

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 12**

**WHILE-LOOP**



**Disusun oleh:**

**NAUFAL BINTANG PRATAMA**

**109082500096**

**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 int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)

    if n == 0 {
        fmt.Println("1")
        return
    }

    for i := n; i >= 1; i-- {
        if i == 1 {
            fmt.Printf("%d", i)
        } else {
            fmt.Printf("%d x ", i)
        }
    }

    fmt.Println()
}
```

### Screenshoot program

```
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var n int
9     fmt.Print("Masukkan bilangan: ")
10    fmt.Scan(&n)
11
12    if n == 0 {
13        fmt.Println("1")
14        return
15    }
16
17    for i := n; i >= 1; i-- {
18        if i == 1 {
19            fmt.Printf("%d", i)
20        } else {
21            fmt.Printf("%d x ", i)
22        }
23    }
24    fmt.Println()
25 }
```

```
PS D:\go> go run .\modul12\guide1mod12\guide1mod12.go
Masukkan bilangan: 5
5 x 4 x 3 x 2 x 1
PS D:\go> go run .\modul12\guide1mod12\guide1mod12.go
Masukkan bilangan: 10
10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1
PS D:\go> go run .\modul12\guide1mod12\guide1mod12.go
Masukkan bilangan: 0
1
PS D:\go>
```

## Deskripsi program

Program ini berfungsi untuk **menampilkan susunan perkalian berurutan dari suatu bilangan hingga** **1.**

Pengguna diminta memasukkan sebuah bilangan n, kemudian program akan mencetak bentuk perkalian dimulai dari n turun hingga angka 1.

Program terlebih dahulu memeriksa apakah nilai n bernilai 0. Jika ya, program langsung menampilkan angka 1 sebagai hasil dan menghentikan proses. Jika n lebih besar dari 0, program menjalankan perulangan dari n ke 1 secara menurun. Pada setiap iterasi, program menampilkan angka saat ini. Tanda x dicetak setelah angka selama angka tersebut belum mencapai 1, agar susunan perkalian terlihat rapi.

Program ini menunjukkan penggunaan struktur kendali if dan perulangan for untuk menghasilkan format tampilan perkalian secara terurut.

## 2. Guided 2 Source Code

```
package main
```

```

import "fmt"

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

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

        if token == validToken {
            fmt.Println("Selamat Anda berhasil login")
            break
        }
    }
}

```

### Screenshoot program

The screenshot shows the Go program being executed in a terminal. The output is as follows:

```

PS D:\go> go run .\modul12\guide2mod12\guide2mod12.go
Masukkan token: abcde12345
Masukkan token: 12345abcde
Selamat Anda berhasil login
PS D:\go>

```

### Deskripsi program

Program ini berfungsi untuk melakukan proses **validasi token login**. Program meminta pengguna memasukkan sebuah token melalui input. Token yang dimasukkan akan dibandingkan dengan token yang dianggap valid, yaitu "12345abcde".

Program menggunakan perulangan for tanpa batas, sehingga akan terus meminta input token selama token yang diberikan belum sesuai. Jika token yang dimasukkan masih salah, program akan kembali meminta input tanpa menampilkan pesan apa pun. Ketika pengguna memasukkan token yang benar, program menampilkan pesan **"Selamat Anda berhasil login"** dan menghentikan perulangannya dengan perintah break.

Program ini menerapkan konsep **kontrol input, perulangan, serta pengecekan kondisi (percabangan)** untuk autentikasi sederhana.

### 3. Guided 3

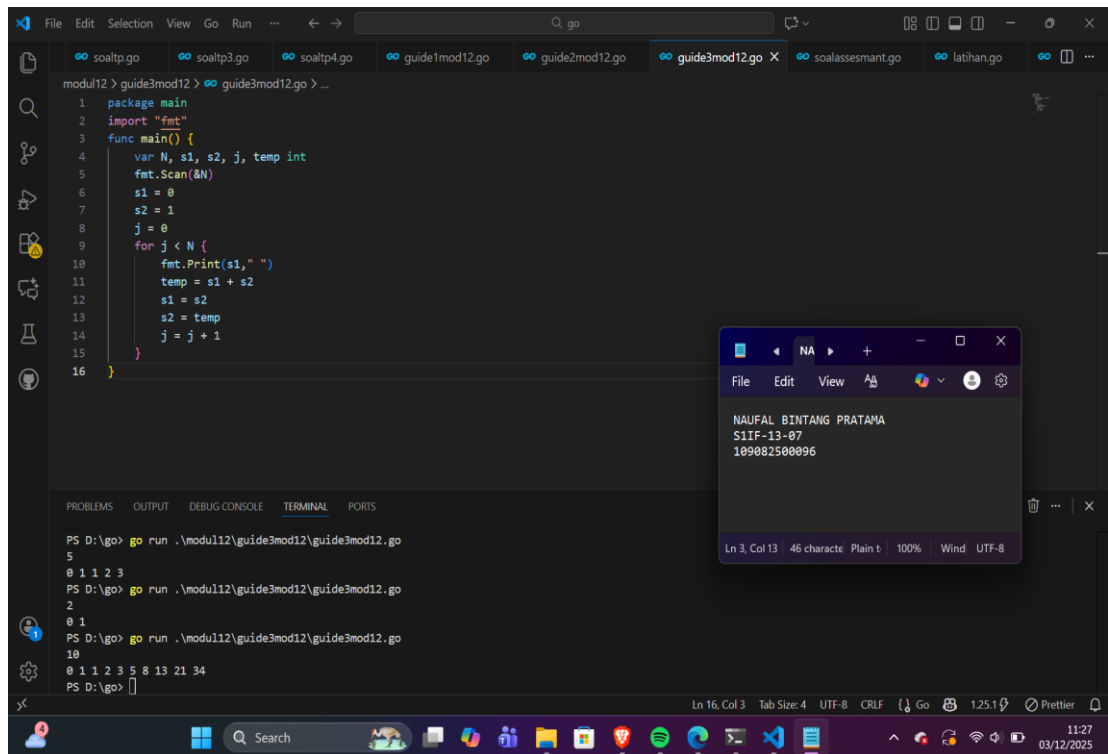
#### Source Code

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

#### Screenshoot program



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

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\go> go run .\modul12\guide3mod12\guide3mod12.go  
5  
0 1 1 2 3

PS D:\go> go run .\modul12\guide3mod12\guide3mod12.go  
10  
0 1 1 2 3 5 8 13 21 34

PS D:\go> go run .\modul12\guide3mod12\guide3mod12.go  
34  
0 1 1 2 3 5 8 13 21 34

NAUFAL BINTANG PRATAMA  
S11F-13-07  
109082500096

## Deskripsi program

Program ini berfungsi untuk menampilkan **N bilangan pertama dari deret Fibonacci** sesuai jumlah yang dimasukkan oleh pengguna. Deret Fibonacci adalah urutan bilangan yang dimulai dari **0 dan 1**, dan setiap bilangan berikutnya merupakan hasil penjumlahan dari dua bilangan sebelumnya.

Program akan meminta input berupa nilai N, kemudian menggunakan perulangan for untuk mencetak setiap bilangan Fibonacci satu per satu. Pada setiap langkah perulangan, program menampilkan nilai s1 sebagai bilangan saat ini, kemudian menghitung bilangan berikutnya menggunakan rumus  $s1 + s2$ . Setelah itu, nilai variabel diperbarui agar dapat menghasilkan angka Fibonacci berikutnya hingga jumlah yang diminta tercapai.

Program ini menunjukkan penggunaan perulangan serta pemrosesan variabel secara berurutan untuk membentuk pola angka matematika.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import (
```

```
"fmt"

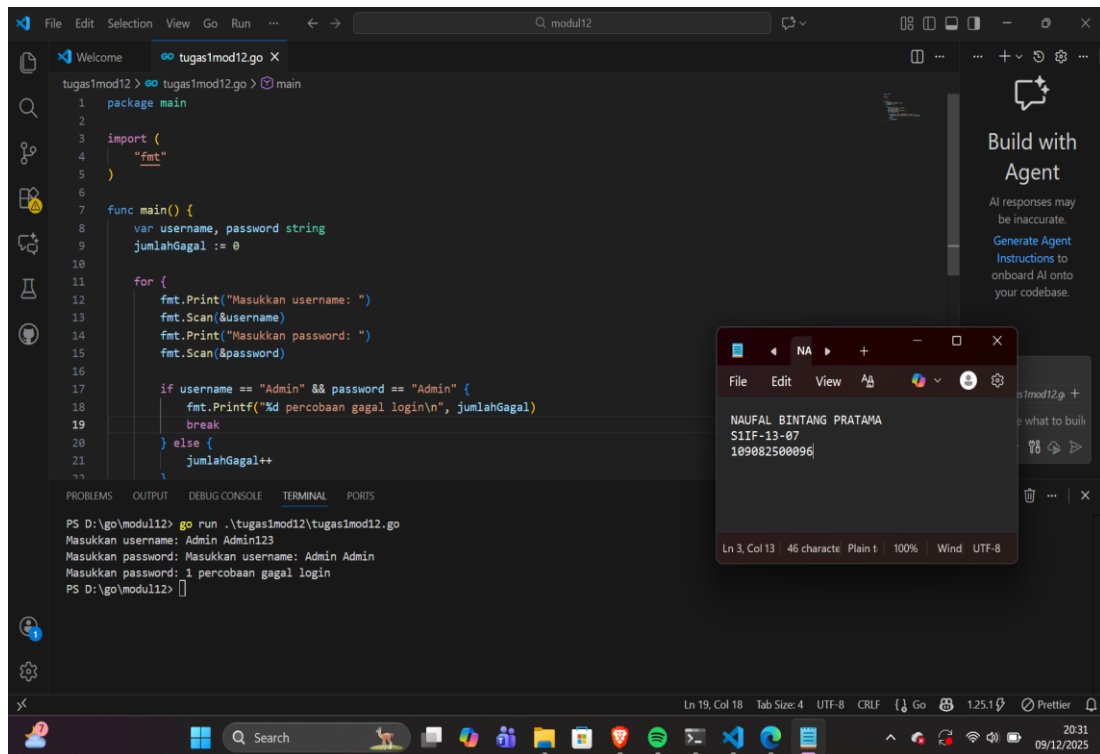
)

func main() {
var username, password string
jumlahGagal := 0

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

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

**Screenshoot program**



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

PS D:\go\modul12> go run .\tugas1mod12\tugas1mod12.go  
Masukkan username: Admin Admin123  
Masukkan password: Masukkan username: Admin Admin  
Masukkan password: 1 percobaan gagal login  
PS D:\go\modul12>

NAUFAL BINTANG PRATAMA  
S1IF-13-07  
109082500096

## Deskripsi program

Program ini berfungsi untuk menghitung jumlah percobaan login yang gagal ketika pengguna salah memasukkan username dan password. Program akan meminta input username dan password secara berulang di dalam sebuah perulangan. Setiap kali pengguna memasukkan kombinasi yang salah, program akan menambahkan jumlah percobaan gagal dan kemudian meminta pengguna untuk memasukkan ulang data login. Proses perulangan akan terus berlangsung sampai pengguna memasukkan username dan password yang benar sesuai ketentuan. Setelah login berhasil, program akan menampilkan total percobaan gagal yang telah dilakukan sebelumnya sebagai informasi bahwa sistem mencatat kesalahan input login.

## 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    // ambil digit terakhir

    fmt.Println(digit)    // cetak digit

    angka = angka / 10    // buang digit terakhir

}

}

```

### Screenshoot program

The screenshot shows the Visual Studio Code editor with a Go file named `tugas2mod12.go`. The code implements a loop that repeatedly takes the last digit of a number using the modulus operator (`%`) and prints it, then divides the number by 10 until it reaches zero. The terminal at the bottom shows the execution of the program twice. In the first run, the input is 2 and the output is 2. In the second run, the input is 2544 and the output is 4, 5, 2, 4, demonstrating the reverse order of digits.

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var angka int
7     fmt.Print("Masukkan bilangan: ")
8     fmt.Scan(&angka)
9
10    for angka > 0 {
11        digit := angka % 10
12        fmt.Println(digit)
13        angka = angka / 10
14    }
15 }

```

```

PS D:\go\modul12> go run .\tugas2mod12\tugas2mod12.go
Masukkan bilangan: 2
2

PS D:\go\modul12> go run .\tugas2mod12\tugas2mod12.go
Masukkan bilangan: 2544
4
5
2
4

```

### Deskripsi program

Program ini digunakan untuk memisahkan dan menampilkan setiap digit dari sebuah bilangan bulat positif secara satu per satu. Digit angka ditampilkan mulai dari digit paling belakang (paling kanan) menuju digit paling depan, sehingga urutannya menjadi terbalik dari bentuk awal bilangan. Program menggunakan operasi modulus untuk mengambil digit terakhir dari bilangan, kemudian membuang digit tersebut melalui pembagian bilangan bulat sehingga loop dapat berjalan ke digit berikutnya.

Proses tersebut diulang hingga seluruh digit bilangan berhasil ditampilkan. Program ini bertujuan untuk menunjukkan cara memproses angka per digit dengan menggunakan operasi aritmatika dasar dan struktur perulangan.

### 3. Tugas 3

#### Source code

```
package main

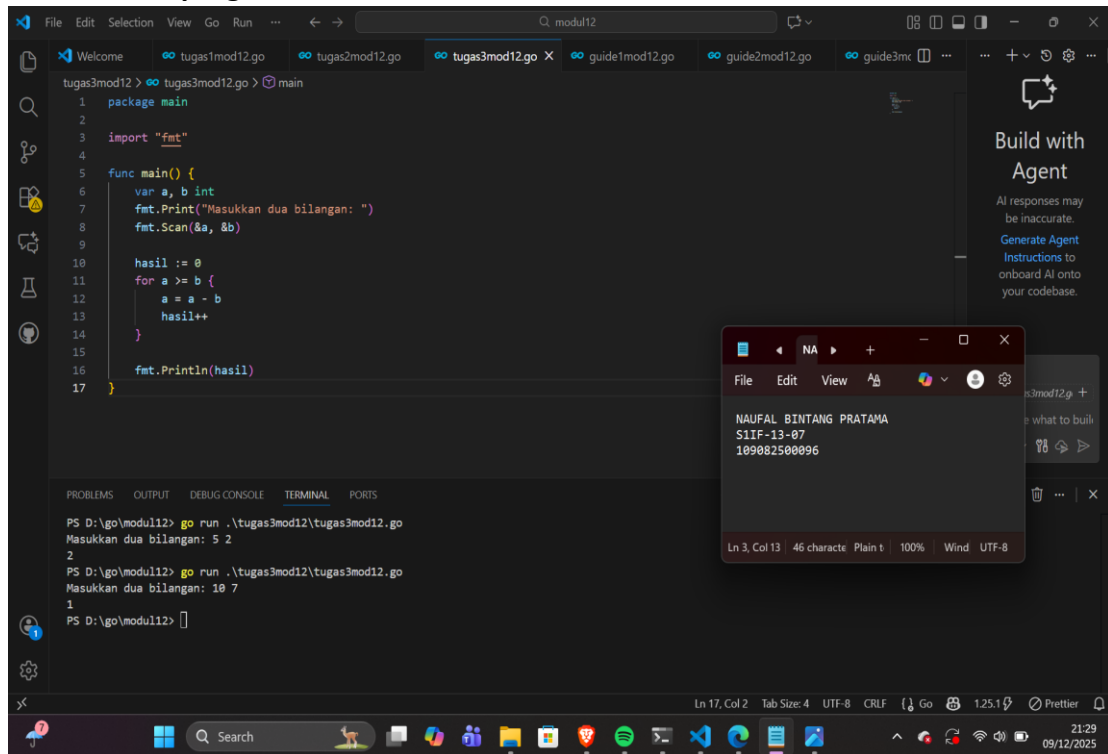
import "fmt"

func main() {
    var a, b int
    fmt.Print("Masukkan dua bilangan: ")
    fmt.Scan(&a, &b)

    hasil := 0
    for a >= b { // kurang sampai tidak cukup lagi
        a = a - b
        hasil++
    }

    fmt.Println(hasil)
}
```

#### Screenshoot program



```
tugas3mod12 > go run .\tugas3mod12\tugas3mod12.go
1 package main
2
3 import "fmt"
4
5 func main() {
6     var a, b int
7     fmt.Print("Masukkan dua bilangan: ")
8     fmt.Scan(&a, &b)
9
10    hasil := 0
11    for a >= b {
12        a = a - b
13        hasil++
14    }
15
16    fmt.Println(hasil)
17 }
```

```
PS D:\go\modul12> go run .\tugas3mod12\tugas3mod12.go
Masukkan dua bilangan: 5 2
2
PS D:\go\modul12> go run .\tugas3mod12\tugas3mod12.go
Masukkan dua bilangan: 10 7
1
PS D:\go\modul12>
```

#### Deskripsi program

Program ini digunakan untuk menghitung hasil pembagian integer dari dua bilangan tanpa menggunakan operator pembagian (/).

Konsep perhitungan yang digunakan adalah pengurangan berulang, yaitu mengurangi bilangan pertama dengan bilangan kedua secara terus-menerus sampai bilangan pertama tidak lagi cukup untuk dikurangi. Setiap kali pengurangan dilakukan, nilai penghitung (counter) ditambah satu. Nilai penghitung inilah yang menjadi hasil pembagian integer. Metode ini menggambarkan cara kerja pembagian secara logis dan sederhana, serta membantu memahami bagaimana suatu pembagian dapat dilakukan tanpa fasilitas operator bawaan. Program ini juga melatih logika perulangan dan pemahaman terhadap proses komputasi dasar.