

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL [13]

REPEAT

UNTIL



Disusun oleh:

Rasya Putra Wibowo

109082500132

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 word string

    var repetitions int

    fmt.Scan(&word, &repetitions)

    counter := 0


    for done := false; !done; {

        fmt.Println(word)

        counter++


        done = (counter >= repetitions)

    }

}
```

Screenshot program

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

- EXPLORER:** Shows a tree view of project files across three folders: TUGAS LAPRAK 10, TUGAS LAPRAK 11, and TUGAS LAPRAK 12. Under TUGAS LAPRAK 13, there are files guide1.go, guide2.go, guide3.go, soal1.go, soal2.go, and soal3.go.
- CODE EDITOR:** The main editor window displays the source code for guide1.go, which contains the logic for reading a word and repetitions from standard input, then printing the word repeatedly until the counter reaches or exceeds the repetitions value.
- TERMINAL:** The terminal window shows the command `PS C:\Users\ACER\OneDrive\Documents\TUGAS LAPORAN PRAKTIKUM> go run "c:\User` followed by several lines of output.
- OUTPUT:** The output window shows the printed words: "pagi", "pagi", "pagi", and "pagi".
- PROBLEMS:** The problems pane shows 21 errors, likely related to the Go code.
- DEBUG CONSOLE:** The debug console pane shows the command `PS C:\Users\ACER\OneDrive\Documents\TUGAS LAPORAN PRAKTIKUM> go run "c:\User`.
- PORTS:** The ports pane is empty.

Deskripsi Program

Program Go ini meminta pengguna untuk memasukkan sebuah kata (string) dan jumlah pengulangan (integer). Kemudian, program akan mencetak kata tersebut ke konsol sebanyak jumlah pengulangan yang dimasukkan oleh pengguna. Program ini menggunakan loop for untuk melakukan pengulangan, dan variabel done untuk mengontrol kapan loop harus berhenti.

2. Guided 2 Source Code

```
package main

import "fmt"

func main() {

    var number int

    var continueLoop bool

    for continueLoop = true; continueLoop; {

        fmt.Scan(&number)

        continueLoop = number <= 0

    }

    fmt.Printf("%d adalah bilangan bulat
positif\n", number)

}

}
```

Screenshot program

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

- Explorer:** Shows a tree view of files under "TUGAS LAPRAK 13".
 - TUGAS LAPRAK 10: guide1.go, guide2.go, guide3.go, soal1.go, soal2.go, soal3.go
 - TUGAS LAPRAK 11: guide1.go, guide2.go, guide3.go, soal1.go, soal2.go, soal3.go
 - TUGAS LAPRAK 12: guide1.go, guide2.go, guide3.go, soal1.go, soal2.go, soal3.go
 - TUGAS LAPRAK 13:
 - guide1.go (5)
 - guide2.go (1)
 - guide3.go (1)
 - soal1.go (1)
 - soal2.go (1)
 - soal3.go (1)
- Terminal:** Shows the command "go run" being run, outputting:

```
PS C:\Users\ACER\OneDrive\Documents\TUGAS LAPRAK 13> go run "c:\Users\ACER\OneDrive\Documents\TUGAS LAPRAK 13\guide2.go"
5
-2
-1
0
5
5 adalah bilangan bulat positif
17
17 adalah bilangan bulat positif
```
- Output:** Shows the output of the program, which prints "5" and "-2" followed by an error message.
- Modal Window:** A "NAMA" window is open, displaying:

NAMA	NIM
RASYA PUTRA WIBOWO	109082500132

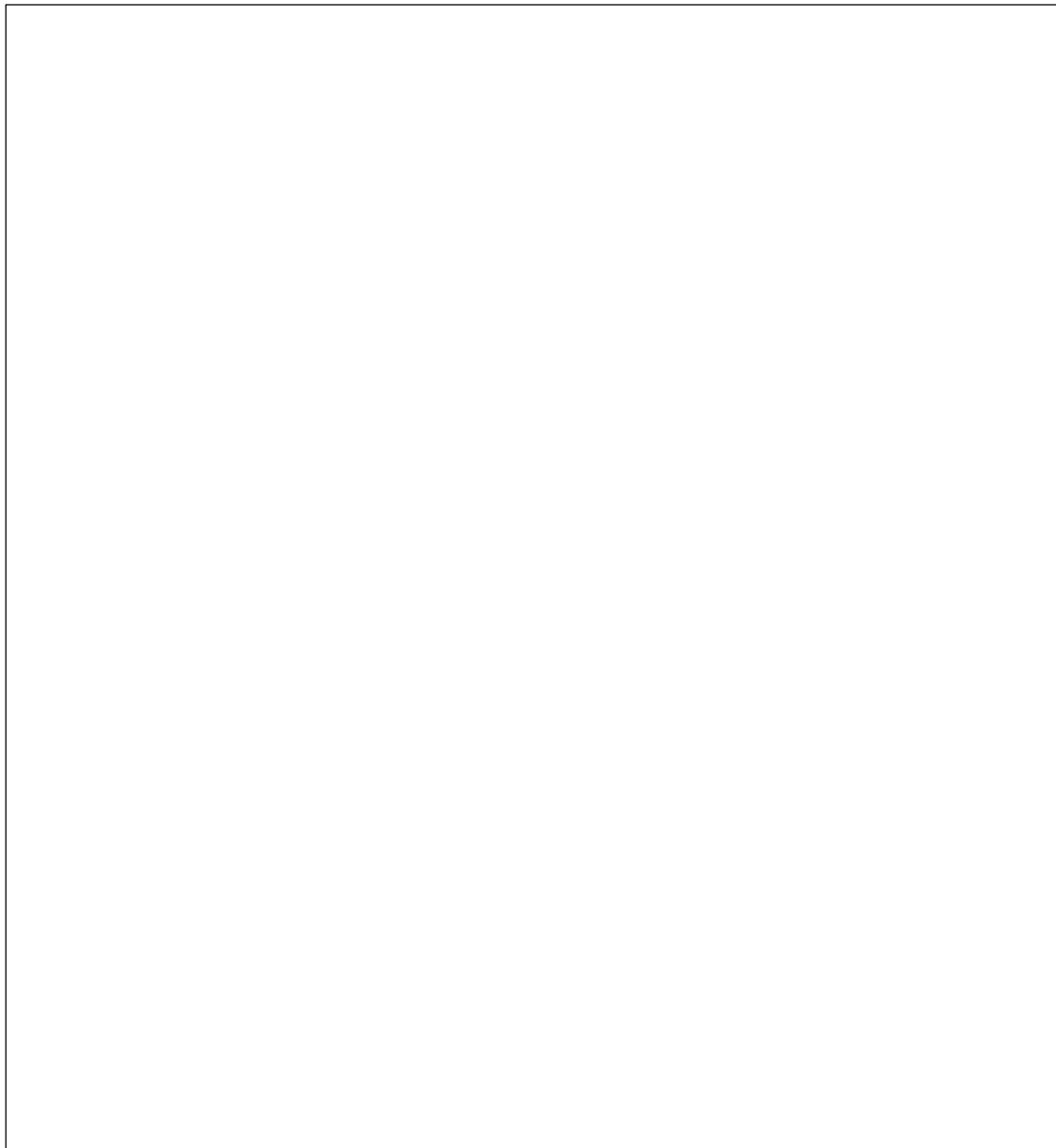
Deskripsi Program

Program Go ini membaca angka dari pengguna berulang kali sampai angka non-positif dimasukkan. Kemudian, program secara keliru mencetak angka non-positif terakhir yang dimasukkan sebagai "bilangan bulat positif". Program ini memiliki kesalahan logika dalam kondisi loop dan outputnya.

3. Guided 3 Source Code

```
package main
import "fmt"
func main() {
    var x int
    var y int
    var selesai bool
    fmt.Scan(&x, &y)

    for selesai = false; !selesai; {
        x = x - y
        fmt.Println(x)
        selesai = x <= 0
    }
    fmt.Println(x == 0)
}
```



Screenshot program

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

- File Explorer:** Shows a tree view of files across three projects:
 - TUGAS LAPORAN PRAKTIKUM: guide2.go, guide3.go, soal1.go, soal2.go, soal3.go
 - TUGAS LAPRAK 11: guide1.go, guide2.go, guide3.go, soal1.go, soal2.go, soal3.go
 - TUGAS LAPRAK 12: guide1.go, guide2.go, guide3.go, soal1.go, soal2.go, soal3.go
- Code Editor:** The main editor window displays the content of `guide3.go` from the TUGAS LAPRAK 13 project. The code is as follows:1 package main
2 import "fmt"
3 func main() {
4 var x int
5 var y int
6 var selesai bool
7 fmt.Scan(&x, &y)
8
9 for selesai = false; !selesai; {
10 x = x - y
11 fmt.Println(x)
12 selesai = x <= 0
13 }
14 fmt.Println(x == 0)
15 }
- Terminal:** A terminal window at the bottom shows the command `go run "c:\Users\ACER\OneDrive\Documents\TUGAS LAPORAN PRAKTIKUM\guide3.go"` being run, with the output:

```
NAMA:RASYA PUTRA WIBOWO
NIM:109082500132
```
- Status Bar:** The status bar at the bottom right indicates the file is `LAPRAK 13\guide3.go`, with line 2, column 17, 40 characters, 100% zoom, and UTF-8 encoding.

Deskripsi Program

Program Go ini menerima dua input integer, x dan y. Kemudian, program mengurangkan y dari x dalam sebuah loop, mencetak nilai x setiap iterasi, hingga x menjadi kurang dari atau sama dengan 0. Terakhir, program mencetak apakah x sama dengan 0 setelah loop selesai.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

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

    count := 0

    for {
        count++
        n = n / 10

        if n == 0 {
            break
        }
    }

    fmt.Println(count)
}
```

Screenshot program

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

- EXPLORER:** Shows a tree view of files and folders under "TUGAS LAPORAN PRAKTIKUM". Folders include TUGAS LAPRAK 10, 11, and 12, each containing several .go files. The current folder is TUGAS LAPRAK 13.
- CODE EDITOR:** Displays the content of `soal1.go` in TUGAS LAPRAK 13.1. The code reads an integer `n` from input, counts the number of digits by repeatedly dividing by 10 until `n` becomes 0, and prints the count.
- TERMINAL:** Shows the command `go run "c:\Users\ACER\OneDrive\Documents\TUGAS LAPORAN PRAKTIKUM\TUGAS LAPRAK 13\soal1.go"` being run, followed by the output `5`.
- OUTPUT:** Shows the output of the terminal command.
- PROBLEMS:** Shows 21 problems, all related to `soal1.go`.
- DEBUG CONSOLE:** Not visible in the screenshot.
- PROMPTS:** Shows the user's name and NIM.
- FILE:** Shows the file `soal1.go` with its details: NAMA: RASYA PUTRA WIBOWO, NIM: 189882500132.
- CODE:** Shows the code of `soal1.go`.

Deskripsi Program

Program Go ini membaca sebuah integer `n` dari input. Program ini kemudian menghitung jumlah digit pada integer `n` dengan membagi `n` dengan 10 secara berulang-ulang sampai `n` menjadi 0. Variabel `count` menyimpan jumlah digit, yang kemudian dicetak ke konsol.

2. Tugas 2

Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var x float64
    fmt.Scan(&x)

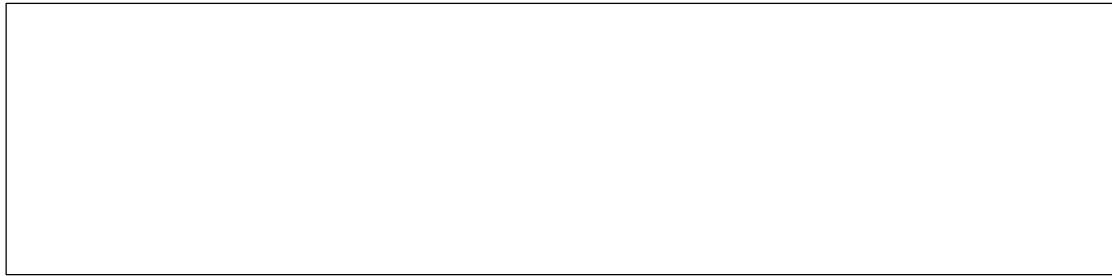
    target := math.Ceil(x)
    current := x

    for {
        current = current + 0.1

        current = math.Round(current*10) / 10

        fmt.Println(current)

        if current >= target {
            break
        }
    }
}
```



Screenshoot program

The screenshot shows a Go development environment. The terminal window displays the following Go code:

```
1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9     var x float64
10    fmt.Scan(&x)
11
12    target := math.Ceil(x)
13    current := x
14
15    for {
16        current = current + 0.1
17
18        current = math.Round(current*10) / 10
19
20        fmt.Println(current)
21
22        if current >= target {
23            break
24        }
25    }
26 }
```

The terminal output shows the execution of the program:

```
PS C:\Users\ACER\OneDrive\Documents\TUGAS LAPORAN PRAKTIKUM> go run "c:\Users\ACER\OneDrive\Documents\TUGAS LAPORAN PRAKTIKUM\soal2.go"
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
```

A separate window titled "NAMA" shows the student's information:

File	Edit	Lihat	H1
NAMA: RASYA PUTRA WIBOWO NIM: 109082500132						

Deskripsi Program

Program Go ini membacax dari input. Program ini kemudian menghitung nilai target sebagai pembulatan ke atas dari x menggunakan math.Ceil. Program ini memulai current dengan nilai x dan kemudian dalam sebuah loop, menambahkan 0.1 ke current, pembulatkan current ke satu tempat desimal menggunakan math.Round, dan mencetak nilai current. Loop berlanjut sampai current lebih besar atau sama dengan target.

3. Tugas 3

Source code

```
package main

import "fmt"

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

    total := 0
    donatur := 0

    for {
        var donasi int
        fmt.Scan(&donasi)

        donatur++
        total += donasi

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

        if total >= target {
            break
        }
    }

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

Screenshot program

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

- File Explorer:** Shows multiple Go files across three projects: TUGAS LAPORAN PRAKTIKUM, TUGAS LAPRAK 11, and TUGAS LAPRAK 13.
- Code Editor:** The file `soal3.go` is open, displaying the provided Go code.
- Terminal:** The terminal window shows the command `go run "c:\Users\ACER\OneDrive\Documents\TUGAS LAPORAN PRAKTIKUM"` being run.
- Output Window:** The output shows the execution of the program, including user input for target and donations, and the final output message indicating the target was reached.

Deskripsi Program.

Program Go ini mensimulasikan penggalangan dana. Pertama, program membaca target donasi dari input. Kemudian, program memasuki loop di mana ia terus-menerus membaca jumlah donasi dari input, menghitung jumlah donatur, dan mengakumulasikan total donasi. Setelah setiap donasi, program mencetak jumlah donatur saat ini, jumlah donasi, dan total dana yang terkumpul. Loop berlanjut hingga total donasi mencapai atau melebihi target. Terakhir, program mencetak pesan yang menyatakan bahwa target telah tercapai, bersama dengan total donasi dan jumlah donatur.