

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

MODUL 13

REPEAT-UNTIL



Disusun oleh:

Anindya Rahadita Yumnaa

109082500138

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

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

```
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)
    }
}
```

```
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\guided 1\guided 1.go"
pagi
pagi
pagi
pagi
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\guided 1\guided 1.go"
kursi
kursi
kursi
kursi
kursi
```

LightClock

```
109082500138
SIIF-13-07
ANINDYA RAHADITA YUMNAA
```

Deskripsi program

Program ini digunakan untuk menerima sebuah kata (string) dan sebuah bilangan bulat (integer) sebagai input dari pengguna, di mana bilangan bulat tersebut menentukan jumlah pengulangan. Sebuah perulangan yang akan mencetak kata yang diinputkan secara berulang-ulang. Program menggunakan sebuah variabel penghitung (counter) yang terus bertambah di setiap putaran loop, dan perulangan akan dihentikan ketika penghitung tersebut telah mencapai atau melebihi jumlah pengulangan yang diminta, memastikan kata tersebut dicetak tepat sesuai jumlah yang diinginkan.

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 the Visual Studio Code interface. In the center-left, there is a code editor window displaying a Go file named 'guided 2.go'. The code defines a main function that reads an integer from standard input, checks if it's less than or equal to zero, and then prints a message indicating whether the number is positive or not. Below the code editor is a terminal window showing the execution of the program and its output. The terminal output includes several negative numbers followed by the message '5 adalah bilangan bulat positif' and '17 adalah bilangan bulat positif'.

```

package main
import "fmt"
func main() {
    var number int
    var continueLoop bool
    for continueLoop = true; continueLoop; {
        fmt.Scan(&number)
        continueLoop = number <= 0
        if number > 0 {
            fmt.Printf("%d adalah bilangan bulat positif\n", number)
        }
    }
}

PS C:\Users\MSI> go run "c:/Users/MSI/Documents/VSCode/MINGGU 13\LAPRAK\guided 2\guided 2.go"
-5
-2
-1
-5
-2
-1
0
5
17
17 adalah bilangan bulat positif
17 adalah bilangan bulat positif
PS C:\Users\MSI>

```

Deskripsi program

Program ini digunakan untuk menampilkan bilangan bulat positif, melalui mekanisme perulangan bersyarat. Program menginisialisasi variabel **continueLoop** menjadi true untuk memastikan perulangan (for) berjalan. Di setiap putaran, program akan meminta dan menerima input bilangan bulat. Perulangan akan terus diulang selama nilai input tersebut adalah nol atau bilangan negatif. Begitu memasukkan bilangan bulat yang lebih besar dari nol (positif), kondisi loop berubah menjadi true, perulangan dihentikan, dan program kemudian mencetak bilangan positif terakhir yang berhasil didapatkan sebagai output akhir.

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 the Visual Studio Code interface with a Go file named `guided 3.go` open. The code implements a simple subtraction loop until the result is less than or equal to zero, then prints the result.

```

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)
}

```

The terminal window shows the execution of the program:

```

PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\guided 3\guided 3.go"
5 2
3
1
-1
false
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\guided 3\guided 3.go"
15 3
12
9

```

A floating terminal window titled "LightClock" displays the output of the program:

```

109082500138
S1F-13-07
ANINDYA RAHADITA YUMNAA

```

The screenshot shows a Microsoft Visual Studio Code interface. In the center-left, there's a code editor with tabs for 'guided 1.go', 'guided 2.go 1', 'guided 3.go 1' (which is currently active), 'soal 1.go 1', 'soal 2.go 1', and 'soal 3.go 1'. The code in 'guided 3.go' is:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x int
7     var v int
```

Below the code editor is a terminal window showing the command 'go run' followed by the path to the file. The output of the terminal is:

```
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\guided 3\guided 3.go"
5 2
3
1
-1
false
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\guided 3\guided 3.go"
15 3
12
9
6
3
0
true
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\guided 3\guided 3.go"
25 5
20
15
10
5
0
true
```

To the right of the terminal is a small window titled 'LightClock' containing student information:

109082500138
S1IF-13-07
ANINDYA RAHADITA YUMNAA

At the bottom of the terminal window, status bars show 'Ln 3, Col 24 48 character Plain t 100%' and 'Wind: UTF-8'.

Deskripsi program

Program ini merupakan implementasi dari algoritma pengurangan berulang untuk mensimulasikan proses pembagian. Program diawali dengan menerima dua input bilangan bulat (x sebagai bilangan yang dikurangi dan y sebagai pengurang) dari pengguna. Inti eksekusi terletak pada sebuah perulangan yang dikendalikan oleh **variabel boolean** (selesai). Perulangan akan terus berjalan dan secara konsisten mengurangi nilai x dengan y , sambil mencetak nilai x yang terbaru di setiap langkah, hingga kondisi $x \leq 0$ terpenuhi. Setelah nilai x mencapai atau melewati nol, variabel kontrol selesai diatur menjadi true, yang menghentikan perulangan. Sebagai hasil akhir, program mencetak **nilai boolean** (true atau false) untuk menunjukkan apakah sisa pengurangan terakhir (nilai akhir x) sama dengan nol, yang secara efektif menguji apakah bilangan awal x habis dibagi oleh y .

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
```

```

var n int

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

fmt.Scan(&n)

count := 0

for n > 0 {

    count++

    n = n / 10

}

fmt.Println(count)
}

```

Screenshot program

The screenshot shows the Visual Studio Code interface. On the left, the code editor displays a file named 'soal 1.go' with the following content:

```

package main
import "fmt"
func main() {
    var n int
    fmt.Print("Masukkan bilangan bulat positif: ")
    fmt.Scan(&n)

    count := 0

    for n > 0 {
        count++
        n = n / 10
    }

    fmt.Println(count)
}

```

Below the code editor is the terminal window, which shows the execution of the program:

```

PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 1\soal 1.go"
Masukkan bilangan bulat positif: 5
1
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 1\soal 1.go"
Masukkan bilangan bulat positif: 234
3
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 1\soal 1.go"
Masukkan bilangan bulat positif: 1894256
7
PS C:\Users\MSI>

```

To the right of the terminal, there is a small window titled 'LightClock' displaying the output of the program:

109082500138
S1IF-13-07
ANINDYA RAHADITA YUMNAA

Deskripsi program

Program berfungsi sebagai alat untuk menghitung jumlah digit yang terdapat dalam sebuah bilangan bulat positif yang dimasukkan oleh pengguna. Setelah menerima

input dan menginisialisasi variabel penghitung (count) dengan nol, program menjalankan sebuah perulangan (for) yang berlanjut selama nilai bilangan masukan (n) masih lebih besar dari nol. Di setiap putaran loop, variabel count dinaikkan satu, yang menandakan telah diprosesnya satu digit. Secara simultan, nilai n terus diperbarui melalui operasi **pembagian integer** oleh 10 ($n = n / 10$), yang secara efektif menghilangkan digit paling kanan. Perulangan akan berhenti ketika nilai n menjadi nol, menunjukkan bahwa semua digit telah dipangkas dan dihitung. Hasil akhirnya, yaitu jumlah total digit, kemudian dicetak ke konsol melalui nilai akhir variabel count.

2. Tugas 2

Source code

```
package main

import (
    "fmt"
    "math"
)

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

    limit := math.Ceil(x)

    for x < limit {
        x += 0.1

        x = math.Round(x*10) / 10
    }

    fmt.Println("Jumlah digit adalah", count)
}
```

```
if x == limit {  
    fmt.Printf("%.0f\n", x)  
  
} else {  
    fmt.Printf("%.1f\n", x)  
  
}  
}  
}
```

Screenshot program

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The title bar includes 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', and other standard options. A search bar is positioned above the main workspace. The left side features a vertical toolbar with icons for file operations like Open, Save, Find, and others. The main workspace displays a Go program named 'main.go' with syntax highlighting for Go code. The terminal at the bottom shows the command 'go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 2\soal 2.go"' followed by the output '0.2'. The status bar at the bottom right indicates the current line (Ln 8), column (Col 14), tab size (Tab Size: 4), encoding (UTF-8), and version (1.25.1).

```
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    limit := math.Ceil(x)
13
14    for x < limit {
15        x += 0.1
16
17        x = math.Round(x*10) / 10
18
19        if x == limit {
20            fmt.Printf("%.0f\n", x)
21        } else {
22            fmt.Printf("%.1f\n", x)
23        }
24    }
25 }
```

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

- File Explorer:** Shows files like "guided 1.go", "guided 2.go", "guided 3.go", "soal 1.go", "soal 2.go", and "soal 3.go".
- Search Bar:** Contains the text "Search".
- Terminal:** Displays the command "go run" followed by the path to "soal 2.go". The output shows numerical values from 0.2 to 3.0, representing the simulation results.
- Output Panel:** Shows the output of the "soal 2.go" run, including the printed values and the author's name, ANINDYA RAHADITA YUMNAA.
- Status Bar:** Shows the current file is "soal 2.go", the line number is 8, the column number is 14, the tab size is 4, and the encoding is UTF-8.

Deskripsi program

Program Go ini berfungsi untuk membuktikan proses kenaikan bilangan desimal secara bertahap (0.1) hingga mencapai nilai pembulatan ke atasnya. Program diawali dengan menerima input float64 (x) dan menetapkan target (limit) menggunakan fungsi math.Ceil. Inti program adalah perulangan yang akan terus berjalan selama x kurang dari limit. Di setiap iterasi, nilai x dinaikkan sebesar 0.1, diikuti dengan langkah krusial **math.Round(x*10)/10** yang secara efektif mengoreksi kesalahan presisi floating-point dan menjaga x tetap pada presisi satu desimal. Nilai x kemudian dicetak di setiap langkah, menggunakan format satu desimal (%.1f), kecuali pada langkah terakhir di mana x telah sama dengan limit, yang dicetak sebagai bilangan bulat (%.0f). Dengan demikian, program ini menyajikan urutan bilangan desimal yang meningkat secara terkontrol dari input awal hingga bilangan bulat optimal.

3. Tugas 3

Source code

```
package main

import "fmt"

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

    total := 0
```

```

donatur := 0

for total < target {
    var donasi int
    fmt.Scan(&donasi)

    donatur++
    total += donasi

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

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

```

Screenshoot program

The screenshot shows a Visual Studio Code (VS Code) interface with a dark theme. On the left is the code editor containing a Go program named `main.go`. The code defines a `main()` function that reads a target value from the user, iterates through values until it reaches or exceeds the target, and prints the progress to the terminal. On the right is a terminal window titled `LightClock` showing the execution results:

```

109082500138
S11F-13-07
ANINDYA RAHADITA YUMNAQ

```

The terminal also displays the current line number (Ln 3), column (Col 24), character count (48 characters), and encoding (UTF-8).

The screenshot shows a VS Code interface with a terminal window displaying the output of a Go program. The terminal shows the program's execution and its output. A light clock window is also visible.

```
C:\> Users > MSI > Documents > VSCode > MINGGU 13 > LAPRAK > soal 3 > soal 3.go > main
5 func main() {
20     donatur := 0 // Menyumbang
21     total := 0 // Total terkumpul
22     for ; donatur < target; donatur++ {
23         donatur, total, _ = sumbang(donatur)
24         fmt.Printf("Donatur %d: Menyumbang %d. Total terkumpul: %d\n", donatur, total, total)
25     }
26     if total < target {
27         fmt.Printf("Target tercapai! Total donasi: %d dari %d donatur.\n", total, target)
28     } else {
29     }
}
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 3\soal 3.go"
300 100 50 200
Donatur 1: Menyumbang 100. Total terkumpul: 100
Donatur 2: Menyumbang 50. Total terkumpul: 150
Donatur 3: Menyumbang 200. Total terkumpul: 350
Target tercapai! Total donasi: 350 dari 3 donatur.
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 3\soal 3.go"
500 150 100 50 300
Donatur 1: Menyumbang 150. Total terkumpul: 150
Donatur 2: Menyumbang 100. Total terkumpul: 250
Donatur 3: Menyumbang 50. Total terkumpul: 300
Donatur 4: Menyumbang 300. Total terkumpul: 600
Target tercapai! Total donasi: 600 dari 4 donatur.
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 3\soal 3.go"
500 150 100 50 300
Donatur 1: Menyumbang 150. Total terkumpul: 150
Donatur 2: Menyumbang 100. Total terkumpul: 250
Donatur 3: Menyumbang 50. Total terkumpul: 300
Donatur 4: Menyumbang 300. Total terkumpul: 600
Target tercapai! Total donasi: 600 dari 4 donatur.

```

The screenshot shows a VS Code interface with a terminal window displaying the output of a Go program. The terminal shows the program's execution and its output. A light clock window is also visible.

```
C:\> Users > MSI > Documents > VSCode > MINGGU 13 > LAPRAK > soal 3 > soal 3.go > main
5 func main() {
15     target := 600 // Target tercapai
16     total := 0 // Total terkumpul
17     for ; total < target; total += sumbang() {
18         fmt.Printf("Donatur %d: Menyumbang %d. Total terkumpul: %d\n", len(donatur), sumbang(), total)
19     }
20     if total < target {
21         fmt.Printf("Target tercapai! Total donasi: %d dari %d donatur.\n", total, len(donatur))
22     } else {
23     }
24 }
25 func sumbang() int {
26     var sumba := 0
27     for ; sumba < 100; sumba++ {
28         if sumba == 50 {
29             sumba = 300
30         }
31     }
32     return sumba
33 }
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 3\soal 3.go"
Donatur 1: Menyumbang 100. Total terkumpul: 100
Target tercapai! Total donasi: 600 dari 1 donatur.
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 3\soal 3.go"
Target tercapai! Total donasi: 600 dari 1 donatur.
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 3\soal 3.go"
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 13\LAPRAK\soal 3\soal 3.go"
200 300
Donatur 1: Menyumbang 300. Total terkumpul: 300
Donatur 1: Menyumbang 300. Total terkumpul: 600
Target tercapai! Total donasi: 600 dari 1 donatur.

```

Deskripsi program

Program Go ini berfungsi sebagai simulasi penggalangan dana yang beroperasi berdasarkan target keuangan yang ditetapkan pengguna sebagai input awal. Logika program diatur oleh perulangan yang akan terus dijalankan selama total donasi yang terkumpul masih kurang dari target yang ditentukan. Di setiap iterasi loop, program menerima input sumbangan dari satu donatur, mencatat jumlah donatur yang berpartisipasi, dan menambahkan nilai sumbangan tersebut ke total akumulasi. Untuk baris pertama berupa bilangan bulat yang merupakan target donasi yang harus dicapai. Masukan pada baris berikutnya merupakan bilangan bulat yang menyatakan donasi oleh setiap donatur, masukan terus diterima hingga target tercapai. Program memberikan **feedback** instan dengan mencetak rincian sumbangan dan total terkini. Perulangan akan dihentikan secara otomatis setelah

total donasi mencapai atau melebihi target. Akhirnya, program menyajikan laporan akhir yang mengkonfirmasi pencapaian target serta mencantumkan jumlah total donasi yang berhasil dihimpun dan jumlah total donatur yang terlibat dalam proses tersebut.