

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL 13

REPEAT-UNTIL



Disusun oleh:

Didi Hermawanto

109082500088

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.Print("Masukkan kata dan jumlah pengulangan: ")
    fmt.Scan(&word, &repetitions)

    counter := 0
    done := false

    for !done {
        fmt.Println(word)
        counter++
        done = (counter >= repetitions)
    }
}
```

Screenshot program

The screenshot shows a code editor interface with a terminal window below it. The terminal window displays the execution of a Go program named 'modul1.go'. The program prompts the user for a word and a repetition count, then prints the word repeatedly until the counter reaches or exceeds the repetition count. The terminal also shows the user's input and the resulting output.

```

modul1.go 5
modul1.go > main
6     var repetitions int
7
8     fmt.Print("Masukkan kata dan jumlah pengulangan: ")
9     fmt.Scan(&word, &repetitions)
10
11    counter := 0
12    done := false
13
14    for !done {
15        fmt.Println(word)
16        counter++
17        done = (counter >= repetitions)
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\coding didi\golang\Laprak 13> go run "c:\coding didi\golang\Laprak 13\modul1.go"
● Masukkan kata dan jumlah pengulangan: pagi 3
pagi
pagi
pagi
● PS C:\coding didi\golang\Laprak 13> go run "c:\coding didi\golang\Laprak 13\modul1.go"
Masukkan kata dan jumlah pengulangan: siang 4
siang
siang
siang
siang

Ln 11, Col 17 Spaces: 4 UTF-8 CRLF { } Go 1.25.1 ⚡ (→) Go Live Windsurf: Login ⚡ Prettier

Deskripsi program

Program ini digunakan untuk mencetak sebuah kata berulang kali sesuai jumlah yang dimasukkan oleh pengguna. Pada awal eksekusi, program meminta dua input, yaitu sebuah kata yang disimpan dalam variabel `word` serta angka yang menyatakan banyaknya pengulangan dan disimpan dalam variabel `repetitions`. Selanjutnya, program membuat variabel `counter` yang berfungsi sebagai pencatat berapa kali kata sudah ditampilkan, serta variabel `done` yang digunakan sebagai penentu kapan proses pengulangan harus dihentikan. Proses pencetakan kata dilakukan di dalam perulangan `for` yang akan terus berjalan selama kondisi `done` masih bernilai salah. Setiap kali perulangan dijalankan, kata ditampilkan ke layar menggunakan `fmt.Println`, lalu nilai `counter` ditambah satu untuk menandai satu kali pengulangan. Setelah itu, program membandingkan nilai `counter` dengan jumlah pengulangan yang diminta, dan jika sudah sama atau lebih besar, nilai `done` diubah sehingga perulangan berhenti.

2. Guided 2

Source Code

```

package main

import "fmt"

func main() {
    var n int
    for {
```

```
fmt.Println("Masukkan bilangan bulat positif: ")  
fmt.Scan(&n)  
  
if n > 0 {  
    fmt.Printf("%d adalah bilangan bulat  
positif\n", n)  
    break  
}  
}
```

Screenshot program

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

- Code Editor Area:** Displays the code for `modul2.go`. The code reads a integer from the user and prints it if it's positive.
- Output Panel:** Shows the terminal output for two runs of the program. The first run takes negative numbers (-5, -2, -1, 0) and prints them as positive. The second run takes positive numbers (20) and prints them as positive.
- Windsurf Panel:** A floating panel titled "NIM" containing student information: NIM :109082500088, KELAS :S1IF-13-07, and NAMA :Didi Hermawanto.
- Bottom Navigation:** Includes tabs for PROBLEMS (10), OUTPUT, DEBUG CONSOLE, TERMINAL (selected), and PORTS. It also shows the current file path: C:\coding\didid\golang\Laprak 13\modul2.go.

Deskripsi program

Program ini dirancang untuk memastikan pengguna memasukkan bilangan bulat positif sebelum proses dihentikan. Pada awalnya, program masuk ke dalam perulangan tanpa batas yang bertujuan untuk terus meminta input dari pengguna. Di setiap perulangan, perintah `fmt.Print` digunakan untuk menampilkan pesan agar pengguna memasukkan sebuah bilangan bulat positif, kemudian nilai tersebut dibaca dan disimpan ke dalam variabel `n` menggunakan `fmt.Scan`. Setelah input diterima, program melakukan pengecekan kondisi untuk memastikan apakah nilai yang

dimasukkan lebih besar dari nol. Jika kondisi tersebut terpenuhi, program akan menampilkan pesan bahwa bilangan yang dimasukkan merupakan bilangan bulat positif, lalu menghentikan perulangan dengan perintah break. Dengan alur seperti ini, program hanya akan selesai ketika pengguna memberikan input yang sesuai dengan kriteria yang ditentukan, sehingga mencegah program berhenti sebelum mendapatkan data yang benar.

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)
}
```

Screenshoot program

The screenshot shows a code editor interface with a terminal window. The code editor displays a Go program named 'modul3.go' with the following content:

```
modul3.go 1 ✘
modul3.go > main
5   var x int
6   var y int
7   var selesai bool
8   fmt.Scan (&x, &y)
9
10  for selesai = false; !selesai; {
11      x = x - y
12      fmt.Println (x)
13      selesai = x <= 0
14  }
15  fmt.Println (x == 0) Ctrl+I Windsurf Command
```

The terminal window shows the execution of the program:

```
PS C:\coding didi\golang\Laprak 13> go run "c:\coding didi\golang\Laprak 13\modul3.go"
1
-1
false
● PS C:\coding didi\golang\Laprak 13> go run "c:\coding didi\golang\Laprak 13\modul3.go"
15 3
12
9
6
3
0
true
```

A status bar at the bottom indicates the current line (Ln 16, Col 2), tab size (4), encoding (UTF-8), and other details.

Deskripsi program

Program ini bertujuan untuk mengurangi suatu bilangan dengan bilangan lain secara berulang hingga nilainya mencapai nol atau kurang. Pada awal program, dua buah bilangan bulat dimasukkan oleh pengguna dan disimpan ke dalam variabel `x` dan `y`. Selain itu, terdapat variabel bertipe boolean bernama `selesai` yang digunakan sebagai penanda untuk mengatur jalannya perulangan. Proses utama program berada di dalam perulangan `for` yang berjalan selama kondisi `selesai` bernilai salah. Di setiap putaran, nilai `x` dikurangi dengan `y`, kemudian hasil pengurangan tersebut langsung ditampilkan ke layar. Setelah itu, program mengecek apakah nilai `x` sudah kurang dari atau sama dengan nol; jika kondisi ini terpenuhi, variabel `selesai` diubah sehingga perulangan berhenti. Setelah keluar dari perulangan, program menampilkan hasil perbandingan `x == 0` yang menunjukkan apakah nilai akhir `x` tepat sama dengan nol atau tidak. Dengan alur seperti ini, program dapat memperlihatkan proses pengurangan bertahap sekaligus menentukan apakah pengurangan tersebut berakhir tepat di nol.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var n int
    jumlahDigit := 0

    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)

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

    fmt.Println("Jumlah digit:", jumlahDigit)
}
```

Screenshoot program

The screenshot shows a code editor interface with a dark theme. On the left is the code editor pane displaying a Go file named 'soal1.go'. The code defines a function 'main' that prompts the user for a number, counts its digits, and prints the result. On the right is a terminal window showing the execution of the program. The user inputs various numbers (234, 78787, 1894256) and the program outputs the digit count (3, 5, 7). A status bar at the bottom indicates the current line (Ln 4, Col 14), tab size (Tab Size: 4), encoding (UTF-8), and other terminal settings.

```
soal1.go 1 ×
soal1.go > main
4 var n int
5 jumlahDigit := 0
6
7 fmt.Print("Masukkan bilangan: ")
8 fmt.Scan(&n)
9
10 for n > 0 {
11     jumlahDigit++
12     n = n / 10
13 }
14 fmt.Println("Jumlah digit:", jumlahDigit)

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\coding\didid\golang\Laprak 13> go run "c:\coding\didid\golang\Laprak 13\soal1.go"
● Masukkan bilangan: 4
Jumlah digit: 1
● PS C:\coding\didid\golang\Laprak 13> go run "c:\coding\didid\golang\Laprak 13\soal1.go"
Masukkan bilangan: 234
Jumlah digit: 3
● PS C:\coding\didid\golang\Laprak 13> go run "c:\coding\didid\golang\Laprak 13\soal1.go"
Masukkan bilangan: 78787
Jumlah digit: 5
● PS C:\coding\didid\golang\Laprak 13> go run "c:\coding\didid\golang\Laprak 13\soal1.go"
Masukkan bilangan: 1894256
Jumlah digit: 7

Ln 4, Col 14 Tab Size: 4 UTF-8 CRLF { Go 1.25.1 ⚡ (→) Go Live Windsurf: Login Prettier
```

Deskripsi program

Program ini dibuat untuk menghitung banyaknya digit dari sebuah bilangan bulat positif yang dimasukkan oleh pengguna. Pada awal eksekusi, program meminta pengguna memasukkan sebuah bilangan dan menyimpannya ke dalam variabel `n`. Selanjutnya, variabel `jumlahDigit` disiapkan untuk mencatat berapa kali proses perhitungan digit dilakukan. Proses utama berlangsung di dalam perulangan yang akan terus berjalan selama nilai `n` masih lebih besar dari nol. Di setiap perulangan, nilai `jumlahDigit` ditambah satu sebagai tanda bahwa satu digit telah dihitung, kemudian nilai `n` dibagi sepuluh untuk menghilangkan digit terakhir dari bilangan tersebut. Perulangan akan berhenti ketika nilai `n` sudah habis atau menjadi nol. Setelah itu, program menampilkan hasil akhir berupa jumlah digit dari bilangan yang dimasukkan.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
```

```
var n float64

fmt.Print("Masukkan bilangan desimal: ")

fmt.Scan(&n)

awal := int(n * 10)
batas := int(n)

if n > float64(batas) {
    batas = batas + 1
}

for i := awal + 1; i <= batas*10; i++ {
    if i%10 == 0 {
        fmt.Println(i / 10)
    } else {
        fmt.Printf("%.1f\n", float64(i)/10)
    }
}
```

Screenshot program

```
soal2.go 1 X
soal2.go > main

10     awal := int(n * 10)
11     batas := int(n)
12
13     if n > float64(batas) {
14         batas = batas + 1
15     }
16
17     for i := awal + 1; i <= batas*10; i++ {
18         if i%10 == 0 {
19             fmt.Println(i / 10)
20         } else {
21             fmt.Printf("%.1f\n", float64(i)/10)

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

Masukkan bilangan desimal: 0.2
0.3
0.4
0.5
0.6
0.6
0.7
0.8
0.9
1

NIM :109082500088
KELAS :S1IF-13-07
NAMA :Didi Hermawanto

Deskripsi program

Program ini berfungsi untuk menampilkan hasil penjumlahan bertahap dari sebuah bilangan desimal hingga mencapai nilai pembulatan ke atasnya. Pada awal program, pengguna diminta memasukkan sebuah bilangan desimal yang kemudian disimpan dalam variabel `n`. Nilai tersebut lalu diolah dengan mengalikan sepuluh untuk mendapatkan posisi awal perhitungan dalam bentuk bilangan bulat, yang disimpan pada variabel `awal`. Selanjutnya, program menentukan batas akhir dengan mengambil bagian bilangan bulat dari input dan menambahkannya satu jika masih terdapat nilai pecahan, sehingga diperoleh batas pembulatan ke atas. Proses utama dilakukan melalui perulangan yang berjalan dari nilai setelah awal hingga mencapai batas akhir dalam skala persepuluhan. Di setiap langkah perulangan, program mengecek apakah nilai saat ini merupakan kelipatan sepuluh; jika iya, hasil ditampilkan sebagai bilangan bulat tanpa desimal, sedangkan jika tidak, hasil dicetak dalam bentuk satu angka di belakang koma. Dengan alur tersebut, program mampu menampilkan urutan nilai desimal secara teratur dan konsisten sampai mencapai bilangan bulat tujuan.

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,
)
}

```

Screenshot program

The screenshot shows a code editor interface with a dark theme. On the left, the file `soal3.go` is open, displaying the provided Go code. On the right, a terminal window shows the execution of the program and its output.

Code Editor:

```

10 soal3.go 1 ✘
11 soal3.go > main
12
13     for total < target {
14         var donasi int
15         fmt.Scan(&donasi)
16
17         donatur++
18         total += donasi
19
20         fmt.Printf(
21             "Donatur %d: Menyumbang %d. Total terkumpul: %d\n",
22             donatur, donasi, total,
23         )
24     }
25
26 }

```

Terminal Output:

```

PS C:\coding\didil\golang\Laprak 13> go run "c:\coding\didil\golang\Laprak 13\soal3.go"
● 300
100
Donatur 1: Menyumbang 100. Total terkumpul: 100
50
Donatur 2: Menyumbang 50. Total terkumpul: 150
200
Donatur 3: Menyumbang 200. Total terkumpul: 350
Target tercapai! Total donasi: 350 dari 3 donatur.

```

A small window titled "NIP" is overlaid on the terminal, displaying student information:

NIM	:109082500088
KELAS	:S1IF-13-07
NAMA	:Didi Hermawanto

Deskripsi program

Program ini dibuat untuk menghitung total donasi yang terkumpul hingga mencapai atau melewati target yang telah ditentukan. Pada awalnya, program menerima sebuah bilangan bulat sebagai target donasi dan menyimpannya ke dalam variabel target. Setelah itu, dua variabel lain disiapkan, yaitu total untuk menyimpan akumulasi donasi dan donatur untuk menghitung jumlah penyumbang yang telah berpartisipasi. Proses utama berlangsung di dalam perulangan yang akan terus berjalan selama nilai total donasi masih lebih kecil dari target. Di setiap putaran, program membaca nilai donasi dari seorang donatur, menambahkan jumlah tersebut ke total donasi, serta menaikkan jumlah donatur satu per satu. Informasi mengenai setiap donasi langsung ditampilkan ke layar agar perkembangan total dapat diketahui. Ketika total donasi sudah mencapai atau melebihi target, perulangan dihentikan dan program menampilkan pesan akhir yang menunjukkan bahwa target telah tercapai beserta jumlah donatur dan total donasi yang terkumpul.