

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL 12

DO WHILE



Disusun oleh:

Andra Dwicki Saputra

109082500206

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 ")

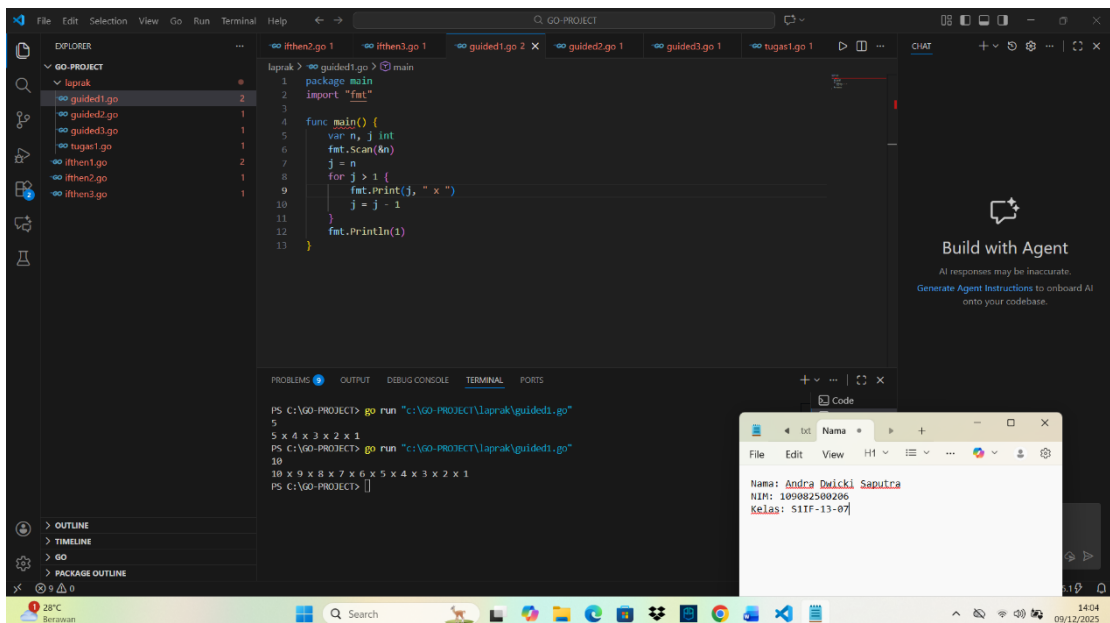
        j = j - 1

    }

    fmt.Println(1)

}
```

Screenshoot program



Deskripsi program

program tersebut berfungsi untuk menampilkan urutan perkalian turun dari suatu bilangan sampai angka 1, mirip seperti cara menuliskan proses faktorial secara manual. Pertama, program meminta pengguna memasukkan sebuah angka melalui `fmt.Scan(&bilangan)`. Lalu nilai tersebut disalin ke variabel `j`, yang akan digunakan sebagai penghitung dalam perulangan. Program kemudian menjalankan perulangan `for j > 1`, yang berarti selama nilai `j` masih lebih besar dari 1, program akan mencetak nilai `j` diikuti tanda " x ", lalu nilai `j` dikurangi satu setiap iterasi. Ketika perulangan selesai (karena `j` sudah menjadi 1), program mencetak angka 1 sebagai bagian akhir dari urutan. Hasil akhirnya adalah tampilan rangkaian seperti `5 x 4 x 3 x 2 x 1` sesuai bilangan yang dimasukkan pengguna.

2. Guided 2

Source Code

```
package main

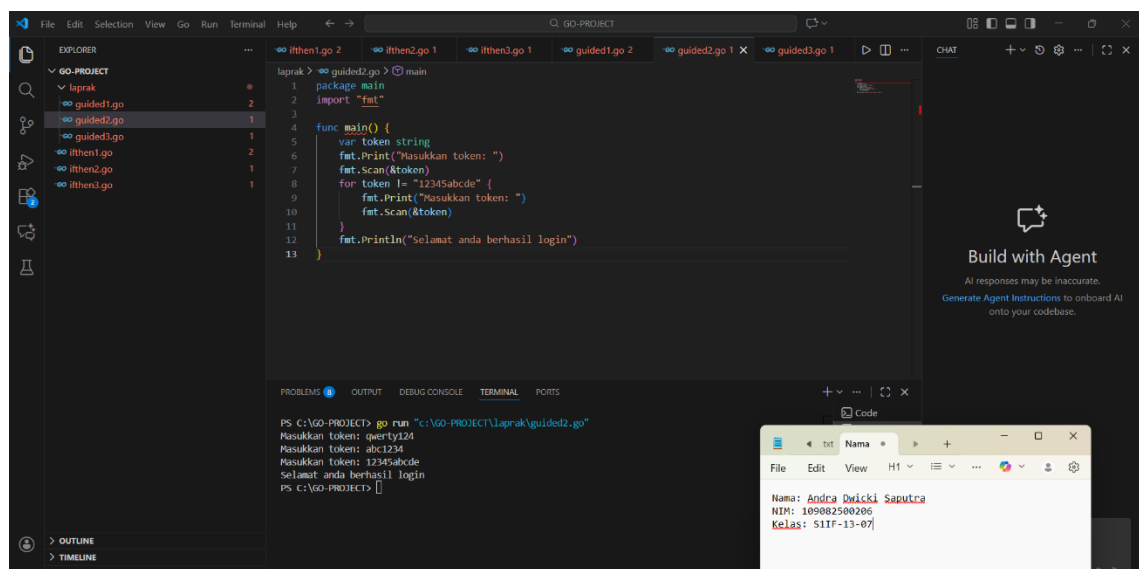
import "fmt"

func main() {
    var token string
    fmt.Print("Masukkan token: ")
    fmt.Scan(&token)

    for token != "12345abcde" {
        fmt.Print("Masukkan token: ")
        fmt.Scan(&token)
    }

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

Screenshoot program



Deskripsi program

program tersebut digunakan untuk melakukan validasi token hingga pengguna memasukkan token yang benar. Pertama, program meminta input token melalui `fmt.Scan(&token)`. Setelah itu, program menjalankan perulangan `for` dengan kondisi `token != "12345abcde"`, artinya selama token yang dimasukkan tidak sama dengan token yang benar, program akan terus meminta input ulang dari pengguna. Setiap kali token salah, program menampilkan pesan “Masukan Token : ” lalu membaca input berikutnya. Ketika akhirnya pengguna memasukkan token yang tepat, perulangan berhenti dan program menampilkan pesan “Selamat anda berhasil login”, yang menandakan validasi berhasil.

3. Guided 3

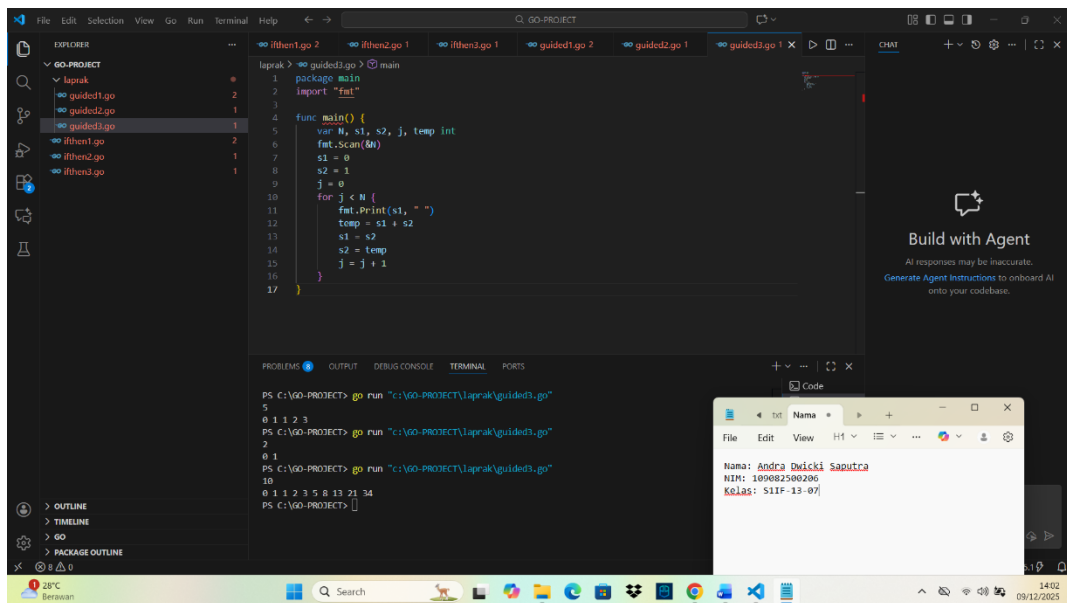
Source Code

```
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.Print(s1, " ")
        temp = s1 + s2
        s1 = s2
        s2 = temp
        j = j + 1
    }
}
```

Screenshoot program



The screenshot shows a Go IDE with a project named 'GO-PROJECT'. The Explorer pane on the left shows the project structure with files like 'laprak', 'guided1.go', 'guided2.go', and 'guided3.go'. The main editor displays the code for 'guided3.go', which is a Go program to calculate the Fibonacci sequence. The code is as follows:

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

The terminal at the bottom shows the execution of the program. It prompts for input '5' and outputs the Fibonacci sequence '0 1 1 2 3'. The output is as follows:

```
PS C:\GO-PROJECT> go run "c:\GO-PROJECT\laprak\guided3.go"
5
0 1 1 2 3
PS C:\GO-PROJECT> go run "c:\GO-PROJECT\laprak\guided3.go"
2
0 1
PS C:\GO-PROJECT> go run "c:\GO-PROJECT\laprak\guided3.go"
10
0 1 1 2 3 5 8 13 21 34
PS C:\GO-PROJECT>
```

A small window titled 'Nama' is also visible, containing the following text:

```
Nama: Andra Dwicki Saputra
NIM: 180902500206
Kelas: SIIF-13-07
```

Deskripsi program

program tersebut menampilkan deret Fibonacci sebanyak N angka berdasarkan input dari pengguna. Pertama, program membaca nilai N , yaitu jumlah angka Fibonacci yang ingin ditampilkan. Dua variabel awal, $s1 = 0$ dan $s2 = 1$, digunakan sebagai dua angka pertama dalam deret Fibonacci. Selanjutnya, program menjalankan perulangan $\text{for } j < N$ untuk mencetak setiap angka Fibonacci dimulai dari $s1$. Di dalam loop, program menghitung angka berikutnya menggunakan $\text{temp} = s1 + s2$, lalu menggeser nilai: $s1$ menjadi $s2$, dan $s2$ menjadi angka baru temp . Proses ini diulang sampai jumlah angka yang dicetak mencapai N , sehingga menghasilkan deret Fibonacci sesuai permintaan.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    const Username = "Admin"
    const Password = "Admin"

    var username, password string
    var gagal int

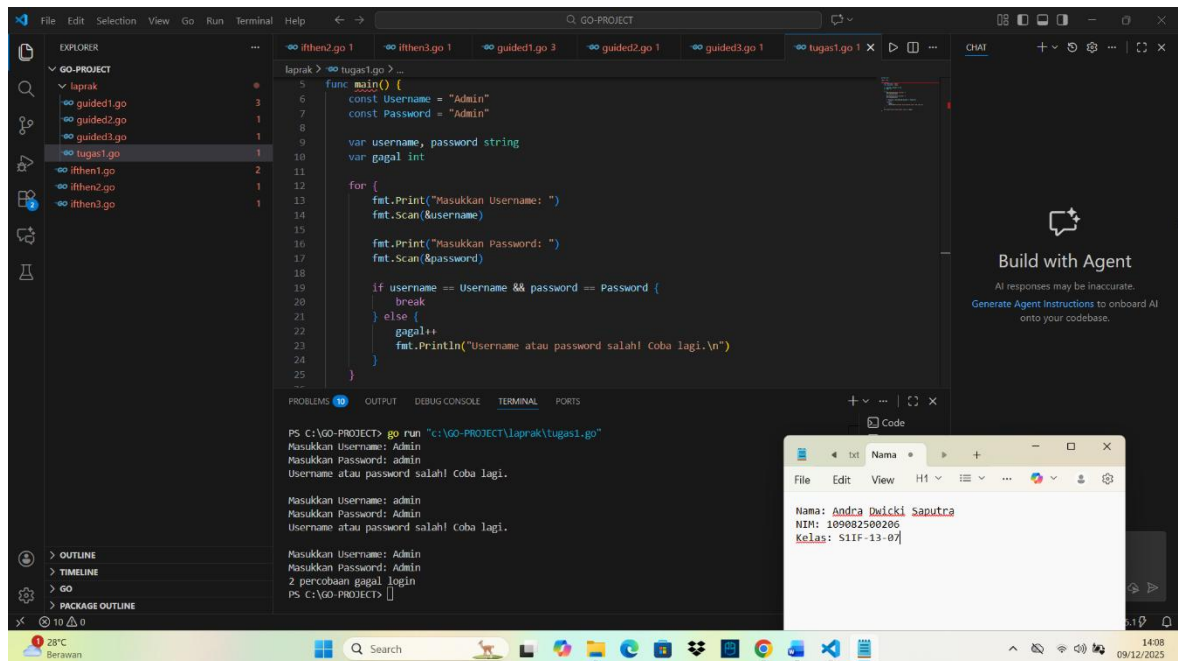
    for {
        fmt.Print("Masukkan Username: ")
        fmt.Scan(&username)

        fmt.Print("Masukkan Password: ")
        fmt.Scan(&password)

        if username == Username && password == Password {
            break
        } else {
            gagal++
            fmt.Println("Username atau password salah! Coba lagi.\n")
        }
    }

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

Screenshoot program



Deskripsi program

Program tersebut digunakan untuk membuat sistem login sederhana dengan batas percobaan yang dihitung. Pertama, program menyimpan username dan password yang benar dalam konstanta `Username` dan `Password`. Lalu program meminta input username dan password dari pengguna dalam sebuah perulangan `for` tanpa batas. Setiap kali pengguna memasukkan data, program mengecek apakah keduanya sama dengan nilai yang benar; jika cocok, loop akan berhenti menggunakan `break`, namun jika salah, variabel `gagal` akan bertambah satu dan program menampilkan pesan "Username atau password salah!". Setelah login berhasil, program mencetak jumlah percobaan gagal yang telah dilakukan sebelum pengguna memasukkan username dan password yang benar.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var angka int

    fmt.Print("Masukkan bilangan : ")

    fmt.Scan(&angka)

    for angka > 0 {

        digit := angka % 10

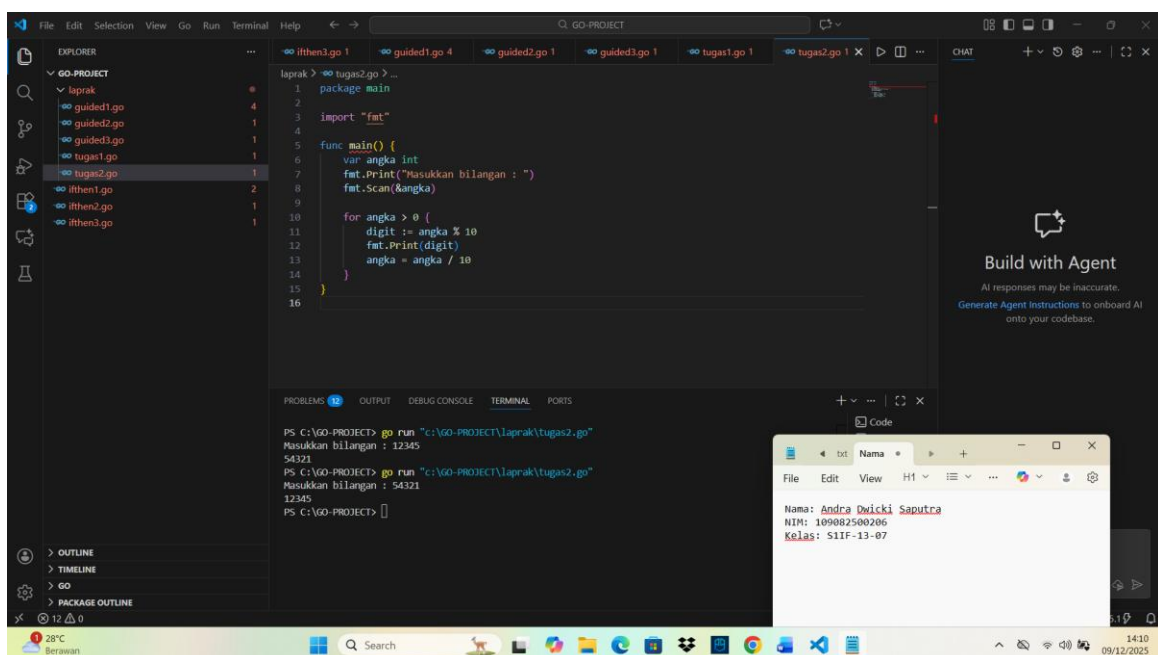
        fmt.Print(digit)

        angka = angka / 10

    }

}
```

Screenshoot program



Deskripsi program

program tersebut digunakan untuk membalik urutan digit dari sebuah bilangan dengan cara mengambil angka dari belakang. Pertama, program meminta pengguna memasukkan sebuah bilangan melalui `fmt.Scan(&angka)`. Setelah itu, program menjalankan perulangan `for` `angka > 0`, yang berarti loop terus berjalan selama nilai angka masih lebih besar dari 0. Di dalam loop, program mengambil digit terakhir dengan rumus `angka % 10` dan langsung mencetaknya sehingga digit paling belakang muncul terlebih dahulu. Setelah dicetak, angka diperkecil dengan `angka = angka / 10` untuk menghapus digit terakhir tersebut. Proses ini berulang sampai semua digit habis, sehingga hasil akhirnya adalah urutan digit bilangan yang terbalik.

3. Tugas 3

Source code

```
package main

import "fmt"

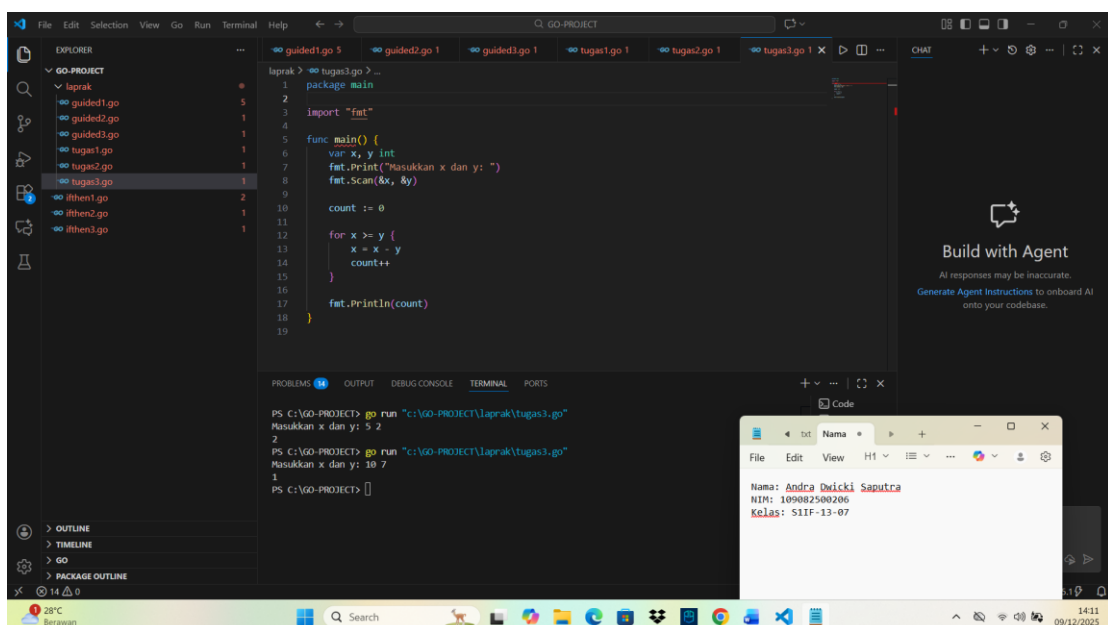
func main() {
    var x, y int
    fmt.Print("Masukkan x dan y: ")
    fmt.Scan(&x, &y)

    count := 0

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

    fmt.Println(count)
}
```

Screenshoot program



Deskripsi program

Program tersebut digunakan untuk menghitung berapa kali nilai y dapat dikurangkan dari x sebelum x menjadi lebih kecil dari y . Pertama, program meminta pengguna memasukkan dua bilangan, yaitu x dan y , melalui `fmt.Scan(&x, &y)`. Selanjutnya, variabel `count` digunakan untuk menghitung jumlah pengurangan yang berhasil dilakukan. Pada bagian perulangan `for x >= y`, program akan terus melakukan $x = x - y$ selama nilai x masih lebih besar atau sama dengan y , dan setiap pengurangan akan menambah nilai `count` sebesar 1. Ketika x sudah lebih kecil dari y , perulangan berhenti, dan program menampilkan nilai `count`, yang merupakan jumlah total pengurangan. Program ini pada dasarnya meniru proses pembagian tetapi menggunakan pengurangan berulang untuk mendapatkan hasilnya.