

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL [12]  
[while loop]**



**Disusun oleh:**

**[Muhammad Nabil Raissa P]**

**[109082500127]**

**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

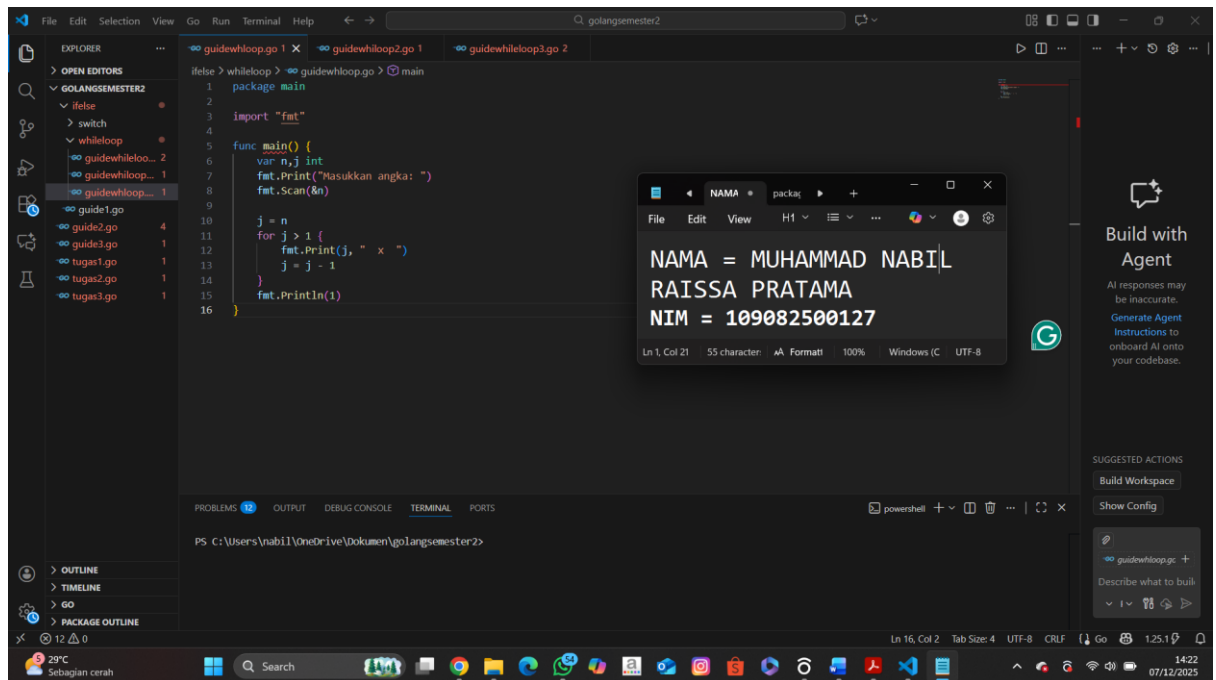
```
package main

import "fmt"

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

    j = n
    for j > 1 {
        fmt.Print(j, " x ")
        j = j - 1
    }
    fmt.Println(1)
}
```

### Screenshoot program



## Deskripsi program

Program tersebut berfungsi untuk menampilkan deret perkalian secara menurun dari angka yang diinput hingga angka 1. Nilai awal disimpan pada variabel `j`, lalu diturunkan satu per satu melalui perulangan `for` hingga mencapai 1. Hasil akhirnya berupa tampilan format perkalian seperti `n x (n-1) x ... x 1`, tanpa menghitung hasil faktorialnya.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

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

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

    for {
        if token == validToken {
```

```

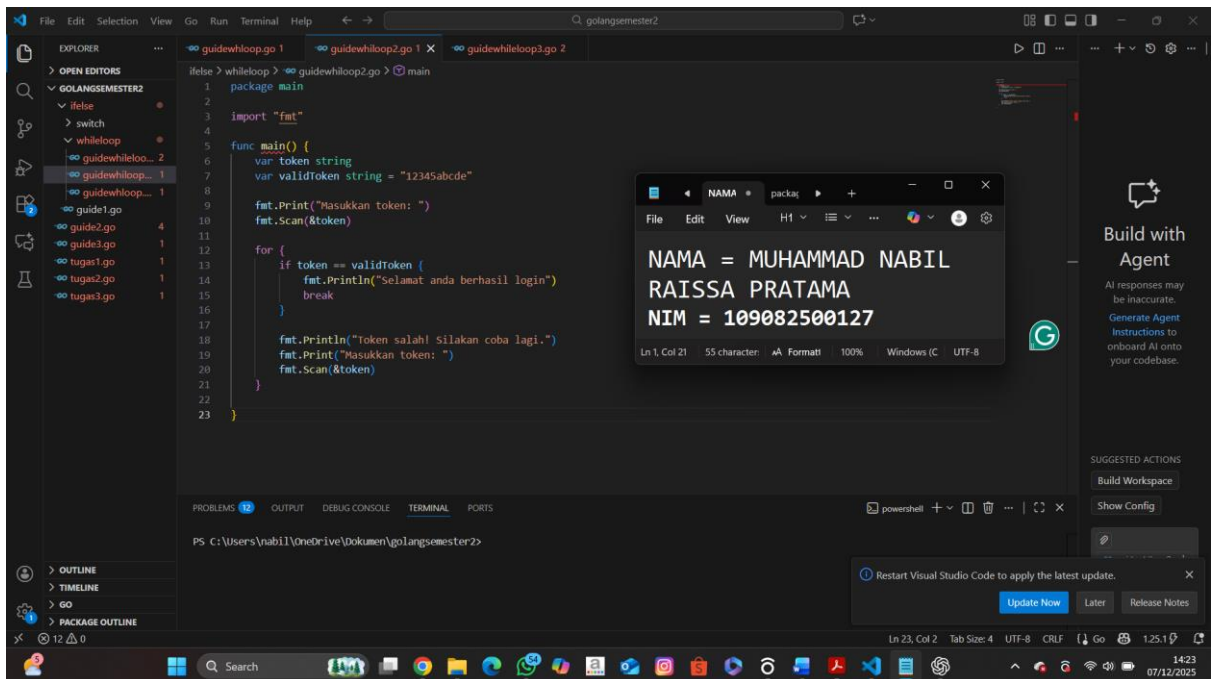
        fmt.Println("Selamat anda berhasil login")
        break
    }

    fmt.Println("Token salah! Silakan coba lagi.")
    fmt.Print("Masukkan token: ")
    fmt.Scan(&token)
}

}

```

### Screenshoot program



### Deskripsi program

Program ini digunakan untuk melakukan proses login menggunakan token. Sistem menyimpan token yang benar dalam variabel `validToken`. Pengguna diminta memasukkan token, lalu program mengecek apakah token tersebut sesuai. Jika cocok, muncul pesan berhasil login dan perulangan berhenti. Jika tidak cocok, program akan terus meminta input ulang hingga pengguna memasukkan token yang benar.

### 3. Guided 3 Source Code

```

package main

import "fmt"

func main() {
    var n int

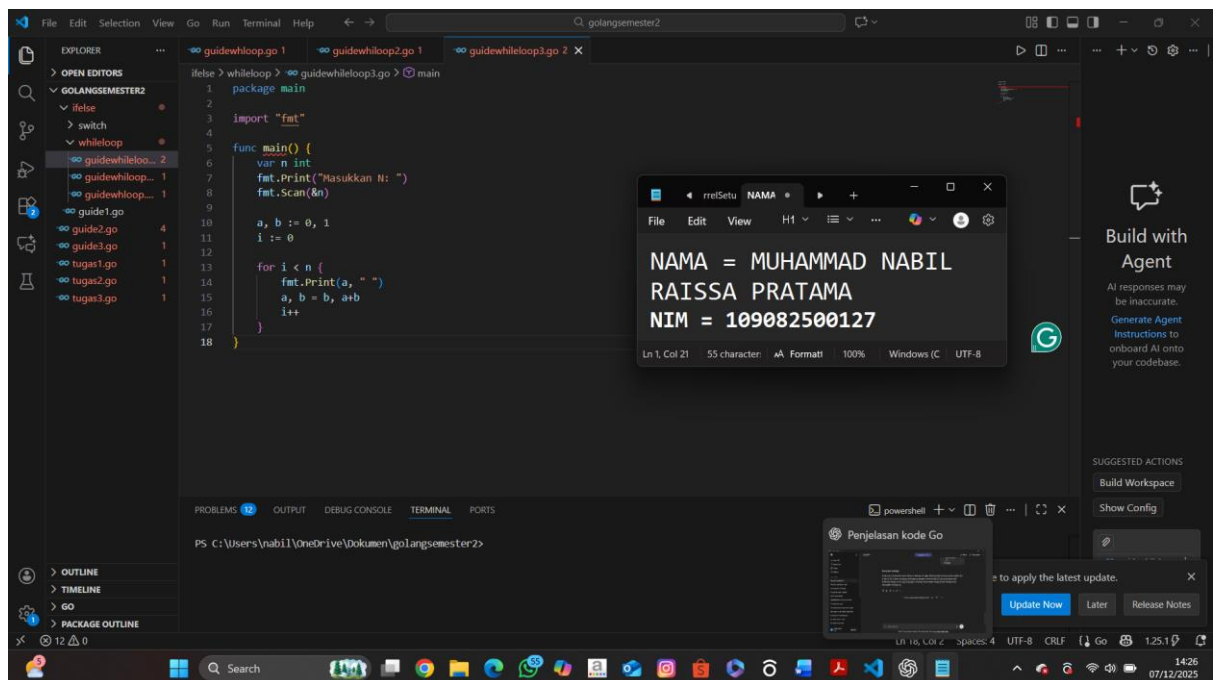
    fmt.Print("Masukkan N: ")
    fmt.Scan(&n)

    a, b := 0, 1
    i := 0

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

```

## Screenshoot program



### Deskripsi program

Program ini menampilkan deret Fibonacci sebanyak n angka. Nilai awal deret disimpan pada variabel a = 0 dan b = 1. Selama perulangan berlangsung, program mencetak nilai a, lalu memperbarui nilai berikutnya dengan rumus a, b = b, a+b. Perulangan terus berjalan hingga jumlah bilangan yang ditampilkan mencapai n.

### TUGAS

#### 1. Tugas 1

##### Source code

```
package main

import "fmt"

func main() {
    const validUser, validPass = "Admin", "Admin"

    var username, password string
    gagal := 0

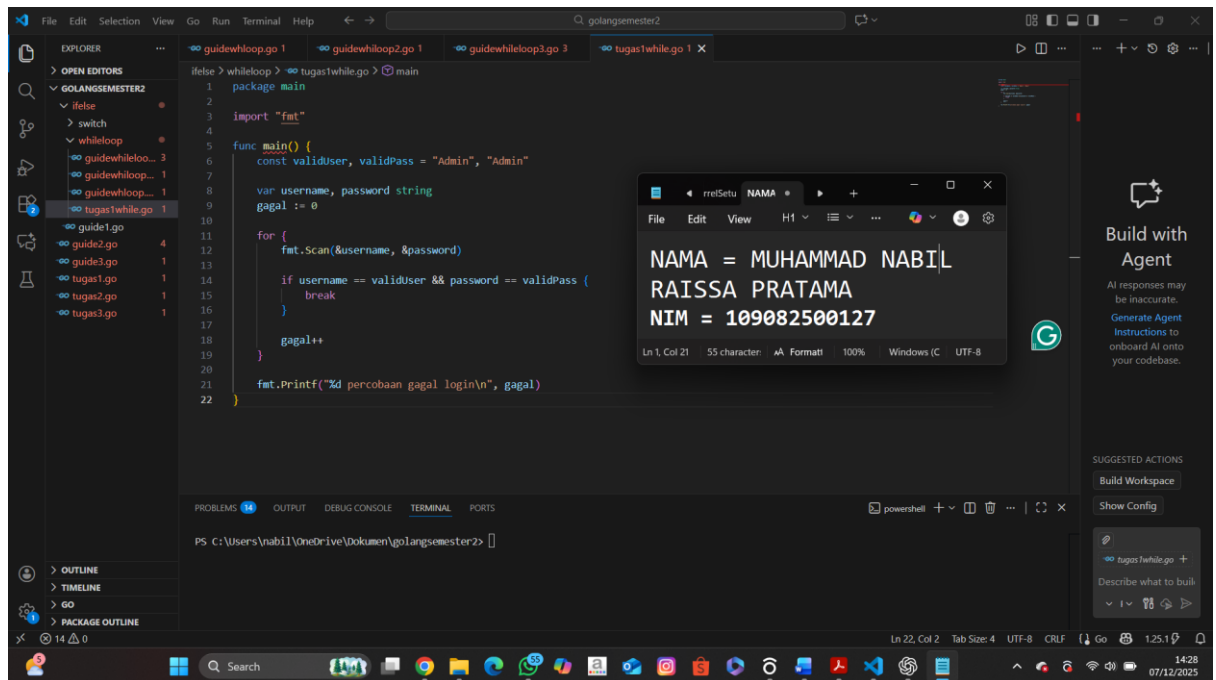
    for {
        fmt.Scan(&username, &password)

        if username == validUser && password == validPass {
            break
        }

        gagal++
    }

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

##### Screenshoot program



## Deskripsi program

Program tersebut berfungsi untuk melakukan proses login menggunakan username dan password secara berulang. Sistem menyimpan data login yang valid pada variabel konstan validUser dan validPass. Pengguna diminta memasukkan username dan password melalui input. Jika keduanya sesuai dengan nilai valid, perulangan berhenti. Jika tidak sesuai, sistem terus meminta input ulang dan setiap kegagalan login dicatat melalui variabel gagal. Setelah login berhasil, program menampilkan jumlah percobaan gagal yang terjadi sebelum login benar.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {

    var bilangan, digit int

    fmt.Print("Masukkan bilangan = ")

    fmt.Scan(&bilangan)
```

```

for bilangan > 0 {

    digit = bilangan % 10

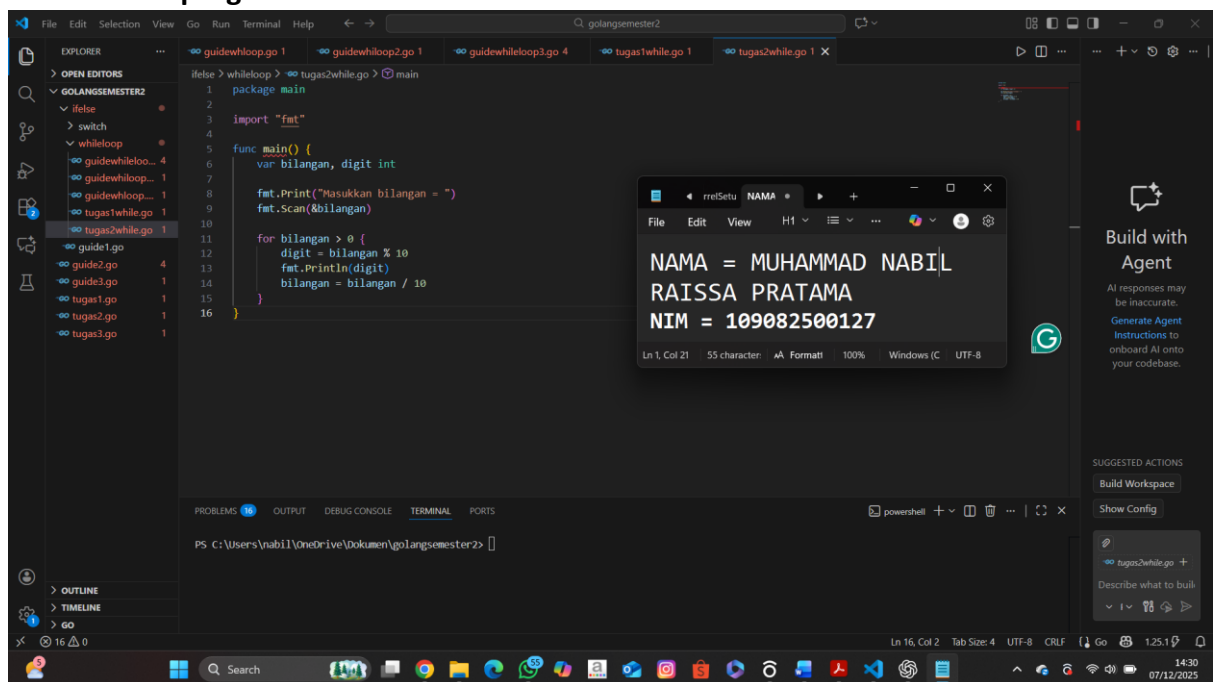
    fmt.Println(digit)

    bilangan = bilangan / 10

}

```

### Screenshoot program





```

func main() {
    var x, y int

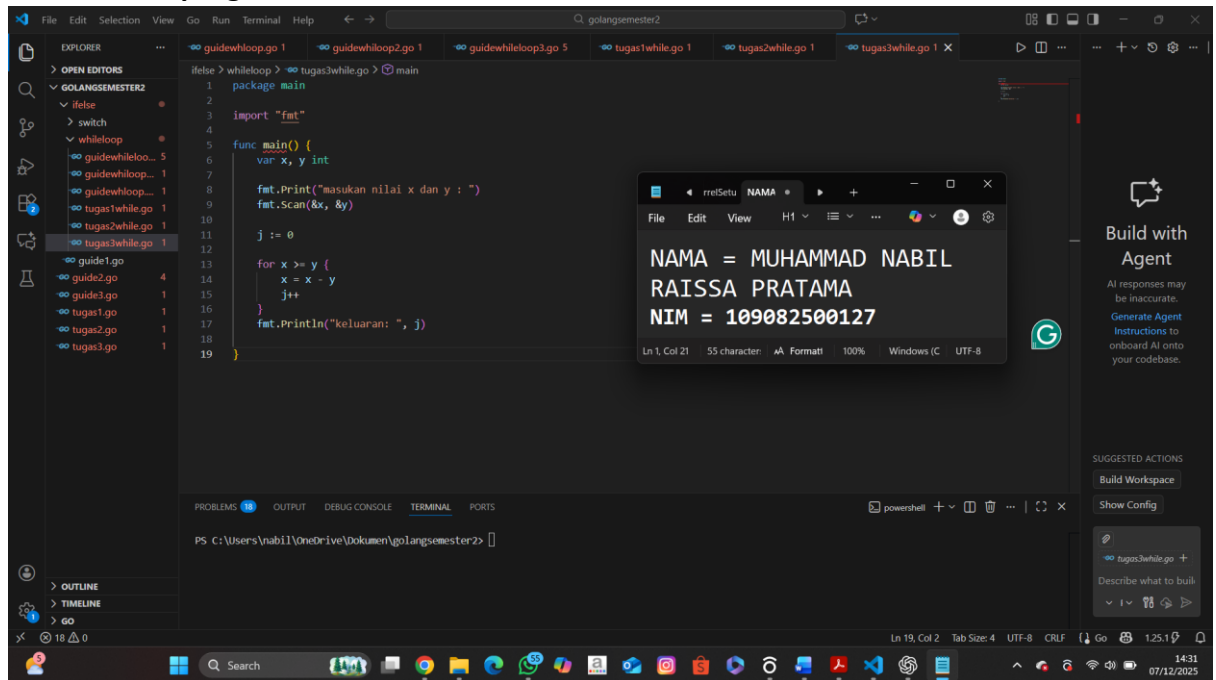
    fmt.Print("masukan nilai x dan y : ")
    fmt.Scan(&x, &y)

    j := 0

    for x >= y {
        x = x - y
        j++
    }
    fmt.Println("keluaran: ", j)
}

```

### Screenshoot program



### Deskripsi program

Program ini berfungsi untuk melakukan pembagian secara manual tanpa menggunakan operator pembagian. Pengguna memasukkan nilai x sebagai bilangan yang akan dibagi dan y sebagai pembaginya. Selama nilai x masih lebih besar atau sama dengan y, program akan terus mengurangi x dengan y dan menghitung berapa kali pengurangan terjadi melalui variabel j. Hasil akhirnya menampilkan nilai j sebagai banyaknya pengurangan, yang merepresentasikan hasil pembagian bilangan x oleh y dalam bentuk bilangan bulat (tanpa sisa).