

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 12

WHILE-LOOP



Disusun oleh:

FAREL TRI JULIAN

109082500163

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.Print("masukan biangan : ")
    fmt.Scan(&n)

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

Screenshoot program

The screenshot shows a code editor interface with a dark theme. On the left, a code editor window displays the following Go code:

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

Below the code editor are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing the command `go run "e:\pemrograman\1des.go\guided1.go"` and its output:

```
PS E:\pemrograman> go run "e:\pemrograman\1des.go\guided1.go"
masukan biangan : 10
10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1
PS E:\pemrograman>
```

To the right of the terminal is a terminal window titled "NAM". It contains the text:

```
NAMA : Farel Tri Julian
NIM : 109082500163
```

At the bottom of the terminal window, status information is displayed: Ln 2, Col 19 | 42 karakter | Terfordi | 100% | Windows | UTF-8.

Deskripsi program

Program ini bertujuan untuk menampilkan urutan perkalian bilangan mundur angka yang diinputkan pengguna hingga angka 1, menyerupai representasi visual operasi faktorial. Program dimulai dengan membaca input bilangan bulat n ke dalam variabel j, kemudian menjalankan perulangan for (yang berfungsi sebagai while-loop) selama nilai j lebih besar dari 1; di dalam perulangan tersebut, program mencetak angka saat ini diikuti tanda " x " dan kemudian mengurangi nilai j sebesar satu (decrement). Setelah perulangan selesai, program mencetak angka 1 sebagai penutup baris, sehingga jika pengguna memasukkan angka 5, hasilnya akan tertulis sebagai $5 \times 4 \times 3 \times 2 \times 1$.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var t string

    for {
        fmt.Print("masukan pass: ")
        fmt.Scan(&t)

        if t == "12345abcde" {
            fmt.Println("selamat anda berhasil login")
            break
        } else {
            fmt.Println("coba lagi")
        }
    }
}
```

Screenshoot program

The screenshot shows a Go development environment. On the left, the code editor displays a file named `1des.go` with the following content:

```
1des.go > ̄∞ guided2.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var t string
7
8      for {
9          fmt.Print("masukan pass: ")
10         fmt.Scan(&t)
11         if t == "12345abcde" {
12             fmt.Println("selamat anda berhasil login")
13             break
14         } else {
15             fmt.Println("coba lagi")
16         }
17     }
18 }
```

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

```
PROBLEMS 56 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\pemrograman> go run "e:\pemrograman\1des.go\guided2.go"
masukan pass: 12345abcde
selamat anda berhasil login
PS E:\pemrograman> go run "e:\pemrograman\1des.go\guided2.go"
masukan pass: hahaha
coba lagi
masukan pass: 12345abcde
selamat anda berhasil login
PS E:\pemrograman> 
```

To the right of the terminal is a modal window titled "NAM". It contains the text:

NAMA : Farel Tri Julian
NIM : 109082500163

At the bottom of the interface, there are status bars for "Ln 2, Col 19 42 karakter", "Terford", "100%", "Windows", "UTF-8", and "Ln 17, Col 6 Tab Size: 4 UTF-8 CRLF { Go Signed out 1.25.1".

Deskripsi program

Program tersebut mengimplementasikan sistem autentikasi sederhana menggunakan struktur perulangan tak terbatas untuk memvalidasi kata sandi. Di dalam blok for, program terus-menerus meminta input pengguna ke dalam variabel string `t` dan membandingkannya dengan nilai kunci "12345abcde"; jika input cocok, program mencetak pesan keberhasilan dan menggunakan perintah `break` untuk menghentikan perulangan, namun jika input tidak cocok, program akan mencetak pesan "coba lagi" dan kembali meminta input hingga kata sandi yang benar dimasukkan.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var n int

    fmt.Print("masukan nilai: ")
    fmt.Scan(&n)

    if n < 2 {
        fmt.Println("masukan nilai minimal 2")
        return
    }

    f1 := 0
    f2 := 1

    fmt.Print(f1, " ", f2)
    for i := 3; i <= n; i++ {
        fn := f1 + f2
        fmt.Print(" ", fn)
        f1 = f2
        f2 = fn
    }

    fmt.Println()
}
```

Screenshoot program

The screenshot shows a code editor interface with a dark theme. On the left is the code editor pane displaying a Go program. On the right is the terminal pane showing the execution and output of the program.

```
4
5 func main() {
6     var n int
7
8     fmt.Print("masukan nilai: ")
9     fmt.Scan(&n)
10
11    if n < 2 {
12        fmt.Println("masukan nilai minimal 2")
13        return
14    }
15
16    f1 := 0
17    f2 := 1
18
19    fmt.Print(f1, " ", f2)
20    for i := 3; i <= n; i++ {
21        fn := f1 + f2
22        fmt.Print(" ", fn)
23        f1 = f2
24        f2 = fn
25    }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\pemrograman> go run "e:\pemrograman\ides.go\guided3.go"
masukan nilai: 5
0 1 1 2 3
PS E:\pemrograman> go run "e:\pemrograman\ides.go\guided3.go"
masukan nilai: 1
masukan nilai minimal 2
PS E:\pemrograman>
```

The terminal output shows two runs of the program. The first run takes input '5' and prints the sequence 0 1 1 2 3. The second run takes input '1', which is invalid, and prints 'masukan nilai minimal 2'. The interface includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS, and a status bar at the bottom.

Deskripsi program

Program ini dirancang untuk menghasilkan dan menampilkan deret bilangan Fibonacci sebanyak n suku berdasarkan input pengguna. Setelah memvalidasi bahwa input minimal bernilai 2, program menginisialisasi dan mencetak dua angka pertama dari deret tersebut (0 dan 1), kemudian menggunakan perulangan for mulai dari suku ketiga hingga suku ke- n untuk menghitung angka berikutnya dengan menjumlahkan dua angka sebelumnya ($f_1 + f_2$), lalu mencetak hasilnya secara berurutan sembari memperbarui nilai variabel untuk iterasi selanjutnya.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    correctuser := "Admin"
    correctpass := "Admin"
    var ipass, iuss string
    failed := 0

    for {
        fmt.Print("masukan ussername dan pass : ")
        fmt.Scan(&ipass, &iuss)

        if ipass == correctpass && iuss == correctuser {
            break
        }
        failed++
    }
    fmt.Println("percobaan gagal login", failed)
}
```

Screenshoot program

```
1des.go > 66 laatihan1.go > ...
2
3 import "fmt"
4
5 func main() {
6     correctuser := "Admin"
7     correctpass := "Admin"
8     var ipass, iuss string
9     failed := 0
10
11    for {
12        fmt.Print("masukan ussername dan pass : ")
13        fmt.Scan(&ipass, &iuss)
14
15        if ipass == correctpass && iuss == correctuser {
16            break
17        }
18        failed++
19    }
20    fmt.Println("percobaan gagal login", failed)
21}
22
```

PROBLEMS 56 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\pemrograman> go run "e:\pemrograman\1des.go\laatihan1.go"
masukan ussername dan pass : imin imin
masukan ussername dan pass : amin anis
masukan ussername dan pass : Admin Admin
percobaan gagal login 2
PS E:\pemrograman> go run "e:\pemrograman\1des.go\laatihan1.go"
masukan ussername dan pass : Admin Admin
percobaan gagal login 0
PS E:\pemrograman> [ ]
```

NAMA : Farel Tri Julian
NIM : 109082500163

Ln 2, Col 1 42 karakter ^ Terfon | 100% | Windows | UTF-8

Ln 22, Col 1 Tab Size: 4 UTF-8 CRLF ⌂ Go ⌂ Signed out 1.25.1 ⌂

Deskripsi program

Program ini berfungsi sebagai simulasi login sederhana yang melacak jumlah kegagalan autentikasi pengguna sebelum akhirnya berhasil masuk. Program menginisialisasi kredensial yang valid (username dan password "Admin") dan variabel penghitung kegagalan, kemudian menjalankan perulangan tak terbatas yang meminta pengguna menginputkan dua data, jika kedua input tersebut cocok dengan kredensial yang benar, perintah break akan menghentikan perulangan, namun jika tidak, variabel penghitung failed akan bertambah satu di setiap iterasi. Setelah loop berhenti (login berhasil), program menampilkan total angka percobaan login yang gagal ke layar.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var n int

    fmt.Print("Masukan: ")
    fmt.Scan(&n)

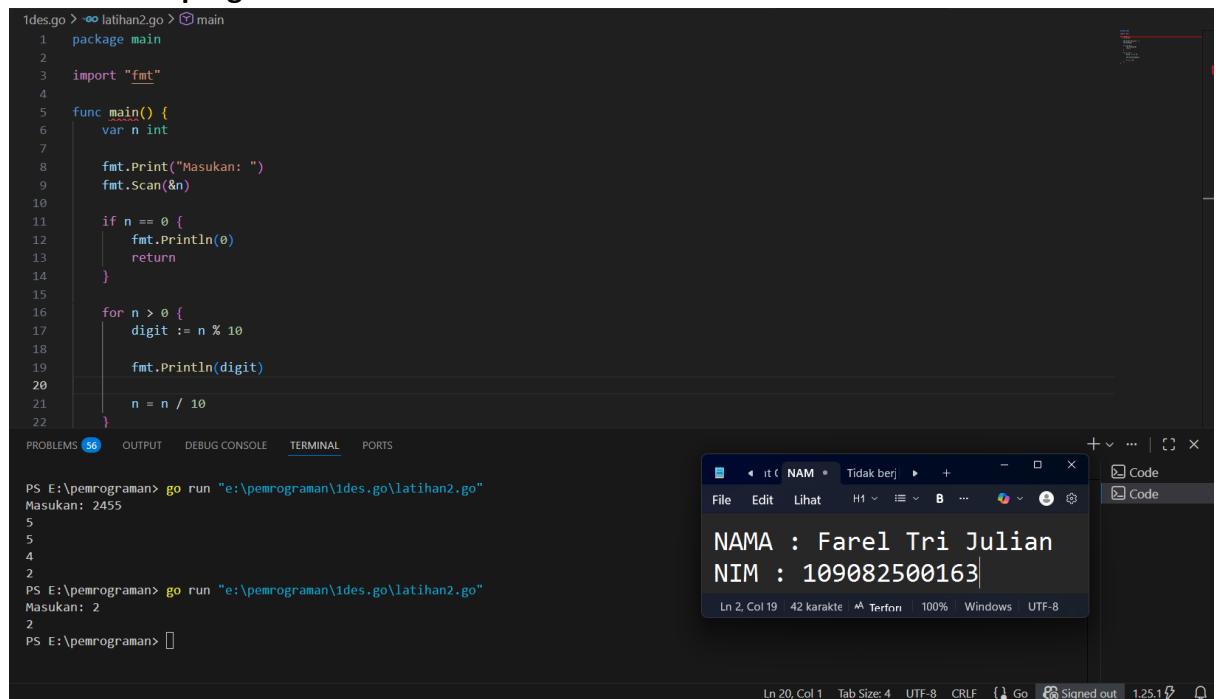
    if n == 0 {
        fmt.Println(0)
        return
    }

    for n > 0 {
        digit := n % 10

        fmt.Println(digit)

        n = n / 10
    }
}
```

Screenshoot program



```
1des.go > ⌂ latihan2.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var n int
7
8      fmt.Print("Masukan: ")
9      fmt.Scan(&n)
10
11     if n == 0 {
12         fmt.Println(0)
13         return
14     }
15
16     for n > 0 {
17         digit := n % 10
18
19         fmt.Println(digit)
20
21         n = n / 10
22     }
}
PROBLEMS 56 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\pemrograman> go run "e:\pemrograman\1des.go\latihan2.go"
Masukan: 2455
5
5
4
2
PS E:\pemrograman> go run "e:\pemrograman\1des.go\latihan2.go"
Masukan: 2
2
PS E:\pemrograman> 
```

The screenshot shows a Go code editor interface. On the left, there is a code editor window displaying a Go program named `latihan2.go` which prints the digits of an input number in reverse order. Below the code editor is a terminal window showing the execution of the program with two different inputs: `2455` and `2`. The terminal output shows the digits printed on new lines. On the right, there is a dark-themed terminal window showing the user's name and NIM.

Deskripsi program

Program ini bertujuan untuk memecah dan menampilkan setiap digit dari sebuah bilangan bulat positif secara terbalik menggunakan operasi matematika dalam sebuah perulangan. Setelah menerima input bilangan n dan menangani kasus khusus untuk angka 0, program memasuki blok perulangan for (yang berfungsi sebagai while-loop) yang berjalan selama nilai n positif di setiap iterasi, program mengekstraksi digit terakhir menggunakan operasi modulus ($n \% 10$), mencetaknya ke baris baru, dan kemudian memangkas digit tersebut dari bilangan utama menggunakan pembagian integer ($n / 10$) hingga seluruh digit selesai diproses.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var x, y int

    fmt.Println("masukan nilai x dan y : ")
    fmt.Scan(&x, &y)

    j := 0

    for x >= y {
        x = x - y
        j++
    }
    fmt.Println("keluaran: ", j)

}
```

Screenshoot program

The screenshot shows a code editor interface with a terminal window below it. The code editor displays a Go program named `ldes.go` with the following content:

```
ldes.go > 1des.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x, y int
7
8     fmt.Print("masukan nilai x dan y : ")
9     fmt.Scan(&x, &y)
10
11    j := 0
12
13    for x >= y {
14        x = x - y
15        j++
16    }
17    fmt.Println("keluaran: ", j)
18
19 }
```

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

```
PROBLEMS 56 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\pemrograman> go run "e:\pemrograman\ldes.go\latihan3.go"
masukan nilai x dan y : 10 2
keluaran: 5
PS E:\pemrograman> go run "e:\pemrograman\ldes.go\latihan3.go"
masukan nilai x dan y : 5 2
keluaran: 2
PS E:\pemrograman> 
```

The terminal also displays the output of the program, which is "NAMA : Farel Tri Julian" and "NIM : 109082500163".

Deskripsi program

Program ini berfungsi untuk menghitung hasil pembagian bilangan bulat dari dua angka input, x dan y , dengan menerapkan metode pengurangan berulang sebagai pengganti operator pembagian standar. Setelah membaca kedua nilai input, program menjalankan perulangan for yang terus mengurangi nilai x dengan y selama x masih lebih besar atau sama dengan y , sembari mencatat berapa kali pengurangan tersebut terjadi menggunakan variabel penghitung j . Ketika perulangan berhenti (karena sisa x sudah kurang dari y), program menampilkan nilai j yang merepresentasikan hasil bagi dari operasi tersebut.