

**LAPORAN PRAKTIKUM ALGORITMA**

**DAN PEMROGRAMAN 1**

**MODUL 12**

**WHILE LOOP**



**Disusun oleh:**

**REZKY FARREL**

**109082500203**

**S1IF-13--02**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## LATIHAN KELAS – GUIDED

### 1. Guided 1

#### Source Code

```
package main

import "fmt"

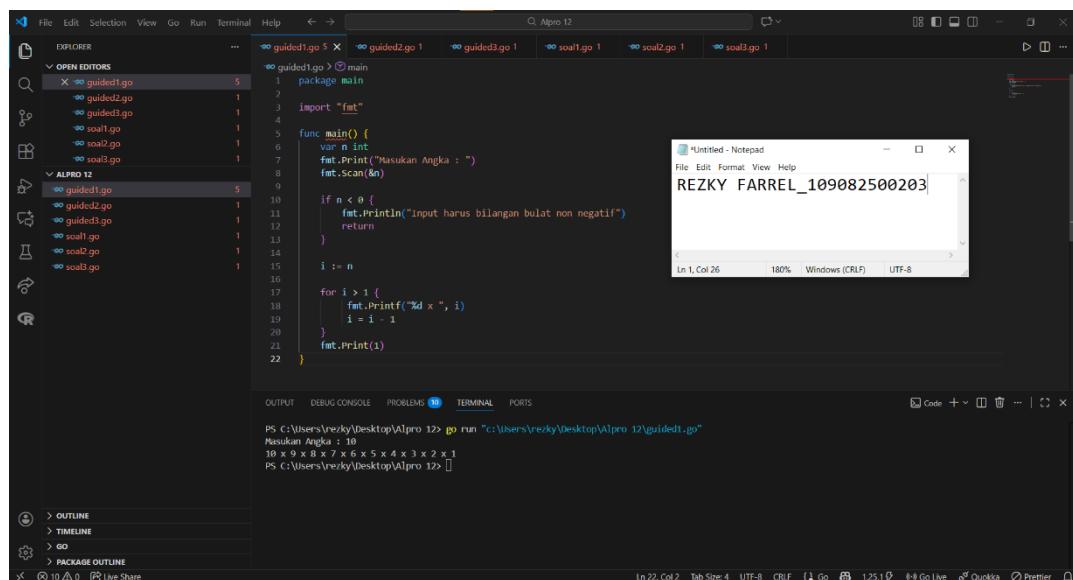
func main() {
    var n int
    fmt.Print("Masukan Angka : ")
    fmt.Scan(&n)

    if n < 0 {
        fmt.Println("Input harus bilangan bulat non negatif")
        return
    }

    i := n

    for i > 1 {
        fmt.Printf("%d x ", i)
        i = i - 1
    }
    fmt.Print(1)
}
```

#### Screenshot program :



#### Deskripsi program :

Program ini bertujuan untuk menampilkan representasi visual urutan perkalian mundur (seperti rumus faktorial) berdasarkan angka yang diinputkan pengguna. Alur program dimulai dengan mendeklarasikan variabel n dan meminta input angka dari pengguna, yang kemudian divalidasi agar tidak bernilai negatif; jika input negatif, program akan menampilkan pesan kesalahan dan berhenti.

## 2. Guided 2

### Source Code

```
package main

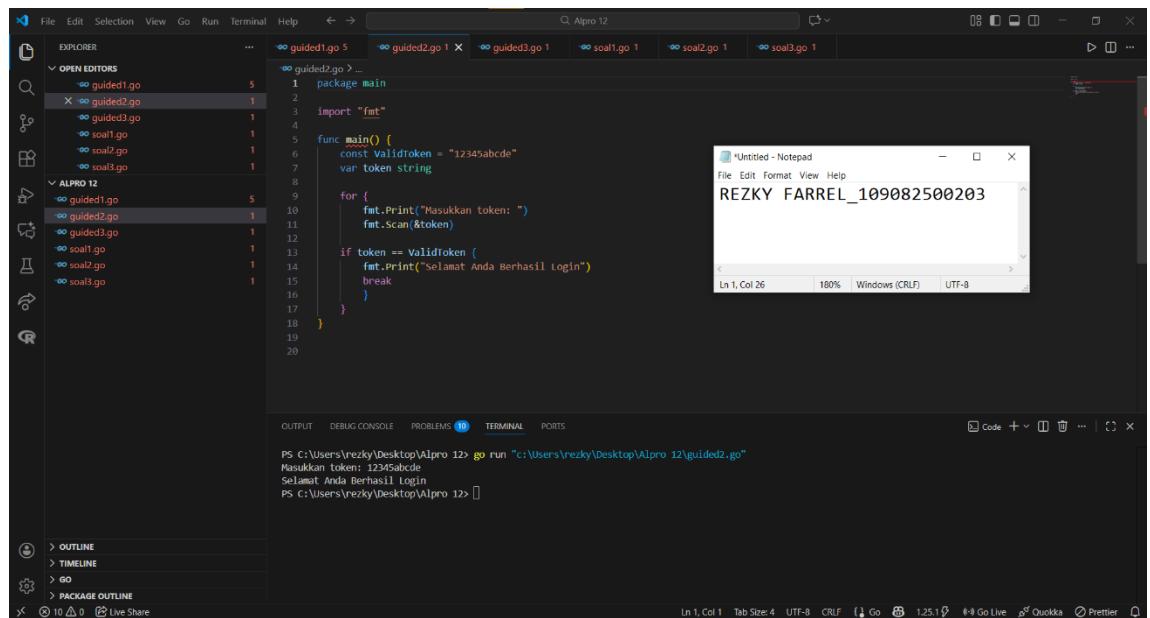
import "fmt"

func main() {
    const ValidToken = "12345abcde"
    var token string

    for {
        fmt.Print("Masukkan token: ")
        fmt.Scan(&token)

        if token == ValidToken {
            fmt.Println("Selamat Anda Berhasil Login")
            break
        }
    }
}
```

### Screenshot program :



### **Deskripsi program :**

Program menetapkan sebuah konstanta bernama ValidToken dengan nilai "12345abcde" sebagai kunci jawaban yang valid. Menggunakan struktur perulangan tanpa henti (infinite loop), program akan terus-menerus meminta pengguna untuk memasukkan token melalui terminal. Setiap input yang dimasukkan akan diperiksa validitasnya; jika input pengguna cocok dengan ValidToken, program akan mencetak pesan keberhasilan login dan mengeksekusi perintah break untuk menghentikan perulangan dan menyelesaikan program, sedangkan jika input salah, program akan terus berulang meminta input kembali sampai pengguna memasukkan token yang benar.

### **Guided 3**

#### **Source Code**

```
package main

import "fmt"

func main() {
    var n int

    fmt.Scan(&n)
    a, b := 0, 1
    i := 0

    for i < n {
        fmt.Print(a, " ")
        a, b=b, a+b
        n--
    }

}
```

#### **Screenshoot program :**

The screenshot shows the Alpro 12 IDE interface. On the left, the Explorer sidebar lists files: guided1.go, guided2.go, guided3.go, soal1.go, soal2.go, and soal3.go. The guided3.go editor tab is active, displaying the following Go code:

```
package main
import "fmt"
func main() {
    var n int
    fmt.Scan(&n)
    a, b := 0, 1
    i := 0
    for i < n {
        fmt.Print(a, " ")
        a, b = a+b, a
        i++
    }
}
```

The Terminal window at the bottom shows the command `go run "c:\Users\rezky\Desktop\Alpro 12\guided3.go"` being run, followed by the output of the Fibonacci sequence: 0 1 1 2 3 5 8 13 21 34.

A Notepad window titled "Untitled - Notepad" contains the text: "REZKY FARREL\_109082500203".

### Deskripsi program :

Program ini dirancang untuk menghasilkan dan menampilkan deret bilangan Fibonacci sebanyak jumlah yang ditentukan oleh input pengguna. Dimulai dengan inisialisasi dua variabel dasar a bernilai 0 dan b bernilai 1, program menggunakan struktur perulangan for yang berjalan selama nilai input n masih lebih besar dari i (yang bernilai 0). Di setiap iterasi, program mencetak nilai a saat ini, kemudian memperbarui variabel dengan logika Fibonacci (a mengambil nilai b, dan b menjadi hasil penjumlahan a dan b sebelumnya), serta mengurangi nilai n sebanyak satu (n--) pada setiap putaran agar perulangan berhenti tepat setelah mencetak jumlah deret yang diminta.

## TUGAS

### 1. Tugas 1

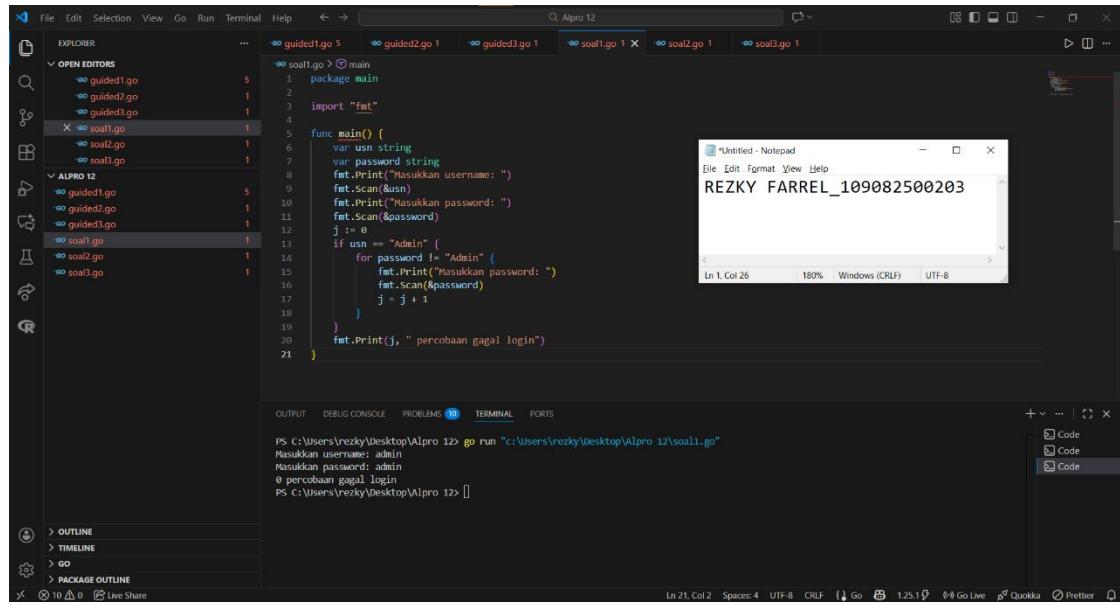
#### Source code

```
package main

import "fmt"

func main() {
    var usn string
    var password string
    fmt.Print("Masukkan username: ")
    fmt.Scan(&usn)
    fmt.Print("Masukkan password: ")
    fmt.Scan(&password)
    j := 0
    if usn == "Admin" {
        for password != "Admin" {
            fmt.Print("Masukkan password: ")
            fmt.Scan(&password)
            j = j + 1
        }
    }
    fmt.Println(j, " percobaan gagal login")
}
```

#### Screenshot program :



#### Deskripsi program :

Program ini berfungsi untuk aplikasi simulasi login sederhana yang dirancang untuk memvalidasi kredensial pengguna dan menghitung frekuensi kesalahan kata sandi khusus untuk akun "Admin". Alur program dimulai dengan meminta input username dan password dari pengguna, kemudian memverifikasi apakah username tersebut adalah "Admin". Jika username sesuai namun password yang dimasukkan salah (bukan "Admin"), program akan memasuki struktur perulangan yang terus-menerus meminta input password baru hingga pengguna memasukkan kata sandi yang benar.

## Tugas 2

### Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)

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

### Screenshot program

The screenshot shows a Go development environment with the following details:

- File Explorer:** Shows files in the current workspace, including `guided1.go`, `guided2.go`, `guided3.go`, `soal1.go`, `soal2.go`, and `soal3.go`.
- Code Editor:** The `soal2.go` file is open, displaying the provided Go code.
- Terminal:** The terminal window shows the command `go run "c:/Users/rezky/Desktop\Alpro 12\soal2.go"` and the output of the program, which asks for a number and prints its digits.
- Output Window:** Shows the command and the resulting digit output.
- Notepad:** A separate window titled "Untitled - Notepad" contains the text "REZKY FARREL\_109082500203".

### Deskripsi program

Program ini berfungsi untuk memisahkan dan menampilkan setiap digit dari sebuah bilangan bulat secara terbalik, dimulai dari digit paling belakang (kanan) menuju ke depan. Setelah pengguna memasukkan angka input, program menjalankan struktur perulangan for yang terus bekerja selama nilai angka tersebut masih lebih besar dari 0.

### Tugas 3

#### Source code

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Scan(&x, &y)

    hasil := 0

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

    fmt.Println(hasil)
}
```

#### Screenshot program :

The screenshot shows a Go development environment with the following components:

- File Explorer:** Shows files in the current workspace, including `guided1.go`, `guided2.go`, `guided3.go`, `soal1.go`, `soal2.go`, and `soal3.go`.
- Code Editor:** The `soal3.go` file is open, displaying the provided Go code.
- Terminal:** The terminal window shows the command `go run c:/Users/rezky/Desktop/Alpro 12/soal3.go` being run, with the output showing the number 10.
- Output Window:** The output window shows the result `REZKY FARREL_109082500203`.
- Status Bar:** Shows the current file is `soal3.go`, with 17 lines of code, and other status information like file encoding and line numbers.

#### Deskripsi program

Program ini untuk melakukan operasi pembagian bilangan bulat (integer division) dengan menggunakan metode pengurangan berulang. Alur program dimulai dengan meminta input dua angka dari pengguna, yaitu  $x$  sebagai bilangan yang akan dibagi dan  $y$  sebagai pembagi, serta menginisialisasi variabel hasil dengan nilai 0. Inti dari program ini terletak pada struktur perulangan for yang terus berjalan selama nilai  $x$  masih lebih besar atau sama dengan  $y$ . Di setiap iterasi perulangan, nilai  $x$  dikurangi sebesar nilai  $y$  dan variabel hasil bertambah satu, yang secara efektif menghitung berapa kali  $y$  dapat diambil dari  $x$ .