

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 12**

**While Loop**



**Disusun oleh:**

**MUHAMMAD ZAKI AL-FIQRI**

**109082500085**

**S1IF-13-02**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## LATIHAN KELAS – GUIDED

### Guided 1

#### Source Code

```
package main

import "fmt"

func main() {
    var n, j int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)
    j = n
    for j > 1 {
        fmt.Print(j, " x ")
        j = j - 1
    }
    fmt.Print(1)
}
```

## Screenshoot program

The screenshot shows a Go program in a VS Code editor. The program is a simple factorial calculator. It prompts the user to enter a number, then uses a for loop to calculate the factorial by multiplying the number by decreasing integers until it reaches 1. The output is displayed in a terminal window, showing the input number 3 and the resulting factorial calculation: 3 x 2 x 1 = 6.

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

Terminal Output:

```
Masukkan bilangan: 3
3 x 2 x 1
PS C:\Users\HP\Desktop\vscode> go run "c:\Users\HP\Desktop\vscode\main.go"
Masukkan bilangan: 7
7 x 6 x 5 x 4 x 3 x 2 x 1
PS C:\Users\HP\Desktop\vscode>
```

## Deskripsi program

Program Go ini berfungsi untuk menampilkan perhitungan mundur perkalian faktorial dari bilangan bulat positif yang dimasukkan pengguna. Program meminta pengguna memasukkan sebuah bilangan bulat  $n$ . Nilai  $n$  disalin ke variabel  $j$ . Kemudian, program menggunakan perulangan `for` yang akan berjalan selama  $j$  lebih besar dari 1. Di dalam perulangan, program mencetak nilai  $j$  diikuti oleh kata `x`, dan kemudian nilai  $j$  dikurangi  $j = j - 1$ . Setelah perulangan selesai, yaitu Ketika  $j$  mencapai 1, program mencetak angka 1, sehingga menghasilkan urutan bilangan yang menunjukkan perhitungan faktorial, seperti  $5 \times 4 \times 3 \times 2 \times 1$ .

## Guided 2

### Source Code

```
package main
```

```
import "fmt"

func main() {
    var n string
    for {
        fmt.Print("masukan token : ")
        fmt.Scan(&n)

        if n == "12345abcde" {
            fmt.Print("selamat anda berhasil login")
            break
        } else {
            fmt.Print("token yang anda masukan salah")
        }
    }
}
```

The image shows a Go program in a code editor and its execution output in a terminal. The program is a simple authentication system that prompts the user for a token. If the token matches '12345abcde', it prints a success message and exits. Otherwise, it prints an error message and loops back to prompt the user again.

```
soal guided modul 12 soal 2 > go guided 2.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n string
7     for {
8         fmt.Print("masukan token : ")
9         fmt.Scan(&n)
10
11         if n == "12345abcde" {
12             fmt.Print("selamat anda berhasil login")
13             break
14         } else {
15             fmt.Print("token yang anda masukan salah")
16         }
17     }
18 }
19
```

Terminal Output:

```
selamat anda berhasil login
PS C:\Users\HP\Desktop\vscode> go run "c:\Users\HP\Desktop\vscode\soal guided modul 12 soal 2.go"
masukan token : 12345
token yang anda masukan salahmasukan token : 12345abcde
selamat anda berhasil login
PS C:\Users\HP\Desktop\vscode>
```

Output Window:

```
NAMA : MUHAMMAD ZAKI AL-FIQRI
NIM : 109082500085
kelas : if-13-02
```

### Deskripsi program

Program Go ini berfungsi sebagai sistem autentikasi sederhana berbasis token yang menggunakan perulangan tak terbatas. Program ini akan terus-menerus meminta pengguna untuk memasukkan sebuah token. Input token yang dimasukkan oleh pengguna kemudian diperiksa menggunakan pernyataan if untuk dicocokkan dengan nilai tetap sebagai acuan tokennya 12345abcde. Jika token yang dimasukkan **sesuai**, program akan mencetak pesan selamat anda berhasil login dan segera **menghentikan perulangan** dengan perintah break. Sebaliknya, jika token yang dimasukkan **salah**, program akan mencetak pesan token yang anda masukan salah dan **kembali mengulang** permintaan input token, memaksa pengguna untuk mencoba lagi hingga token yang benar dimasukkan.

### Guided 3

#### Source Code

```
package main

import "fmt"

func main() {
    var N int
    fmt.Scan(&N)
```

```
    if N >= 2 {  
        a, b := 0, 1  
        fmt.Print(a, " ", b)  
  
        for i := 2; i < N; i++ {  
            next := a + b  
            fmt.Print(" ", next)  
            a, b = b, next  
        }  
  
        fmt.Println()  
    }  
}
```

**Screenshoot program**

```
soal guided modul 12 soal 3 > guided 3.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var N int
7     fmt.Scan(&N)
8
9     if N >= 2 {
10         a, b := 0, 1
11         fmt.Print(a, " ", b)
12
13         for i := 2; i < N; i++ {
14             next := a + b
15             fmt.Print(" ", next)
16             a, b = b, next
17         }
18
19         fmt.Println()
20     }
21 }
22
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
12
0 1 1 2 3 5 8 13 21 34 55 89
PS C:\Users\HP\Desktop\vscode> go run "c:\User
7
0 1 1 2 3 5 8
PS C:\Users\HP\Desktop\vscode> 
```

NAMA MUHAMMAD ZAKI AL-FIQF

NIM : 109082500085  
kelas : if-13-02

Ln 3, Col 17 65 characters xA Formatted 100% Windows (CRLF) UTF-8

### Deskripsi program

Program Go ini berfungsi untuk **mencetak deret bilangan Fibonacci** sebanyak  $n$  suku, di mana  $n$  adalah bilangan bulat yang dimasukkan oleh pengguna. Setelah menerima input  $n$ , program akan mengecek apakah  $n$  lebih besar atau sama dengan 2. Jika ya, program akan menginisialisasi dan mencetak dua suku pertama deret, yaitu 0 dan 1. Kemudian, perulangan for dimulai dari suku ke-3 hingga suku ke- $n$ , menghitung suku berikutnya  $next$  sebagai hasil penjumlahan dua suku sebelumnya  $a + b$ , mencetaknya, dan memperbarui nilai  $a$  menjadi  $b$  dan  $b$  menjadi  $next$  untuk mempersiapkan perhitungan suku berikutnya.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {
```

```
const usernamebenar, passwordbenar = "Admin", "Admin"

var username, password string
gagallogin := 0

for {
    fmt.Print("masukan username : ")
    fmt.Scan(&username)
    fmt.Print("masukan password : ")
    fmt.Scan(&password)

    if username == usernamebenar && password == passwordbenar {
        break
    }

    gagallogin = gagallogin + 1
}

fmt.Printf("%d percobaan gagal login\n", gagallogin)
}
```



## Screenshoot program

The screenshot shows a Go program in a text editor and its execution in a terminal. The program is a simple login system that prompts for a username and password, checks them against predefined values, and either breaks the loop on success or increments a failure counter on failure.

```
modul 12 soal 1 > -go soal 1.go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      const usernamebenar, passwordbenar = "Admin", "Admin"
7
8      var username, password string
9      gagallogin := 0
10
11     for {
12         fmt.Print("masukan username : ")
13         fmt.Scan(&username)
14         fmt.Print("masukan password : ")
15         fmt.Scan(&password)
16
17         if username == usernamebenar && password == passwordbenar {
18             break
19         }
20
21         gagallogin = gagallogin + 1
22     }
23
24     fmt.Printf("%d percobaan gagal login\n", gagallogin)
25 }
26
```

The terminal output shows the program's execution with the following input and output:

```
masukan username : admin
masukan password : admin123
masukan username : Admin
masukan password : Admin
3 percobaan gagal login
PS C:\Users\HP\Desktop\vscode>
```

## Deskripsi program

Program Go ini berfungsi sebagai sistem autentikasi sederhana berbasis token yang menggunakan perulangan tak terbatas. Program ini akan terus-menerus meminta pengguna untuk memasukkan sebuah token . Input token yang dimasukkan oleh pengguna kemudian diperiksa menggunakan pernyataan if untuk dicocokkan dengan nilai tetap (12345abcde). Jika token yang dimasukkan sesuai, program akan mencetak pesan selamat anda berhasil login dan segera menghentikan perulangan dengan perintah break. Sebaliknya, jika token yang dimasukkan salah, program akan mencetak pesan token yang anda masukan salah dan kembali mengulang permintaan input token, memaksa pengguna untuk mencoba lagi hingga token yang benar dimasukkan.

## Tugas 2

### Source code

```
package main
```

```
import "fmt"

func main() {
    var n int

    fmt.Print("masukan bilangan bulat Positif : ")
    fmt.Scan(&n)

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

**Screenshoot program**

```
modul 12 soal 2 > soal 2.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7
8     fmt.Print("masukan bilangan bulat Positif : ")
9     fmt.Scan(&n)
10
11     for n > 0 {
12         digit := n % 10
13         fmt.Println(digit)
14         n = n / 10
15     }
16 }
17
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

masukan bilangan bulat Positif : 1234  
4  
3  
2  
1  
PS C:\Users\HP\Desktop\vscode>

NAMA MUHAMMAD ZAKI AL-FIQF

File Edit View H1 B I ...

NIM : 109082500085  
kelas : if-13-02

Ln 3, Col 17 65 characters AA Formatted 100% Windows (CRLF) UTF-8

### Deskripsi program

Program Go ini berfungsi untuk memisahkan dan mencetak setiap digit dari suatu bilangan bulat positif secara berurutan, dimulai dari digit satuan. Program meminta pengguna memasukkan sebuah bilangan bulat positif  $n$ . Kemudian, program menggunakan perulangan for yang berjalan selama  $n$  lebih besar dari 0. Di setiap iterasi, digit terakhir satuan dari  $n$  diambil menggunakan operasi modulus  $n \bmod 10$  dan langsung dicetak. Setelah itu,  $n$  diperbarui dengan hasil pembagian integer  $n / 10$ , yang secara efektif menghilangkan digit terakhir, sehingga pada iterasi berikutnya dapat memproses digit puluhan, dan seterusnya, sampai seluruh digit selesai diproses.

### Tugas 3

#### Source code

```
package main

import (
    "fmt"
)

func main() {
```

```
var x, y int
hasilbagi := 0

fmt.Print("masukan x dan y : ")
fmt.Scan(&x, &y)

if y == 0 {
    fmt.Print("input y tidak boleh 0")
    return
}

for x >= y {
    x = x - y
    hasilbagi = hasilbagi + 1
}
fmt.Print(hasilbagi)
}
```

**Screenshoot program**

```
3 import (
4     "fmt"
5 )
6
7 func main() {
8
9     var x, y int
10    hasilbagi := 0
11
12    fmt.Print("masukan x dan y : ")
13    fmt.Scan(&x, &y)
14
15    if y == 0 {
16        fmt.Print("input y tidak boleh 0")
17        return
18    }
19
20    for x >= y {
21        x = x - y
22        hasilbagi = hasilbagi + 1
23    }
24
25    fmt.Print(hasilbagi)
26 }
27
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

masukan x dan y : 5 2  
2  
PS C:\Users\HP\Desktop\vscode> go run "c:\Users\HP\Desktop\vscode\main.go"  
masukan x dan y : 15 7  
2  
PS C:\Users\HP\Desktop\vscode>

NAMA MUHAMMAD ZAKI AL-FIQF

File Edit View H1 B I ...

NIM : 109082500085  
kelas : if-13-02

Ln 3, Col 17 65 characters AA Formatted 100% Windows (CRLF) UTF-8

### Deskripsi program

Program Go ini bertujuan untuk mencari hasil pembagian bulat dari dua bilangan yang dimasukkan pengguna melalui metode pengurangan berulang. Pengguna diminta memasukkan dua bilangan bulat positif, x pembilang dan y penyebut. Program terlebih dahulu memastikan penyebut y tidak nol. Jika y valid, program menggunakan perulangan for yang secara terus-menerus **mengurangi x dengan y** dan secara simultan **menghitung berapa kali pengurangan** tersebut terjadi, menyimpan hitungan di variabel hasilbagi. Proses ini berlangsung hingga x menjadi kurang dari y. Pada akhirnya, hasilbagi yang merupakan **hasil bagi** dari pembagian x oleh y dicetak ke layar.