input ulang. Proses ini berlangsung dalam sebuah perulangan sehingga pengguna tidak bisa lanjut sebelum memasukkan token yang tepat.

Setelah pengguna akhirnya memasukkan token yang benar, barulah program menghentikan perulangan dan menampilkan pesan "Selamat Anda berhasil login". Dengan cara ini, program memastikan bahwa hanya token yang valid yang bisa digunakan untuk masuk, sehingga mekanismenya mirip dengan sistem login sederhana.

### 3. Guided 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.Println(s1, " ")
        temp = s1 + s2
        s1 = s2
        s2 = temp
        j = j + 1
    }
}
```

#### Screenshot program

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run guide3.go
5
0 1 1 2 3
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run guide3.go
10
0 1 1 2 3 5 8 13 21 34
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12>
```

The screenshot shows a terminal window with two command-line sessions. The first session runs the Go program 'guide3.go' with the argument '5', resulting in the output '0 1 1 2 3'. The second session runs it with the argument '10', resulting in the output '0 1 1 2 3 5 8 13 21 34'. The terminal window has tabs for 'powershell' and 'cmd'.

## Deskripsi program

Program ini berfungsi untuk menampilkan deret Fibonacci sebanyak  $N$  bilangan.

Pengguna memasukkan nilai  $N$  terlebih dahulu, kemudian program memulai deret dengan dua nilai awal yaitu 0 dan 1. Variabel  $s1$  mewakili bilangan saat ini, sedangkan  $s2$  adalah bilangan berikutnya. Pada setiap iterasi, program mencetak nilai  $s1$ , lalu menghitung bilangan Fibonacci berikutnya dengan menjumlahkan dua bilangan sebelumnya.

Setelah nilai baru dihitung, program memperbarui posisi bilangan:  $s1$  menjadi nilai lama  $s2$ , dan  $s2$  menjadi hasil penjumlahan sebelumnya. Proses ini terus berulang hingga jumlah bilangan yang dicetak mencapai  $N$ . Dengan cara tersebut, program menghasilkan deret Fibonacci secara berurutan mulai dari 0.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {
    var usr, pwd string
    var count int

    for {
        fmt.Scan(&usr, &pwd)

        if usr == "Admin" && pwd == "Admin" {
            fmt.Printf("%d percobaan gagal login\n",
count)
            break
        } else {
            count++
        }
    }
}
```

#### Screenshot program

```
Praktek 12
powershell X
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal1.go
User123 user123
User admin
Admin admin
Admin Admin123
Admin Admin
4 percobaan gagal login
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal1.go
Admin Admin
0 percobaan gagal login
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12>
```

109082500010  
IF-13-07  
ALIN KARISA HIZANNAH

## Deskripsi program

Program ini dibuat untuk melakukan validasi login dengan mencocokkan username dan password yang dimasukkan pengguna. Program meminta input secara berulang di dalam sebuah loop tanpa batas hingga kredensial yang dimasukkan sesuai dengan data yang telah ditentukan, yaitu username "Admin" dan password "Admin". Selama pengguna belum berhasil login, proses input akan terus dilakukan.

Selama percobaan login berlangsung, setiap input yang tidak sesuai akan dihitung sebagai percobaan gagal, dan nilainya disimpan dalam variabel count. Ketika akhirnya pengguna memasukkan kombinasi yang benar, loop dihentikan dan program menampilkan jumlah percobaan gagal yang telah terjadi sebelumnya. Dengan cara ini, program tidak hanya memvalidasi login, tetapi juga memberikan laporan jumlah kesalahan input sebelum login berhasil.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {
    var num,digit int
    fmt.Print("Masukkan sebuah bilangan bulat positif:
")
    fmt.Scan(&num)

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

### Screenshoot program

```
Praktek 12
powershell
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal2.go
Masukkan sebuah bilangan bulat positif: 2
2
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal2.go
Masukkan sebuah bilangan bulat positif: 2544
4
4
5
2
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal2.go
Masukkan sebuah bilangan bulat positif: 3423554654
4
5
6
4
5
5
3
2
4
3
3
2
4
3
ALIN KARISA HIZANNAH
```

## Deskripsi program

Program ini berfungsi untuk menampilkan setiap digit dari sebuah bilangan bulat positif yang dimasukkan oleh pengguna. Setelah nilai dimasukkan melalui input, program mulai memproses bilangan tersebut dari digit paling belakang. Proses ini dilakukan dengan mengambil sisa hasil bagi bilangan terhadap 10, sehingga digit terakhir dapat diperoleh dan kemudian langsung ditampilkan.

Setelah digit terakhir ditampilkan, program menghapus digit tersebut dari bilangan dengan membaginya menggunakan pembagian bulat. Langkah ini dilakukan secara berulang selama masih ada digit yang tersisa. Dengan demikian, seluruh digit akan ditampilkan satu per satu dari posisi paling belakang hingga digit paling depan, menghasilkan tampilan angka yang terurai secara terbalik sesuai urutan digit aslinya.

### 3. Tugas 3

#### Source code

```
package main

import "fmt"

func main() {
    var x, y, hasil int
    fmt.Print("Masukkan 2 bilangan x dan y: ")
    fmt.Scan(&x, &y)

    hasil = 0

    for x >= y {
        x = x - y
        hasil ++
    }

    fmt.Print(hasil)
}
```

#### Screenshoot program

The screenshot shows a code editor with a file named 'soal3.go'. The code implements integer division without using the division operator. It prompts the user for two integers, x and y, and then uses a for loop to repeatedly subtract y from x until x is less than or equal to y. The result is stored in a variable 'hasil'. Finally, it prints the value of 'hasil'. Below the code editor is a terminal window titled 'powershell' showing the execution of the program. The user inputs '5 2' and '10 7', and the program outputs '2' and '128' respectively. The terminal also shows the path 'D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12'. At the bottom of the screen, there is a status bar with various icons and text.

```
1 package main
2 import "fmt"
3
4 func main() {
5     var x, y, hasil int
6     fmt.Print("Masukkan 2 bilangan x dan y: ")
7     fmt.Scan(&x, &y)
8
9     hasil = 0
10
11    for x >= y {
12        x = x - y
13        hasil ++
14    }
15    fmt.Println(hasil)
16 }
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal3.go
Masukkan 2 bilangan x dan y: 5 2
2
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal3.go
Masukkan 2 bilangan x dan y: 10 7
128
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12> go run soal3.go
Masukkan 2 bilangan x dan y: 128 4
30
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Praktek 12>
```

## Deskripsi program

Program ini dirancang untuk menghitung hasil integer division dari dua bilangan tanpa menggunakan operator pembagian. Alih-alih memanfaatkan fitur bawaan seperti `/`, program ini menggunakan proses pengurangan berulang sebagai dasar perhitungan. Dengan cara tersebut, pembagian ditiru melalui operasi matematis sederhana yang diulang hingga tidak memungkinkan lagi melakukan pengurangan berikutnya.

Pada awal program, pengguna diminta memasukkan dua bilangan, yaitu `x` sebagai bilangan yang akan dibagi dan `y` sebagai pembagi. Variabel `hasil` disiapkan dan diinisialisasi dengan nilai nol untuk menyimpan banyaknya pengurangan yang dapat dilakukan. Selanjutnya, program memasuki perulangan `for` yang bekerja selama nilai `x` masih lebih besar atau sama dengan `y`. Setiap kali pengurangan terjadi, nilai `x` dikurangi oleh `y` dan penghitung `hasil` ditambah satu.

Logika ini meniru konsep dasar pembagian, yaitu melihat berapa kali sebuah bilangan dapat diambil dari bilangan lainnya. Ketika `x` sudah tidak cukup besar untuk dikurangi oleh `y`, proses perulangan berhenti. Hal ini menunjukkan bahwa tak ada lagi “kelipatan” dari `y` yang bisa diambil dari sisa nilai `x`, dan hasil bagi akhirnya sudah lengkap.

Setelah perulangan selesai, program menampilkan nilai hasil sebagai hasil pembagian versi integer. Nilai ini menggambarkan berapa kali bilangan pembagi dapat masuk sepenuhnya ke dalam bilangan yang dibagi. Pendekatan seperti ini sering digunakan dalam pemrograman dasar untuk memahami proses aritmatika secara lebih mendalam tanpa bergantung pada operator bawaan.