

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 12

WHILE-LOOP



Disusun oleh:

ZHAFIF YUSUF AL AMIN

109082500137

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1

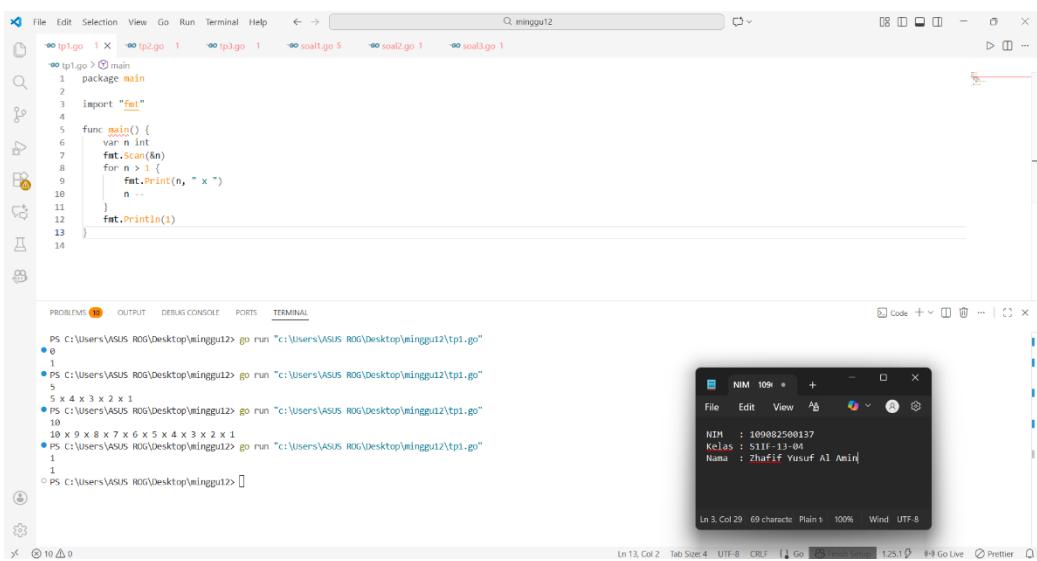
Source Code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Scan(&n)
    for n > 1 {
        fmt.Print(n, " x ")
        n --
    }
    fmt.Println(1)
}
```

Screenshot program



Deskripsi program

package main: menandakan program utama ini ibaratnya kita akan membangun rumah & package main ini sebagai temboknya.

import "fmt": berfungsi memanggil package "fmt" ini ibarat bangun rumah sebagai semen, pasir, batu, air.

func main () {...}: ini bagaikan akses dengan filosofi pintu rumah, sebuah rumah tidak akan bisa di masuki tanpa pintu, ibarat rumah full tembok tanpa ada pintu masuk sama sekali.

var n int: deklarasi variabel n menggunakan tipe data integer.
fmt.Scan (&n) :fungsi untuk menginput dan menyimpan isi dari variabel jam.

```
for n > 1 {  
    fmt.Println(n, " x ")  
    n --  
}  
fmt.Println(1)
```

:perulangan for yang akan di eksekusi jika n lebih besar dari 1. Isi dari perulangan adalah menampilkan output yang berupa n lalu ditambahkan string x dan nilai n akan dikurangi 1 setiap perulangan, jika n tidak memenuhi maka akan langsung menampilkan output berupa angka 1

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var token string

    fmt.Scan(&token)

    for token != "12345abcde" {

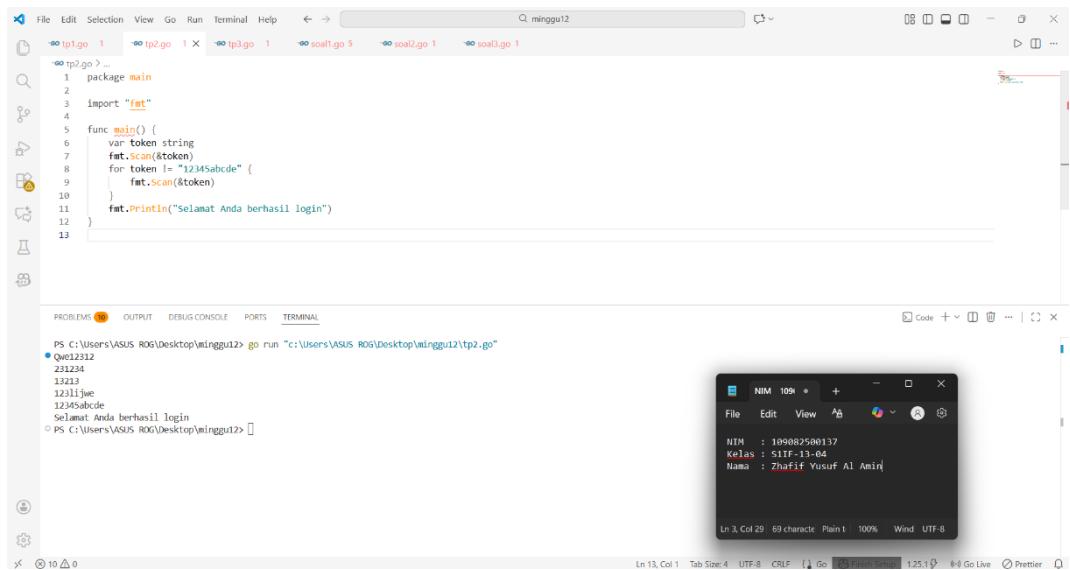
        fmt.Scan(&token)

    }

    fmt.Println("Selamat Anda berhasil login")

}
```

Screenshot program



Deskripsi program

package main : menandakan program utama ini ibaratnya kita akan membangun rumah & package main ini sebagai temboknya.

import "fmt" : berfungsi memanggil package "fmt" ini ibarat dalam membangun rumah sebagai semen, pasir, batu, air.

func main() {...}: ini bagaikan akses dengan filosofi pintu rumah, sebuah rumah tidak akan bisa di masuki tanpa pintu, masa iya rumah full tembok.

var token string : deklarasi variabel token dengan tipe data string.

fmt.Scan(&token) : fungsi untuk menginput dan menyimpan isi dari variabel token.

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

: perulangan yang akan di eksekusi kalo isi dari variabel token itu bukan 12345abcde, perulangan akan di eksekusi sampai isi dari variabel token itu benar. Jika sudah benar maka akan tampil sebuah string ("Selamat Anda berhasil login").

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var N, d1, d2, i, temp int

    fmt.Scan(&N)

    d1 = 0

    d2 = 1

    i = 0

    for i < N {

        fmt.Print(d1, " ")

        temp = d1 + d2

        d1 = d2

        d2 = temp

        i ++
    }
}
```

Screenshot program

The screenshot shows a Go development environment with a code editor and a terminal window.

Code Editor: The left pane displays the source code for `tp3.go`. The code implements a Fibonacci sequence generator using a simple iterative approach with variables `d1`, `d2`, and `i`.

Terminal: The right pane shows the terminal output of the program's execution. It starts with the command `go run "c:/Users/ASUS/ROG/Desktop/minggu12/tp3.go"`. The output lists the first 19 numbers of the Fibonacci sequence: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, and 4181.

Deskripsi program

package main : menandakan program utama ini ibaratnya kita akan membangun rumah & package main ini sebagai temboknya.

import "fmt" : berfungsi memanggil package "fmt" ini ibarat dalam membangun rumah sebagai semen, pasir, batu, air.

func main() {...}: ini bagaikan akses dengan filosofi pintu rumah, sebuah rumah tidak akan bisa di masuki tanpa pintu, masa iya rumah full tembok.

var N, d1, d2, i, temp int : deklarasi variabel dengan tipe data int.

fmt.Scan(&N) : fungsi untuk menginput dan menyimpan isi dari variabel N.

```
d1 = 0  
d2 = 1  
I=0  
: deklarasi isi dari variabel d1, d2 dan i
```

```
for i < N {  
    fmt.Print(d1, " ")  
    temp = d1 + d2  
    d1 = d2  
    d2 = temp  
    i ++  
}
```

: Perulangan yang akan di eksekusi selama nilai i lebih kecil daripada N. isi dari perulangan adalah menampilkan output dari variabel d1 dulu lalu menjumlahkan d1 dengan d2 lalu mengganti d1 menjadi d2 lalu d2 menjadi temp dan menambahkan nilai i sebanyak 1 di setiap perulangan.

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var u, p string
    fmt.Scan(&u, &p)
    ux := "Admin"
    px := "Admin"
    gagal := 0
    for u != ux || p != px {
        gagal++
        fmt.Scan(&u, &p)
    }
    fmt.Println(gagal)
    fmt.Print("percobaan gagal login ")
}
```

Screenshot program

The screenshot shows a VS Code interface with the following details:

- File Explorer:** Shows files tp1.go, tp2.go, tp3.go, soal1.go (highlighted in red), soal2.go, and soal3.go.
- Code Editor:** Displays the content of soal1.go, which contains a Go program for a login system. The code uses pointers and loops to check user input against predefined values.
- Terminal:** Shows the execution of the program. It prompts for a user and password, then prints "Admin" if the input matches. It also includes a section labeled "percobaan gagal login".
- Status Bar:** Shows file paths like C:\Users\ASUS ROG\Desktop\minggu12\soal1.go, line numbers (Ln 9, Col 13), and other standard status bar information.

Deskripsi program

package main : menandakan program utama ini ibaratnya kita akan membangun rumah & package main ini sebagai temboknya.

import "fmt" : berfungsi memanggil package "fmt" ini ibarat dalam membangun rumah sebagai semen, pasir, batu, air.

func main () {...} : ini bagaikan akses dengan filosofi pintu rumah, sebuah rumah tidak akan bisa di masuki tanpa pintu, masa iya rumah full tembok.

var u, p string : Pendeklarasian variabel u dan p menggunakan tipe data string

fmt.Scan (&u, &p) : fungsi untuk menginput dan setelah di input isi dari variabel akan disimpan di &pH.

ux:=Admin

px:=Admin

: deklarasi otomatis variabel ux dan px itu isi variabelnya "Admin".

gagal:=0: deklarasi otomatis variabel gagal dengan isi variabelnya 0.

```
for u != ux || p != px {  
    gagal++  
    fmt.Scan (&u, &p)  
}  
fmt.Println(gagal)  
fmt.Print("percobaan gagal login ")  
}
```

: fungsi perulangan yang akan di eksekusi jika u dan p yang kita input bukan Admin dan Admin sesuai pada variabel ux dan px. gagal akan ditambah 1 di setiap perulangan. Perulangan akan berhenti jika u dan p benar kemudian menampilkan output berapa kali percobaan yang gagal dan sebuah string ("percobaan gagal login ").

2. Tugas 2

Source code

```
package main

import "fmt"

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

    h := 0
    for x > 0 {
        h = x % 10
        fmt.Println(h)

        x /= 10
    }
}
```

Screenshot program

The screenshot shows a Go development environment with the following components:

- Code Editor:** Displays `soal2.go` with the following code:

```
soal2.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x int
7     fmt.Scan(&x)
8     h:=x
9     for x > 0 {
10         h = x % 10
11         fmt.Println(h)
12         x /= 10
13     }
14 }
15
16 }
```

- Terminal:** Shows the command line interface with the following session:

```
PS C:\Users\ASUS ROG\Desktop\minggu12
2> go run "c:/Users/ASUS ROG/Desktop\minggu12/soal2.go"
3423554654
4
5
6
7
8
9
10
11
12
13
14
15
16
```

- Output Window:** Displays the program's output:

```
NIM : 109082500137
Kelas : S1IF-13-04
Nama : Zhefif Yusuf Al Amin
```

Deskripsi program

package main : menandakan program utama ini ibaratnya kita akan membangun rumah & package main ini sebagai temboknya.

import "fmt": berfungsi memanggil package "fmt" ini ibarat dalam membangun rumah sebagai semen, pasir, batu, air.

func main () {...}: ini bagaikan akses dengan filosofi pintu rumah, sebuah rumah tidak akan bisa di masuki tanpa pintu, masa iya rumah full tembok.

var x int : Deklarasi variabel x menggunakan tipe data integer.

fmt.Scan (&x) : digunakan untuk membaca inputan dari variabel x lalu disimpan.

h:=0: deklarasi otomatis variabel h dengan isi 0.

```
for x > 0 {  
    h = x % 10  
    fmt.Println(h)  
    x /= 10  
  
}
```

: perulangan yang akan di eksekusi selama x lebih besar dari 0. Isi dari perulangannya adalah variabel h dengan operasi x mod 10 agar bisa mengambil angka paling belakang dari x lalu di outputkan selama perulangan dan di div 10 agar bisa mendapatkan angka kedua, ketiga dan seterusnya dari x selama perulangan.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Scan(&x, &y)
    i := 0
    for x >= y {
        x = x - y
        i++

    }
    fmt.Println(i)
}
```

Screenshoot program

The screenshot shows a Go development environment with the following components:

- Editor Area:** Displays the source code for `soal3.go`. The code defines a package `main` with a `main()` function that calculates the number of iterations required to reduce `x` to `y` by subtracting `y` from `x` in each iteration.
- Terminal:** Shows the command `go run "c:\Users\ASUS ROG\Desktop\minggu12\soal3.go"` being run three times, resulting in outputs 5, 2, and 1 respectively.
- Modal Window:** A small window titled "NIM 109" displays student information:
 - NIM : 109082500137
 - Kelas : SITF-13-94
 - Nama : Zhafif Yusuf Al Amin
- Status Bar:** Shows file paths like `C:\Users\ASUS ROG\Desktop\minggu12\soal3.go`, line numbers (Ln 3, Col 29), character count (69), tab size (4), encoding (UTF-8), and other developer tools like "Prettier".

Deskripsi program

`package main` : menandakan program utama ini ibaratnya kita akan membangun rumah & `package main` ini sebagai temboknya.

`import "fmt"` : berfungsi memanggil package "fmt" ini ibarat dalam membangun rumah sebagai semen, pasir, batu, air.

`func main () {...}`: ini bagaikan akses dengan filosofi pintu rumah, sebuah rumah tidak akan bisa di masuki tanpa pintu, masa iya rumah full tembok.

```
var x, y int: deklarasi variabel x dan y menggunakan tipe data integer.
```

```
fmt.Scan(&x, &y) : berfungsi untuk membaca inputan yang telah kita input lalu disimpan ke variabel &x dan &y.
```

```
i:=0 : deklarasi otomatis variabel i dengan isi 0.
```

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

```
: fungsi perulangan for yang akan diulang terus selama x lebih besar atau sama dengan y dengan isi perulangannya adalah operasi variabel x di kurangi y sampai x lebih kecil daripada y. isi dari variabel i akan ditambah 1 disetiap perulangan sampai x lebih kecil daripada y. i ditampilkan diluar perulangan.
```