

**LAPORAN PRAKTIKUM ALGORITMA**

**DAN PEMROGRAMAN 1**

**MODUL 12**

**WHILE-LOOP**



**Disusun oleh:**

**Andromeda Alika Ramadhani**

**109082500156**

**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, i int
    fmt.Scan(&n)

    i = n

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

#### Screenshot program

The screenshot shows a code editor interface with a dark theme. On the left, the code file `guided1.go` is displayed:

```

GUIDED1 > -o guided1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n, i int
7     fmt.Scan(&n)
8
9     i = n
10
11    for i > 1 {
12        fmt.Print(i, " x ")
13        i = i - 1
14    }
15    fmt.Println(1)
16 }
17

```

To the right of the code, there is a white box containing student information:

**109082500156**  
**IF-13-02**  
**Andromeda Alika Ramadhani**

Below the code editor is a terminal window showing the execution of the program:

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\GUIDED1\guided1.go"
0
1
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\GUIDED1\guided1.go"
5
5 x 4 x 3 x 2 x 1
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\GUIDED1\guided1.go"
10
10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\GUIDED1\guided1.go"
1
1
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12>

```

The terminal also displays status information at the bottom:

Ln 14, Col 6 Tab Size: 4 UTF-8 CRLF { } Go ⚙ 1.25.1 ⚙ (↻) Go Live 🔍

## Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk deret bilangan faktorial dari suatu bilangan:

1. Deklarasikan variabel yang akan kita gunakan. Variabel yang kita gunakan adalah variabel `n` dan `i` (untuk perulangannya) dengan tipe data integer. Tipe data yang kita gunakan yaitu tipe data integer karena yang diminta adalah bilangan bulat.
2. Gunakan “`fmt.Scan(&n)`” untuk menginputkan bilangan yang akan kita hitung.
3. Inisiasi variabel `i = n`. Hal ini berarti program akan memulai perulangan dari angka yang diinputkan, lalu menghitung mundur hingga 1.
4. Selanjutnya kita gunakan gunakan perulangan `for` dengan kondisi dimana `i > 1`, sehingga perulangan akan terus berjalan selama nilai `i` lebih besar dari 1, kemudian mengurangi nilai `i` sebesar 1 pada setiap iterasi. Setelah perulangan selesai, program akan mencetak deret bilangan dari faktorial `n`.
5. Gunakan “`fmt.Print`” untuk menampilkan hasil deret bilangan dari faktorial `n` yang berupa perkalian berurutan dari bilangan yang diinputkan pengguna.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {
    var token string

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

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

## Screenshot program

```
GUIDED1 > GUIDED2 > ➜ guided2.go > ⌂ main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var token string
7
8     for token != "12345abcde" {
9         fmt.Print("Masukkan Password: ")
10        fmt.Scan(&token)
11    }
12    fmt.Println("Selamat Anda berhasil login")
13
14 }
```

109082500156  
IF-13-02  
Andromeda Alika Ramadhani

```
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 1 2\GUIDED1\GUIDED2\guided2.go"
Masukkan Password: Qwe12312
Masukkan Password: 231234
Masukkan Password: 13213
Masukkan Password: 1231ijwe
Masukkan Password: 12345abcde
Selamat Anda berhasil login
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 1 2\GUIDED1\GUIDED2\guided2.go"
Masukkan Password: 12345abcde
Selamat Anda berhasil login
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12>
```

### Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk melakukan login ke dalam suatu aplikasi:

1. Deklarasikan variabel yang akan kita gunakan. Disini kita menggunakan variabel token dengan tipe data string untuk menampung token yang diinputkan pengguna. Tipe data string adalah tipe data yang digunakan untuk menyimpan karakter kata, angka, kalimat dengan menggunakan tanda "".
2. Selanjutnya kita gunakan perulangan for dengan kondisi "token != "12345abcde" yang artinya perulangan akan terus berjalan selama token yang diinputkan tidak sama dengan token yang benar. Selanjutnya kita gunakan "fmt.Scan(&token)" untuk menginputkan token. Jika input masih salah, maka perulangan akan terus berjalan sampai token yang diinputkan bernilai benar. Setelah kondisi terpenuhi, maka program akan keluar dari loop dan menampilkan pesan "Selamat anda berhasil login"
3. Terakhir, gunakan "fmt.Println" untuk mencetak pesan ketika perulangan telah berhenti.

### 3. Guided 3

#### Source Code

```
package main

import "fmt"

func main() {
    var n, i, a, b, hasil int
    fmt.Scan(&n)
```

```

    a = 0

    b = 1

    i = 0

for i < n {

    fmt.Println(a, " ")

    hasil = a + b

    a = b

    b = hasil

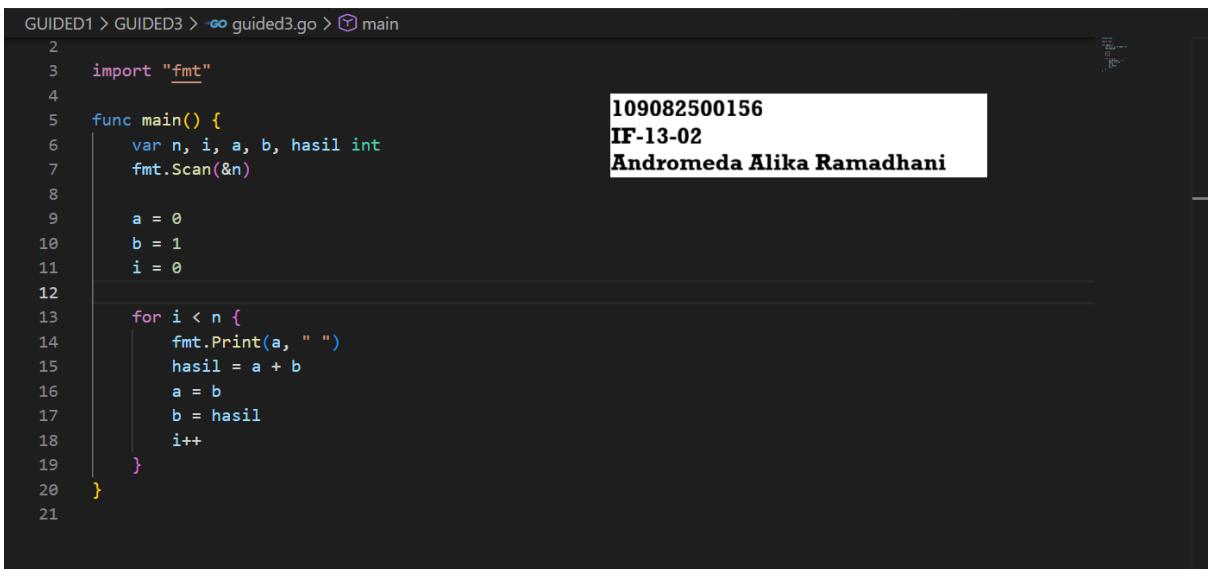
    i++

}

}

```

## Screenshot program



The screenshot shows a terminal window with the following content:

```

GUIDED1 > GUIDED3 > ➜ guided3.go > main
2
3 import "fmt"
4
5 func main() {
6     var n, i, a, b, hasil int
7     fmt.Scan(&n)

8
9     a = 0
10    b = 1
11    i = 0
12
13    for i < n {
14        fmt.Println(a, " ")
15        hasil = a + b
16        a = b
17        b = hasil
18        i++
19    }
20
21

```

The output of the program is displayed in a white box:

```

109082500156
IF-13-02
Andromeda Alika Ramadhani

```

The screenshot shows a terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\GUIDED1\GUIDED3\guided3.go"
5
0 1 2 3
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\GUIDED1\GUIDED3\guided3.go"
2
0 1
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\GUIDED1\GUIDED3\guided3.go"
10
0 1 1 2 3 5 8 13 21 34
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12>
```

At the bottom of the terminal window, status indicators show: Ln 12, Col 1, Tab Size: 4, UTF-8, CRLF, {1 Go, 88, 1.25.1, (→) Go Live, and a bell icon.

## Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk mencetak N bilangan pertama dalam deret Fibonacci:

1. Deklarasikan variabel yang akan kita gunakan. Disini kita menggunakan variabel n sebagai jumlah suku yang ingin ditampilkan, i sebagai iterasi, a, b, dan hasil sebagai proses perhitungan bilangan fibonanci. Tipe data yang digunakan adalah tipe data integer.
2. Gunakan “fmt.Scan(&n)” untuk menginputkan bilangan yang akan kita hitung.
3. Selanjutnya inisiasikan a=0, b= 0 sebagai 2 nilai awal deret fibonacci, sedangkan i=0 karena perhitungan dimulai dari 0.
4. Selanjutnya kita gunakan for dengan kondisi i < n. Pada setiap iterasi, program akan mencetak nilai a sebagai suku fibonacci berikutnya, lalu menghitung nilai berikutnya dengan menjumlahkan a dan b. Setelah itu, nilai a diubah menjadi nilai b dan nilai b diubah menjadi hasil, sehingga pola fibonacci terus berlanjut. Variabel i kemudian ditambah satu untuk mendekati batas perulangan sehingga program akan menghasilkan deret fibonacci sebanyak n suku.
5. Terakhir, gunakan “fmt.Print” untuk mencetak deret bilangan fibonacci.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {
```

```

var username, pass string

var gagal int

for {

    fmt.Scan(&username, &pass)

    if username == "Admin" && pass == "Admin" {

        break

    }

    gagal++

}

fmt.Println(gagal, "percobaan gagal login")

}

```

### Screenshoot program

```

SOAL1 > go run soal1.go
1  package main
2
3  import "fmt"
4
5  func main() {
6      var username, pass string
7      var gagal int
8
9      for {
10          fmt.Scan(&username, &pass)
11
12          if username == "Admin" && pass == "Admin" {
13              break
14          }
15
16          gagal++
17      }
18
19      fmt.Println(gagal, "percobaan gagal login")
20
21

```

The screenshot shows a terminal window with the following content:

```
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\SOAL1\soal1.go"
User123 user123
User admin
Admin admin
Admin Admin123
Admin Admin
4 percobaan gagal login
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12\SOAL1\soal1.go"
Admin Admin
0 percobaan gagal login
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12>
```

At the bottom of the terminal window, there is a status bar with the following information: Ln 21, Col 1 Tab Size: 4 UTF-8 CRLF { Go ⌂ 1.25.1 ⌂ (→) Go Live ⌂

### Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk menghitung seberapa banyak pengguna gagal melakukan login:

1. Deklarasikan variabel yang akan kita gunakan. Disini kita menggunakan variabel username dan pass dengan tipe data string untuk menampung data login dari pengguna dan variabel gagal dengan tipe data integer untuk menghitung jumlah kesalahan.
2. Gunakan perulangan for. Didalam perulangan for, kita gunakan fmt.Scan(&username, &pass) untuk menginputkan username dan password dari pengguna. Setelah itu akan dilakukan pengecekan apakah username dan password yang diinputkan benar atau tidak. Jika password yang diinputkan benar, maka program akan keluar dari perulangan. Jika tidak sama, maka variabel gagal akan ditambah satu untuk mencatat satu kali percobaan login yang gagal sampai input username dan password benar, perulangan akan berhenti dan akan menampilkan jumlah total percobaan yang gagal.
3. Terakhir, gunakan “fmt.Println” untuk menampilkan jumlah total percobaan yang gagal.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {
    var bilangan, digit int
```

```
fmt.Scan(&bilangan)

for bilangan > 0 {
    digit = bilangan % 10

    fmt.Println(digit)

    bilangan = bilangan / 10
}
```

## Screenshot program

```
SOAL1 > SOAL2 > soal2.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan, digit int
7     fmt.Scan(&bilangan)
8
9     for bilangan > 0 {
10         digit = bilangan % 10
11         fmt.Println(digit)
12         bilangan = bilangan / 10
13     }
14 }
15
```

109082500156  
IF-13-02  
**Andromeda Alika Ramadhani**

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 1 2\SOAL1\SOAL2\soal2.go"

2

2

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 1 2\SOAL1\SOAL2\soal2.go"

2544

4

4

5

2

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 1 2\SOAL1\SOAL2\soal2.go"

3423554654

4

5

6

4

5

5

3

2

4

3

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\WHILE-LOOP MODUL 12>

### **Deskripsi program**

Pada program diatas, kita diminta untuk membuat program untuk mencacah setiap digit yang terdapat di dalam suatu bilangan bulat positif:

1. Deklarasikan variabel yang akan kita gunakan. Disini kita menggunakan variabel bilangan untuk input angka dan digit untuk menampung hasil pencacahan tiap digit dengan tipe data integer karena yang diminta adalah bilangan bulat.
2. Gunakan “fmt.Scan(&bilangan)” untuk menginputkan bilangan.
3. Selanjutnya gunakan perulangan for dengan kondisi bilangan > 0. Di dalam perulangan, program akan menghitung digit terakhir dari bilangan dengan menggunakan modulus (bilangan%10). Hasil dari modulus tersebut kemudian ditampilkan dengan menggunakan perintah “fmt.Println”. Setelah itu, bilangan tersebut diperkecil lagi dengan cara membagi dirinya sendiri menggunakan pembagian bilangan bulat (bilangan/10), sehingga digit terakhir yang sudah ditampilkan akan terhapus. Proses ini terus berulang hingga bilangan menjadi 0. Setelah itu, program akan menampilkan semua digit bilangan yang diinputkan secara terbalik, mulai dari satuan, puluhan, ratusan, ribuan, dst.

### **3. Tugas 3**

#### **Source code**

```
package main

import "fmt"

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

    hasil = 0

    for x >= y {
        x = x - y
        hasil++
    }
    fmt.Println(hasil)
}
```

#### **Screenshoot program**

## Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk mencari hasil integer division dari 2 bilangan:

1. Deklarasikan variabel yang akan kita gunakan. Variabel yang kita gunakan adalah variabel x dan y untuk menampung 2 input bilangan dan variabel hasil untuk menampung hasil pembagian. Tipe data yang digunakan adalah tipe data integer.
  2. Gunakan “fmt.Scan(&x, &y)” untuk menginputkan 2 buah bilangan yang akan dihitung.
  3. Selanjutnya inisiasikan variabel hasil = 0 karena perhitungan dimulai dari 0.
  4. Selanjutnya gunakan perulangan for dengan kondisi  $x \geq y$ , artinya selama nilai x masih lebih dari sama dengan y, maka program akan terus melakukan pengurangan  $x - y$ . Setiap kali pengurangan dilakukan, variabel hasil akan ditambah satu. Setelah perulangan berakhir, program akan mencetak nilai hasil pembagian x/y.
  5. Selanjutnya gunakan “fmt.Println” untuk menampilkan hasil pembagian x dan y yang telah dihitung menggunakan perulangan.