

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 12
WHILE-LOOP



Disusun oleh:

Ichsan Maulana Muhammad

109082500093

S1IF-13-04

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

The screenshot shows a Go IDE with the following components:

- EXPLORER:** A file tree on the left showing a project structure with folders like 'MODUL 10', 'MODUL 11', and 'MODUL 12'. The file 'guided 11.go' is selected under 'MODUL 12'.
- EDITOR:** The main window displays the source code of 'guided 11.go':

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n, j int
7     fmt.Scan(&n)
8     j = n
9     for j > 1 {
10        fmt.Print(j, " x ")
11        j = j - 1
12    }
13    fmt.Println(1)
14 }
15
```
- TERMINAL:** A terminal window at the bottom shows the execution of the program. It displays the command 'go run "c:\Users\VASUS\yy\MODUL 12\guided 11.go"' and the resulting output: '5 x 4 x 3 x 2 x 1'.
- logUp:** A small window titled 'logUp' is overlaid on the terminal, showing the user's name 'NAMA: ICHSAN MAULANA M', NIM 'NIM: 109082500093', and class 'KELAS: IF-13-04'.

Deskripsi program

1. Program diawali dengan deklarasi package main, import "fmt", serta fungsi main() sebagai fungsi utama yang akan dieksekusi saat program dijalankan.
2. Mendeklarasikan variabel n dan j dengan tipe data integer, di mana variabel n digunakan untuk menampung input bilangan dari pengguna, dan j digunakan sebagai penghitung dalam proses menampilkan deret faktorial.
3. Program meminta pengguna memasukkan sebuah bilangan bulat non-negatif, kemudian membaca input tersebut menggunakan fungsi fmt.Scan(&n).
4. Menginisialisasi variabel j dengan nilai n, karena deret faktorial akan dimulai dari bilangan yang dimasukkan pengguna.
5. Program melakukan perulangan menggunakan for loop dengan kondisi j > 1 untuk menampilkan deret faktorial secara menurun.
 - Pada setiap iterasi, program mencetak nilai j diikuti dengan tanda " x ".
 - Setelah mencetak, nilai j akan dikurangi satu dengan operasi j = j - 1.
6. Setelah perulangan selesai (saat j bernilai 1), program mencetak angka 1 menggunakan fmt.Println(1) sebagai akhir dari deret faktorial.
7. Output yang dihasilkan berupa deret bilangan faktorial dari n sampai 1, ditampilkan dalam format misalnya:
 - Jika input 5 → 5 x 4 x 3 x 2 x 1
 - Jika input 10 → 10 x 9 x 8 x ... x 1
 - Jika input 0 → 1

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var token string

    fmt.Scan(&token)

    for token != "12345abcde" {

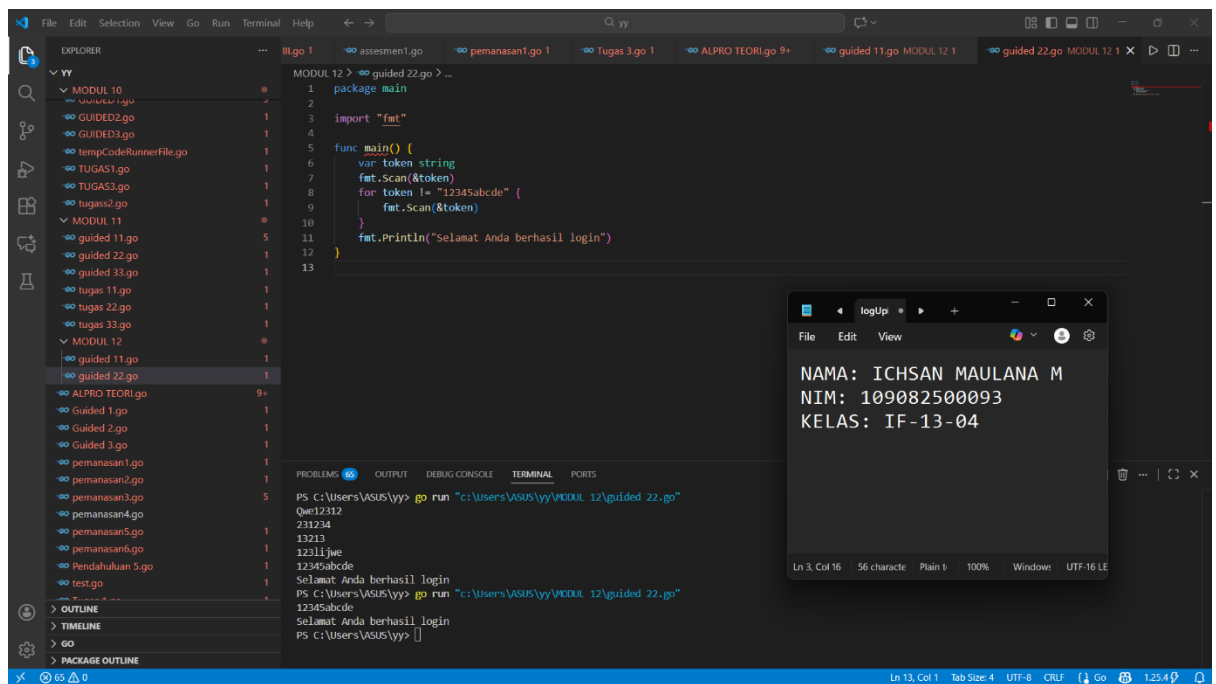
        fmt.Scan(&token)

    }

    fmt.Println("Selamat Anda berhasil login")

}
```

Screenshoot program



Deskripsi program

1. Program dimulai dengan deklarasi package main, import "fmt", dan fungsi main() sebagai fungsi utama yang akan dieksekusi oleh program.
2. Program mendeklarasikan variabel token dengan tipe data string, yang digunakan untuk menampung input token dari pengguna.
3. Program meminta pengguna memasukkan sebuah token, kemudian membaca nilai token pertama menggunakan fmt.Scan(&token).
4. Program melakukan pengecekan token menggunakan perulangan for dengan kondisi token != "12345abcde", selama token yang dimasukkan tidak sama dengan token valid "12345abcde", program akan:
 - Meminta pengguna memasukkan token lagi,
 - Membaca input token baru menggunakan fmt.Scan(&token).
5. Perulangan akan berhenti ketika pengguna memasukkan token yang benar, yaitu "12345abcde".
6. Setelah kondisi terpenuhi, program menampilkan teks "Selamat Anda berhasil login" menggunakan fmt.Println(...).
7. Output akhir berupa pesan bahwa pengguna berhasil login, setelah token yang dimasukkan sesuai dengan token valid yang sudah ditentukan.

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

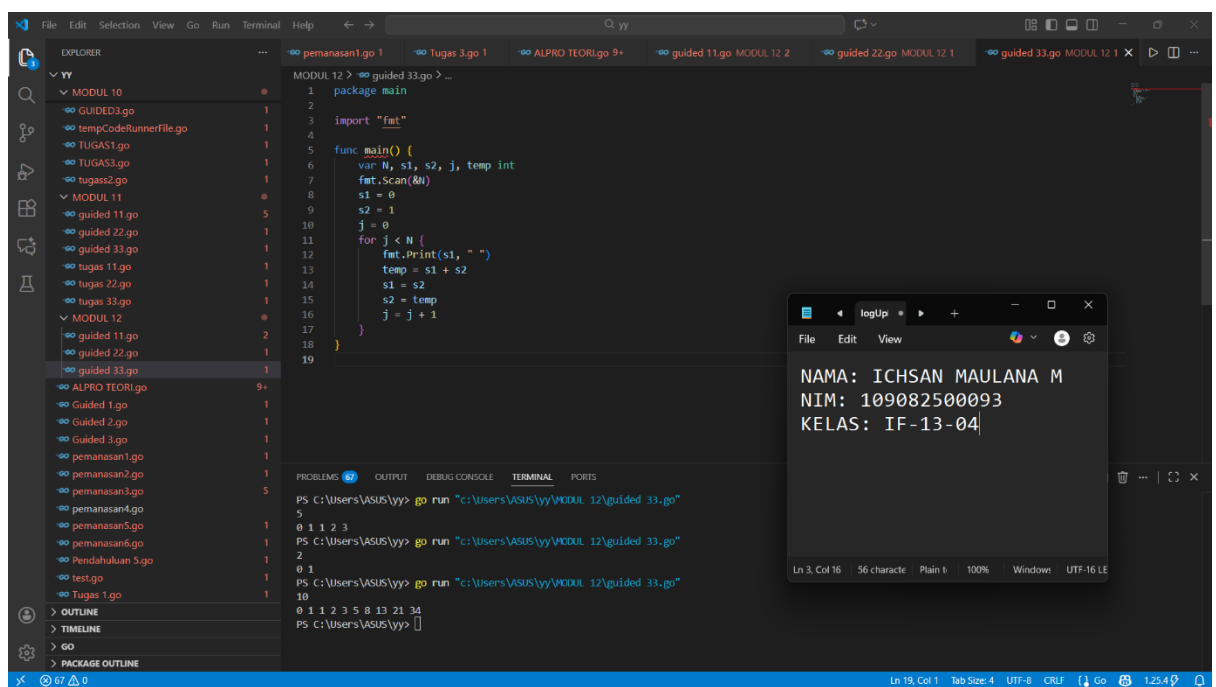
        j = j + 1

    }

}

```

Screenshoot program



Deskripsi program

1. Program diawali dengan deklarasi `package main`, `import "fmt"`, serta fungsi `main()` sebagai fungsi utama yang akan dijalankan ketika program dieksekusi.
2. Program mendeklarasikan beberapa variabel integer, yaitu:
 - `N` → jumlah bilangan Fibonacci yang ingin ditampilkan.
 - `s1` dan `s2` → dua bilangan awal deret Fibonacci.
 - `j` → penghitung iterasi.
 - `temp` → variabel sementara untuk menyimpan hasil penjumlahan Fibonacci.

3. Program meminta pengguna memasukkan sebuah bilangan bulat positif N menggunakan `fmt.Scan(&N)`. Nilai N minimal adalah 2 sesuai dengan ketentuan soal.
4. Program menginisialisasi nilai awal dari deret Fibonacci, yaitu
 - $s1 = 0$
 - $s2 = 1$
 - $j = 0$
5. Program menjalankan perulangan for dengan kondisi $j < N$ untuk mencetak sebanyak N bilangan Fibonacci pertama, pada setiap iterasi
 - Program mencetak nilai $s1$ sebagai salah satu angka dalam deret Fibonacci.
 - Menghitung angka Fibonacci berikutnya dengan rumus $temp = s1 + s2$.
 - Memperbarui nilai Fibonacci dengan: $s1 = s2, s2 = temp$
 - Meningkatkan nilai penghitung $j = j + 1$.
6. Perulangan berhenti setelah jumlah bilangan yang dicetak mencapai N.
7. Output program berupa N bilangan pertama dari deret Fibonacci, dimulai dari 0, 1, 1, 2, 3, 5, dan seterusnya sesuai dengan nilai N yang dimasukkan.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var username, password string

    gagal := 0

    for {

        fmt.Scan(&username, &password)

        if username == "Admin" && password == "Admin" {

            break

        }

        gagal = gagal + 1

    }

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

}
```


Screenshoot program

The screenshot displays a Go IDE with a project named 'yy'. The Explorer panel on the left shows a directory structure with files like 'MODUL 10', 'TUGAS 1.go', 'TUGAS 3.go', etc. The main editor shows the source code for 'MODUL 12 > tugas 11.go'. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var username, password string
7     gagal := 0
8
9     for {
10         fmt.Scan(&username, &password)
11
12         if username == "Admin" && password == "Admin" {
13             break
14         }
15
16         gagal = gagal + 1
17     }
18
19     fmt.Println(gagal, "percobaan gagal login")
20 }
21
```

The bottom panel shows the terminal output of the program:

```
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 12\tugas 11.go"
User123 user123
User admin
Admin admin
Admin Admin123
Admin Admin
4 percobaan gagal login
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 12\tugas 11.go"
Admin Admin
0 percobaan gagal login
PS C:\Users\ASUS\yy>
```

Overlaid on the terminal is a small window titled 'logUp' with the following text:

```
NAMA: ICHSAN MAULANA M
NIM: 10908250093
KELAS: IF-13-04
```

Deskripsi program

1. Program diawali dengan deklarasi package main, import paket "fmt", serta mendefinisikan func main() sebagai fungsi utama.
2. Program mendeklarasikan variabel username dan password bertipe string untuk menampung input pengguna, serta variabel gagal bertipe integer untuk menghitung jumlah percobaan login yang salah.
3. Program menjalankan perulangan while-loop menggunakan sintaks for {} agar program terus meminta input username dan password.
4. Di dalam perulangan, program membaca input username dan password menggunakan fmt.Scan(&username, &password).
5. Program mengecek apakah username dan password yang dimasukkan sama dengan "Admin" dan "Admin".
 - Jika benar, perulangan dihentikan menggunakan break.
 - Jika salah, variabel gagal ditambah 1 untuk mencatat percobaan gagal.
6. Setelah perulangan berhenti, program menampilkan jumlah percobaan salah melalui fmt.Println(...).
7. Output program berupa jumlah berapa kali pengguna gagal login sebelum akhirnya memasukkan username dan password yang benar.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var n int

    fmt.Scan(&n)

    for n > 0 {

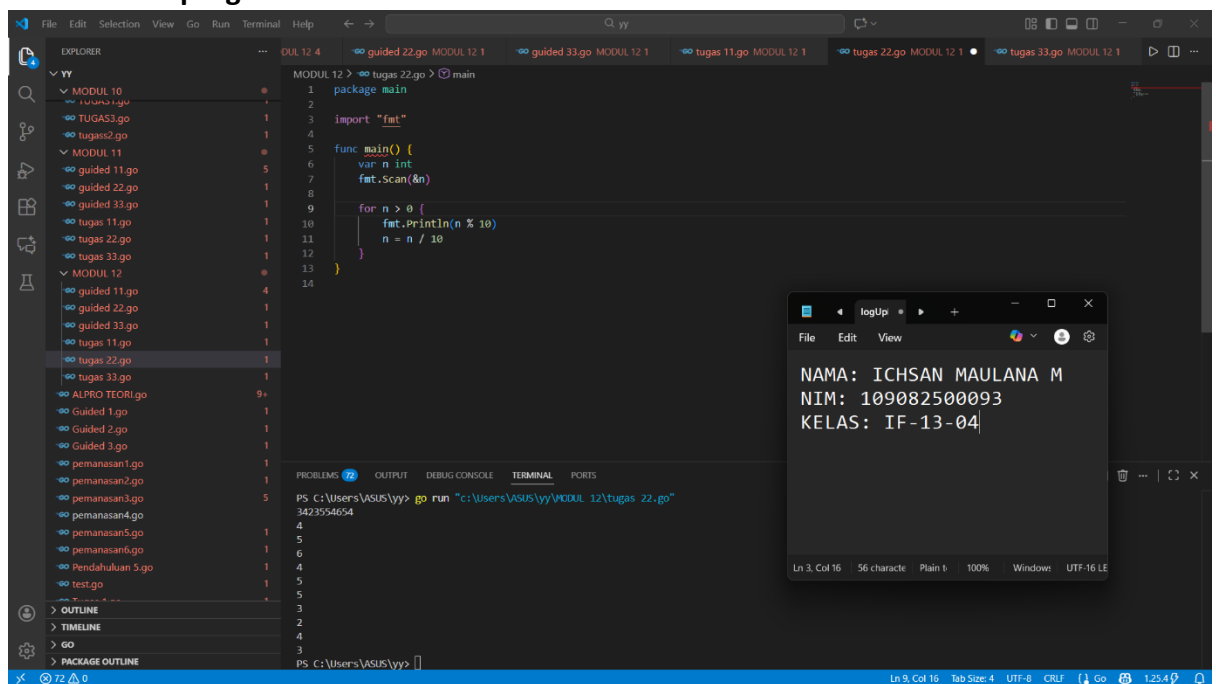
        fmt.Println(n % 10)

        n = n / 10

    }

}
```

Screenshoot program



Deskripsi program

1. Program diawali dengan deklarasi package main, import paket "fmt", serta fungsi main() sebagai fungsi utama.
2. Program mendeklarasikan variabel n bertipe integer yang digunakan untuk menyimpan input bilangan dari pengguna.
3. Program membaca input bilangan bulat positif menggunakan fmt.Scan(&n).
4. Program menggunakan perulangan while-loop dengan kondisi $n > 0$ untuk mencetak setiap digit bilangan dari kanan ke kiri.
5. Pada setiap iterasi:
 - Program menampilkan digit terakhir dari bilangan menggunakan operasi $n \% 10$.
 - Program menghapus digit terakhir dari bilangan menggunakan $n = n / 10$.
6. Proses ini berlangsung sampai nilai n menjadi 0.
7. Output program berupa daftar digit bilangan dari digit paling belakang (paling kanan) hingga digit pertama (paling kiri), sesuai contoh keluaran.

3. 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 = hasil + 1
    }

    fmt.Println(hasil)
}
```

Screenshoot program

```
MODUL 12 > tugas 33.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x, y int
7     fmt.Scan(&x, &y)
8
9     hasil := 0
10
11     for x >= y {
12         x = x - y
13         hasil = hasil + 1
14     }
15
16     fmt.Println(hasil)
17 }
18
```

```
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 12\tugas 33.go"
5 2
2
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 12\tugas 33.go"
10 7
120 4
30
PS C:\Users\ASUS\yy> go run "c:\Users\ASUS\yy\MODUL 12\tugas 33.go"
120 4
30
PS C:\Users\ASUS\yy>
```

```
NAMA: ICHSAN MAULANA M
NIM: 109082500093
KELAS: IF-13-04
```

Deskripsi program

1. Program dimulai dengan deklarasi package main, import "fmt", dan fungsi utama main().
2. Program mendeklarasikan dua variabel x dan y bertipe integer sebagai input, serta variabel hasil untuk menyimpan hasil pembagian.
3. Program membaca dua bilangan bulat positif x dan y menggunakan `fmt.Scan(&x, &y)`.
4. Program menghitung operasi integer division ($x \div y$) menggunakan perulangan while-loop dengan kondisi $x \geq y$.
5. Pada setiap iterasi:
 - Nilai x dikurangi dengan y ($x = x - y$).
 - Variabel hasil ditambah 1 untuk mencatat berapa kali pengurangan dilakukan.
6. Perulangan berhenti ketika nilai x menjadi lebih kecil dari y, yang berarti y sudah tidak bisa dikurangkan lagi.
7. Program menampilkan nilai hasil sebagai hasil akhir integer division.
8. Output berupa nilai $x \div y$ tanpa menggunakan operator pembagian, sesuai contoh tabel hasil.