

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 13

REPEAT-UNTIL



Disusun oleh:

Dharma Chandra Viriya

109082500052

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 - Program Menampilkan Teks Sesuai Angka Yang Diinputkan

```
package main

import "fmt"

func main() {
    var text string
    var counter int

    fmt.Print("Masukkan teks: ")
    fmt.Scanln(&text)
    fmt.Print("Masukkan banyak keluaran (angka): ")
    fmt.Scanln(&counter)

    i := 0
    for {
        fmt.Println(text)
        i++
        if i == counter {
            break
        }
    }
}
```

Screenshot program

The screenshot shows a Microsoft Visual Studio Code interface. On the left is the Explorer sidebar with a tree view of files under 'PERTEMUAN13' (Guided1, Guided2, Guided3, Soal1, Soal2, Soal3). The main editor area displays a Go file named 'main.go' with the following code:

```
package main
import "fmt"
func main() {
    var text string
    var counter int
    fmt.Println("Masukkan teks: ")
    fmt.Scanln(&text)
    fmt.Println("Masukkan banyak keluaran (angka): ")
    fmt.Scanln(&counter)
    i := 0
    for {
        fmt.Println(text)
        i++
        if i == counter {
            break
    }
}
```

The terminal tab at the bottom shows the output of running the program:

```
Masukkan teks: pagi
Masukkan banyak keluaran (angka): 3
pagi
pagi
pagi
• iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan13$ go run Guided1/main.go
Masukkan teks: kursi
Masukkan banyak keluaran (angka): 5
kursi
kursi
kursi
kursi
kursi
```

Deskripsi program

Kode program di atas merupakan program untuk menampilkan sebuah teks secara berulang sesuai dengan jumlah yang ditentukan oleh user. Program diawali dengan deklarasi package main dan impor library fmt untuk menangani proses input dan output. Pada function main, program mendeklarasikan variabel text bertipe string dan counter bertipe int. Program meminta input dari user untuk mengisi kedua variabel tersebut menggunakan fmt.Scanln. Selanjutnya, program melakukan perulangan menggunakan for dan menampilkan teks ke console hingga jumlah keluaran sesuai dengan nilai counter, kemudian perulangan dihentikan menggunakan break.

2. Guided 2 - Program Meminta User Terus Menginputkan Angka Sampai Bilangan Bulat Positif

```
package main

import "fmt"

func main() {
    var n int
    for {
        fmt.Print("Masukkan angka: ")
        fmt.Scanln(&n)
        if n > 0 {
            fmt.Printf("%d adalah bilangan bulat positif\n", n)
            break
        }
    }
}
```

Screenshot program

The screenshot shows a dark-themed instance of Visual Studio Code. In the left sidebar, there's a file tree under the 'PERTEMUAN13' folder. The 'main.go' file in the 'Guided2' folder is currently selected. The code editor displays the following Go code:

```
package main

import "fmt"

func main() {
    var n int
    for {
        fmt.Print("Masukkan angka: ")
        fmt.Scanln(&n)
        if n > 0 {
            fmt.Printf("%d adalah bilangan bulat positif\n", n)
            break
        }
    }
}
```

In the bottom right corner of the code editor, there's a floating panel with the user information: 'Dharma Chandra Viriya' and the ID '10908250052'. At the bottom of the screen, the terminal window shows the command `go run Guided2/main.go` being run, followed by several inputs and outputs indicating the program's behavior.

Deskripsi program

Kode program di atas merupakan program untuk memeriksa apakah angka yang dimasukkan merupakan bilangan bulat positif. Program diawali dengan deklarasi package main dan impor library fmt untuk proses input dan output. Pada function main, program mendeklarasikan variabel n bertipe int. Program kemudian meminta input angka dari user menggunakan fmt.Scanln dan melakukan perulangan menggunakan for. Jika nilai n lebih dari 0, maka program akan menampilkan pesan bahwa angka tersebut merupakan bilangan bulat positif, kemudian perulangan dihentikan menggunakan break.

3. Guided 3 - Program Cek Kelipatan

Source Code

```
package main

import "fmt"

func main() {
    var x, y int
    var isKelipatan bool

    fmt.Print("Masukkan x: ")
    fmt.Scanln(&x)
    fmt.Print("Masukkan y: ")
    fmt.Scanln(&y)

    for {
        x -= y
        fmt.Println(x)
        if x == 0 {
            isKelipatan = true
            break
        }

        if x < 0 {
```

```

        isKelipatan = false
        break
    }
}

fmt.Println(isKelipatan)
}

```

Screenshot program

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows a folder structure under "PERTEMUAN13" containing several subfolders ("Guided1", "Guided2", "Guided3") and files ("main.go").
- Code Editor (Center):** Displays the content of "main.go" which includes imports for "main" and "fmt", defines a main function that reads two integers from the user, and prints whether the first integer is divisible by the second.
- Terminal (Bottom):** Shows the output of running the program twice. In the first run, it asks for "Masukkan x: 5" and "Masukkan y: 2", then prints "false". In the second run, it asks for "Masukkan x: 15" and "Masukkan y: 3", then prints "true".
- Status Bar:** Shows the current file path as "Pertemuan13", the date and time as "Dec 14 23:48", and other system information.

Deskripsi program

Kode program di atas merupakan program untuk menentukan apakah suatu bilangan merupakan kelipatan dari bilangan lainnya. Program diawali dengan deklarasi package main dan impor library fmt untuk menangani proses input dan output. Pada function main, program mendeklarasikan variabel x dan y bertipe int, serta variabel isKelipatan bertipe bool. Program meminta input nilai x dan y dari user menggunakan fmt.Scanln. Selanjutnya, program melakukan perulangan dengan mengurangkan nilai y dari x secara berulang. Jika hasil pengurangan bernilai 0, maka x merupakan kelipatan y dan nilai isKelipatan akan bernilai true. Namun, jika hasil pengurangan bernilai kurang dari 0, maka x bukan kelipatan y dan nilai isKelipatan akan bernilai false. Hasil pengecekan kemudian ditampilkan ke console.

TUGAS

1. Tugas 1 - Program Hitung Banyak Digit Angka

Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Print("Masukkan angka: ")
    fmt.Scanln(&n)
    i := 0
    for {
        n /= 10
        i++
        if n == 0 {
            break
        }
    }
    fmt.Println(i)
}
```

Screenshoot program

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

- File Explorer (Left):** Shows a folder structure under "PERTEMUAN13" containing "Guided1", "Guided2", "Guided3", "Soal1", "Soal2", and "Soal3", each with a "main.go" file.
- Code Editor (Center):** Displays a Go code snippet in the "main.go" file of the "Soal1" folder. The code imports "fmt", declares a package "main", and defines a function "main()". It prompts the user to input a number, reads it using fmt.Scanln, initializes a counter "i" to 0, and uses a for loop to divide the number by 10 until it becomes 0. It then prints the value of "i" using fmt.Println.
- Terminal (Bottom):** Shows the output of running the program three times with different inputs: 234, 78787, and 1. The terminal also displays the user's session information: "iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan13\$".
- Status Bar (Bottom):** Provides information about the current file, line, column, encoding, and other settings.

Deskripsi program

Kode program di atas merupakan program untuk menghitung jumlah digit dari sebuah bilangan bulat. Program diawali dengan deklarasi package main dan impor library fmt untuk proses input dan output. Pada function main, program mendeklarasikan variabel n bertipe int dan meminta input angka dari user menggunakan fmt.Scanln. Selanjutnya, program melakukan perulangan dengan membagi nilai n dengan 10 secara berulang sambil menambah nilai variabel penghitung i. Perulangan akan berhenti ketika nilai n menjadi 0, dan program menampilkan jumlah digit dari angka tersebut ke console.

2. Tugas 2 - Program Untuk Mendapatkan Bilangan Optimal Dari Bilangan Yang Diinputkan
- Source code**

```
package main

import "fmt"

func main() {
    var bil float64
    fmt.Print("Masukkan angka: ")
    fmt.Scanln(&bil)

    i := int(bil * 10)
    end := int(bil)*10 + 10

    for {
        i++
        fmt.Printf("%.1f\n", float64(i)/10)

        if i >= end {
            break
        }
    }
}
```

Screenshot program

```
package main

import "fmt"

func main() {
    var bil float64
    fmt.Println("Masukkan angka: ")
    fmt.Scanln(&bil)

    i := int(bil * 10)
    end := int(bil)*10 + 10

    for {
        i++
        fmt.Printf("%.1f\n", float64(i)/10)

        if i >= end {
    
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE DEVDB

```
● iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan13$ go run Soal2/main.go
Masukkan angka: 2.7
2.8
2.9
3.0
● iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan13$ go run Soal2/main.go
Masukkan angka: 0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
```

Deskripsi program

Kode program di atas merupakan program untuk menampilkan deret bilangan desimal dengan kenaikan 0,1 dari suatu nilai awal. Program diawali dengan deklarasi package main dan impor library fmt untuk proses input dan output. Pada function main, program mendeklarasikan variabel bil bertipe float64 dan meminta input angka dari user. Selanjutnya, program mengonversi nilai tersebut ke bentuk bilangan bulat untuk mengatur proses perulangan. Program kemudian menampilkan bilangan desimal secara bertahap hingga mencapai batas tertentu, dan perulangan dihentikan menggunakan break.

3. Tugas 3 - Program Mengumpulkan Donasi

Source code

```
package main

import "fmt"

func main() {
    var target int
    fmt.Print("Masukkan target donasi: ")
    fmt.Scanln(&target)

    total := 0
    donatur := 0

    for {
        var sumbangan int
        fmt.Print("Masukkan donasi: ")
        fmt.Scanln(&sumbangan)

        donatur++
        total += sumbangan

        fmt.Printf(
            "Donatur %d: Menyumbang %d. Total
terkumpul: %d\n",
            donatur, sumbangan, total,
        )
    }

    if total >= target {
        break
    }
}

fmt.Printf(
```

```

        "Target tercapai! Total donasi: %d dari %d
        donatur.\n",
    total, donatur,
)
}

```

Screenshot program

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

- File Explorer:** Shows a project structure under 'PERTEMUAN13' containing 'Guided1', 'Guided2', 'Guided3', 'Soal1', 'Soal2', and 'Soal3'. Each folder contains a 'main.go' file.
- Code Editor:** The 'Soal3/main.go' file is open, displaying the following Go code:


```

package main

import "fmt"

func main() {
    var target int
    fmt.Println("Masukkan target donasi: ")
    fmt.Scanln(&target)

    total := 0
    donatur := 0

    for {
        var sumbangan int
        fmt.Println("Masukkan donasi: ")
        fmt.Scanln(&sumbangan)
      }
```
- Terminal:** The terminal shows two runs of the program:
 - First run: Target donasi: 300. Donatur 1: Menyumbang 150. Total terkumpul: 150. Donatur 2: Menyumbang 200. Total terkumpul: 350. Target tercapai! Total donasi: 350 dari 2 donatur.
 - Second run: Target donasi: 600. Donatur 1: Menyumbang 500. Total terkumpul: 500. Donatur 2: Menyumbang 200. Total terkumpul: 700. Target tercapai! Total donasi: 700 dari 2 donatur.

Deskripsi program

Kode program di atas merupakan program untuk menghitung total donasi hingga mencapai target yang ditentukan. Program diawali dengan deklarasi package main dan impor library fmt untuk menangani proses input dan output. Pada function main, program mendeklarasikan variabel target untuk menyimpan target donasi, serta variabel total dan donatur untuk menyimpan total donasi dan jumlah donatur. Program kemudian meminta input donasi dari user secara berulang. Setiap donasi akan dijumlahkan ke dalam total dan jumlah donatur akan bertambah. Program akan terus menerima donasi hingga total donasi mencapai atau melebihi target yang ditentukan. Setelah target tercapai, program akan menampilkan pesan bahwa target donasi telah terpenuhi beserta total donasi dan jumlah donatur yang telah berpartisipasi.