

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 12

WHILE-LOOP



Disusun oleh:

DYAH IMANSARI

109082500130

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.Scan(&n)

    j = n

    for j > 1 {

        fmt.Print(j, " x ")

        j = j - 1

    }

    fmt.Println(1)

}
```

Screenshot program

```
2-1coba1.go 1 0o tes1.go 1 0o guided12-2.go 1 0o guided12-3.go 1 0o tugas12-1.go 1 0o guided12-1.go 1
EXPLORER .MODUL11&12
File Edit Selection View Go Run ...
1 package main
2 import "fmt"
3 func main() {
4     var n, j int
5     fmt.Scan(&n)
6     j = n
7     for j > 1 {
8         fmt.Print(j, " x ")
9         j = j - 1
10    }
11    fmt.Println(1)
12 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\LENOVO\modul11&12> go run "c:\Users\LENOVO\modul11&12\guided12-1.go"
0
1
5 x 4 x 3 x 2 x 1
PS C:\Users\LENOVO\modul11&12> go run "c:\Users\LENOVO\modul11&12\guided12-1.go"
10
10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1
PS C:\Users\LENOVO\modul11&12> go run "c:\Users\LENOVO\modul11&12\guided12-1.go"
1
1
PS C:\Users\LENOVO\modul11&12>
NAME Nama: Dyah Imansari
Nim: 109082500130
Kelas: S1IF-13-02
```

Deskripsi program

Program ini digunakan untuk menampilkan deret bilangan faktorial dari suatu bilangan. Program dimulai dengan deklarasi dua variabel bertipe integer, yaitu n dan j. Nilai n diperoleh dari input pengguna melalui fungsi fmt.Scan(&n), kemudian disalin ke variabel j sebagai penghitung dalam proses perulangan. Struktur perulangan menggunakan for j > 1, yang berarti selama nilai j lebih besar dari 1, program akan mencetak nilai j diikuti dengan tanda kali "x" tanpa pindah baris. Setelah mencetak, nilai j dikurangi satu setiap iterasi. Ketika kondisi j > 1 tidak lagi terpenuhi, program keluar dari perulangan dan mencetak angka 1 sebagai penutup dengan menggunakan fmt.Println(1). Hasil akhirnya akan jadi seperti 5x4x3x2x1, yang menunjukkan bentuk faktorial.

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")
}
```

Screenshot program

```

package main

import "fmt"

func main() {
    var token string
    fmt.Scan(&token)
    for token != "12345abcde" {
        fmt.Scan(&token)
    }
    fmt.Println("Selamat Anda berhasil login")
}

```

Deskripsi program

Program Go ini digunakan untuk login ke dalam suatu aplikasi dengan asumsi token untuk yang valid adalah "12345abcde". Selama token yang diberikan salah, maka program akan meminta token secara terus menerus hingga token yang diberikan benar. Program dimulai dengan deklarasi satu variabel bertipe string, yaitu token. Nilai token diperoleh dari input pengguna melalui fungsi `fmt.Scan(&token)`, kemudian dilanjutkan dengan struktur perulangan. Struktur perulangan menggunakan `for token != "12345abcde"` lalu diikuti dengan `fmt.Scan(&token)` yang berarti jika token yang dimasukkan bukan "12345abcde", maka akan diminta memasukkan token lagi. Ketika token yang dimasukkan sudah benar, maka program akan mencetak "Selamat Anda berhasil login".

3. Guided 3

Source Code

```

package main

import "fmt"

func main() {
    var N, a, b, j, temp int
    fmt.Scan(&N)
    if N < 2 {
        fmt.Println("Masukkan minimal 2")
    }
}

```

```

        return

    }

a = 0

b = 1

j = 0

for j < N {

    fmt.Println(a, " ")

    temp = a + b

    a = b

    b = temp

    j++

}

}

```

Screenshot program

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows a folder structure for ".MODUL11&12" containing various Go files like guided11-1.go, guided11-2.go, etc.
- Code Editor:** The active file is `guided12-3.go`. The code is as follows:

```

package main
import "fmt"
func main() {
    var N, a, b, j, temp int
    fmt.Scan(&N)
    if N < 2 {
        fmt.Println("Masukkan minimal 2")
        return
    }
    a = 0
    b = 1
    j = 0
    for j < N {
        fmt.Println(a, " ")
        temp = a + b
        a = b
        b = temp
        j++
    }
}

```

- Terminal:** Shows the command `go run "c:/Users/LENOVO/.modul11&12/guided12-3.go"` being run twice, once with input 5 and once with input 20, displaying the Fibonacci sequence.
- Output Panel:** Displays the generated Fibonacci sequence for both inputs.

Deskripsi program

Program Go ini digunakan untuk mencetak N bilangan pertama dalam deret Fibonacci. Program dimulai dengan deklarasi lima variabel bertipe integer: N, a, b, j, dan temp (penyimpan sementara untuk perhitungan). Nilai N diperoleh dari input pengguna melalui fungsi `fmt.Scan(&N)`. Kemudian, sebelum dilanjutkan dengan struktur perulangan, program memastikan bahwa angka yang masuk lebih dari atau sama dengan 2 dengan melakukan pengecekan menggunakan `if N<2` atau jika angka tersebut <2 maka akan mencetak "Masukkan minimal 2" dan langsung berhenti menggunakan perintah `return`. Jika nilai N valid, maka program menginisialisasi dua bilangan pertama dalam deret Fibonacci, yaitu `a = 0` dan `b = 1`. Variabel `j` digunakan sebagai penghitung iterasi dan diinisialisasi dengan nilai 0. Program kemudian menjalankan perulangan selama `j < N`. Di dalam perulangan, nilai `a` sebagai bilangan Fibonacci pertama dicetak diikuti spasi menggunakan `fmt.Print(a, " ")`. Selanjutnya, nilai Fibonacci berikutnya dihitung dengan menyimpan hasil penjumlahan `a + b` ke dalam variabel `temp`. Nilai `a` kemudian diisi dengan `b`, dan `b` diisi dengan `temp` (nilai Fibonacci baru). Variabel `j` ditambah satu setiap iterasi hingga mencapai nilai `N`.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var username, password string

    var total int

    total = 0

    for {

        fmt.Println("Masukkan Username dan Password: ")

        fmt.Scan(&username)

        fmt.Scanln(&password)

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

            break

        }

        total++
    }
}
```

```

        }

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

    }
}

```

Screenshot program

The screenshot shows a code editor with multiple tabs open, including files like `led12-3.go`, `tugas12-1.go`, `guided12-1.go`, etc. The `tugas12-1.go` tab contains the following Go code:

```

package main
import "fmt"
func main() {
    var username, password string
    var total int
    total = 0
    for {
        fmt.Println("Masukkan Username dan Password: ")
        fmt.Scan(&username)
        fmt.Scanln(&password)
        if username == "Admin" && password == "Admin" {
            break
        }
        total++
    }
    fmt.Printf("%d percobaan gagal login\n", total)
}

```

The terminal below shows the execution of the program and its output:

```

PS C:\Users\LENOVO\.modul11&12> go run "c:\Users\LENOVO\.modul11&12\tugas12-1.go"
Masukkan Username dan Password:
User123 user123
Masukkan Username dan Password:
User admin
Masukkan Username dan Password:
Admin admin
Masukkan Username dan Password:
Admin Admin123
Masukkan Username dan Password:
Admin Admin
Admin Admin
4 percobaan gagal login
PS C:\Users\LENOVO\.modul11&12> go run "c:\Users\LENOVO\.modul11&12\tugas12-1.go"
Masukkan Username dan Password:
Admin Admin
0 percobaan gagal login

```

A separate window shows the user's details:

Nama:	Dyah Imansari
Nim:	109082500130
Kelas:	S1IF-13-02

Deskripsi program

Program Go ini meminta pengguna untuk memasukkan username dan password secara berulang hingga keduanya sesuai dengan nilai yang ditentukan, yaitu "Admin". Setiap kali input salah, program mencatat satu percobaan gagal dengan menaikkan nilai variabel total. Proses ini berlangsung dalam loop tak terbatas (`for {}`) yang hanya akan berhenti jika kondisi `username == "Admin" && password == "Admin"` terpenuhi. Setelah login berhasil, program mencetak jumlah total percobaan gagal menggunakan `fmt.Printf`, sehingga pengguna tahu berapa kali mereka salah memasukkan data sebelum berhasil login.

2. Tugas 2

Source code

```

package main

import "fmt"

func main() {
}

```

```
var n, digit int

fmt.Println("Masukkan bilangan bulat positif: ")

fmt.Scan(&n)

if n <= 0 {

    fmt.Println("Input harus bilangan bulat
positif")

    return

}

for n > 0 {

    digit = n % 10

    fmt.Println(digit)

    n = n/10

}
```

Screenshot program

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- File Explorer (Left):** Shows a tree view of files in the ".MODUL11&12" folder. Files listed include: guided11-1.go, guided11-2.go, guided11-3.go, guided12-1.go, guided12-1coba.go, guided12-1coba1.go, guided12-2.go, guided12-3.go, tempCodeRunnerFile.go, tes.go, tes1.go, tugas11-1.go, tugas11-2.go, tugas11-3.go, tugas12-1.go, tugas12-2.go, and tugas12-3.go.
- Editor (Center):** Displays the content of the "tugas12-2.go" file. The code is as follows:

```
1 package main
2 import "fmt"
3 func main() {
4     var n, digit int
5     fmt.Println("Masukkan bilangan bulat positif: ")
6     fmt.Scan(&n)
7     if n <= 0 {
8         fmt.Println("Input harus bilangan bulat positif")
9         return
10    }
11    for n > 0 {
12        digit = n % 10
13        fmt.Println(digit)
14        n = n/10
15    }
16 }
```

- Terminal (Bottom):** Shows the output of running the "tugas12-2.go" file. It prompts for input, receives "2544", and prints each digit on a new line.

```

package main
import "fmt"
func main() {
    var n, digit int
    fmt.Print("Masukkan bilangan bulat positif: ")
    fmt.Scan(&n)
    if n <= 0 {
        fmt.Println("Input harus bilangan bulat positif")
        return
    }
    for n > 0 {
        digit = n % 10
        fmt.Println(digit)
        n = n/10
    }
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\LENOVO\modul11&12> go run "c:\Users\LENOVO\modul11&12\tugas12-2.go"
Masukkan bilangan bulat positif: 3423554654
4
5
6
4
5
5
3
2
4
3
PS C:\Users\LENOVO\modul11&12>

Deskripsi program

Program Go ini meminta pengguna untuk memasukkan bilangan bulat positif, lalu mencetak setiap digit dari bilangan tersebut secara terbalik (digit terakhir (paling kanan) sampai dengan digit pertama (paling kiri)). Setelah input diterima melalui `fmt.Scan(&n)`, program memeriksa apakah nilai `n` lebih kecil atau sama dengan nol. Jika ya, maka akan menampilkan pesan kesalahan dan keluar dari program dengan `return`. Jika valid, program masuk ke dalam loop `for n > 0`, di mana setiap iterasi mengambil digit terakhir dari `n` menggunakan operasi `n % 10`, mencetaknya, lalu menghapus digit tersebut dengan `n = n / 10`. Proses ini berlanjut hingga semua digit tercetak dalam urutan terbalik.

3. Tugas 3

Source code

```

package main

import "fmt"

func main() {

    var x, y, i int
    fmt.Print("Masukkan dua bilangan bulat positif (x y): ")
    fmt.Scan(&x, &y)
    if x < y || y <= 0 {
        fmt.Println("Input tidak valid. Pastikan x >= y dan y > 0.")
        return
    }
}

```

```

        }

        i = 0

        for x >= y {

            x = x-y

            i = i+1

        }

        fmt.Println("hasil integer division:", i)

    }

```

Screenshot program

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows a directory structure under ".MODUL11&12" containing various Go files (e.g., guided11-1.go, guided11-2.go, guided11-3.go, guided12-1.go, guided12-1coba.go, guided12-1coba1.go, guided12-2.go, guided12-3.go, tempCodeRunnerFile.go, tes.go, tes1.go, tugas11-1.go, tugas11-2.go, tugas11-3.go, tugas12-1.go, tugas12-2.go, tugas12-3.go).
- Code Editor:** The current file is "tugas12-3.go". The code is as follows:

```

package main
import "fmt"
func main() {
    var x, y, i int
    fmt.Print("Masukkan dua bilangan bulat positif (x y): ")
    fmt.Scan(&x, &y)
    if x < y || y <= 0 {
        fmt.Println("Input tidak valid. Pastikan x >= y dan y > 0.")
        return
    }
    i = 0
    for x >= y {
        x = x-y
        i = i+1
    }
    fmt.Println("hasil integer division:", i)
}

```
- Terminal:** Shows the command "go run" being run in the terminal, followed by user inputs "5 2" and "10 7", and the resulting outputs "hasil integer division: 2" and "hasil integer division: 10".
- Status Bar:** Shows the user's name, NIM, and class information.

Deskripsi program

Program Go ini melakukan integer division (pembagian bilangan bulat) secara manual menggunakan pengurangan berulang. Pengguna diminta memasukkan dua bilangan bulat positif, yaitu x dan y , lalu program memeriksa input, dengan x harus lebih besar atau sama dengan y , dan y harus lebih besar dari nol. Jika tidak valid, program akan menampilkan pesan kesalahan dan berhenti. Jika valid, program akan mengurangi nilai x dengan y secara berulang sambil menghitung berapa kali pengurangan dilakukan melalui variabel i . Hasil akhir dari proses ini adalah nilai i , yang menunjukkan hasil pembagian bulat $x \div y$, dan ditampilkan sebagai "hasil integer division".