

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL 13

REPEAT-UNTIL



Disusun oleh:

MOHAMAD ERLANGGA ZEIN

109082500020

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 kata string

    var kondisi bool

    var jumlahKata int


    fmt.Println("Masukkan kata dan jumlah kata: ")

    fmt.Scan(&kata, &jumlahKata)

    for kondisi = false; !kondisi; {

        fmt.Println(kata)

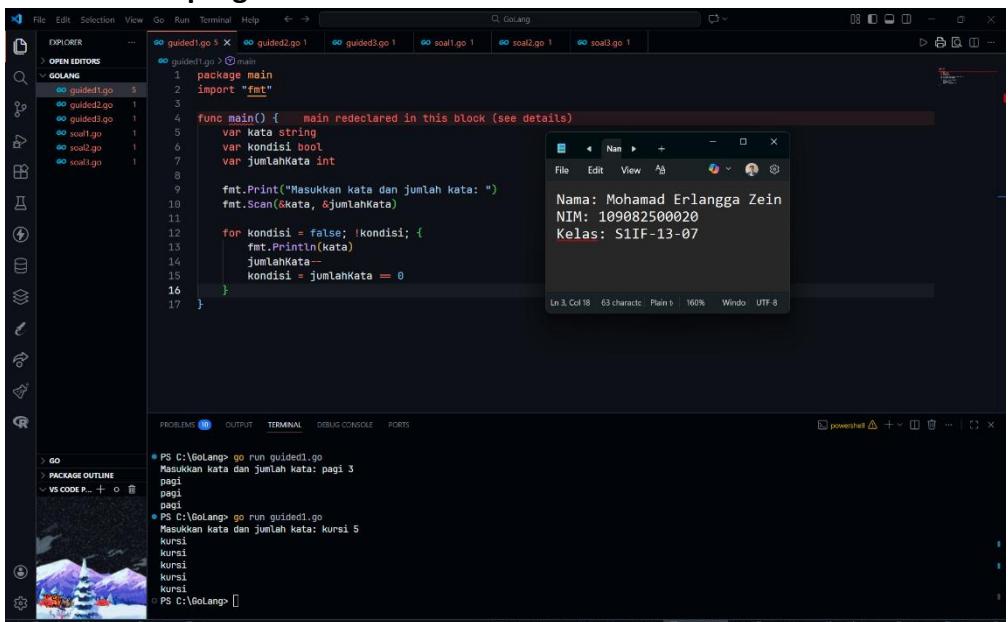
        jumlahKata--

        kondisi = jumlahKata == 0

    }

}
```

Screenshoot program



Deskripsi program

Program ini adalah program untuk menampilkan kata yang sama sebanyak yang diinputkan oleh user, program ini menggunakan algoritma repeat-until yang dimana program akan terus berulang atau program akan tereksekusi sesuai dengan apa permintaan/input an oleh user, program ini menggunakan 3 tipe data yaitu string, boolean, dan integer, dimana untuk algoritmanya adalah disini untuk nilai dari boolean nya sendiri secara default adalah false, dan untuk perulangannya terjadi ketika kondisi nya tidak true atau bisa dibilang false, untuk iterasi nya adalah decrement(pengurangan) dimana perulangannya akan berhenti apabila jumlah kata yang diinput oleh user berkurang sampai dengan 0 (jumlahKata == 0), sehingga output yang dihasilkan adalah kata yang diinputkan oleh user yang tercetak ulang sesuai dengan jumlah kata yang diinputkan oleh user juga, contohnya disini user menginput pagi 3, maka keluarannya adalah kata pagi sebanyak 3 kali(pagi pagi pagi), karena disini menggunakan decrement dan program akan berhenti ketika kondisi dari jumlah kata nya habis(0), maka logikanya seperti ini: pagi 3, 3 apakah == 0? Tidak, maka dikurang 1, lalu 3-1 adalah 2, 2 apakah == 0? Tidak, maka dikurangi lagi, 2-1 apakah == 0? Tidak, maka di kurangi lagi, 1-1 apakah == 0? Ya, maka program akan berhenti di perulangan yang ke 3 dan hasil outputnya adalah kata yang telah diinput oleh user berulang(pagi pagi pagi).

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var n int
    var aselole bool

    for aselole = false; !aselole; {
        fmt.Print("masukkan angka: ")
        fmt.Scan(&n)

        aselole = (n * -1) < 0
    }
}
```

```

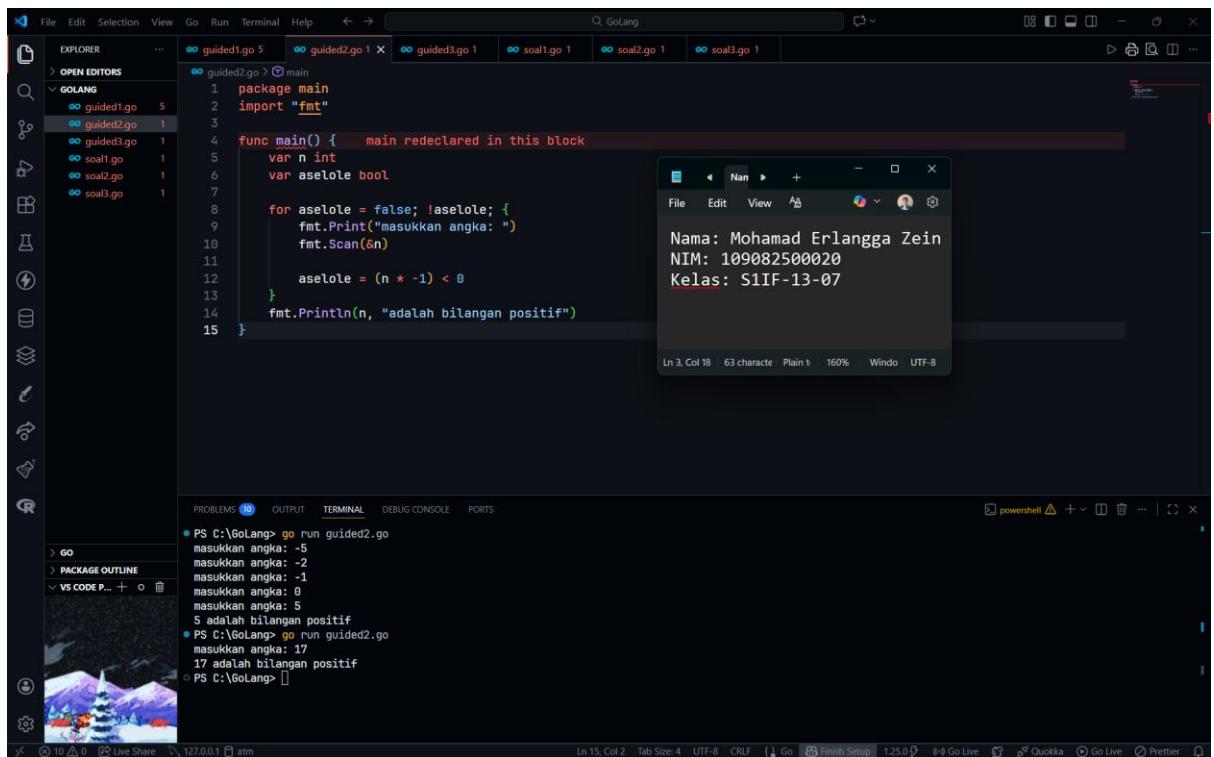
        }

        fmt.Println(n, "adalah bilangan positif")

    }

```

Screenshot program



Deskripsi program

Program ini adalah program untuk mengetahui status dari suatu bilangan bulat, khususnya untuk mengetahui apakah bilangan ini merupakan bilangan bulat positif atau bukan, untuk tipe data nya ada dua yaitu integer dan Boolean, sama seperti pada program sebelumnya, untuk algoritmanya yakni program akan berulang sampai menemukan kondisi yang sesuai(true), dimana kondisi dalam program ini adalah ketika bilangan di kali dengan -1 dan nilainya kurang dari($<$) 0 maka akan bernilai true, sebagai contoh disini user menginputkan angka -5, maka $-5 * -1 = 5$, namun hasilnya > 0 , maka program akan terus berlanjut hingga menemukan hasil < 0 , sebagai contoh ketika user menginputkan 5, maka $5 * -1 = -5$, nilai dari $-5 < 0$, sehingga programnya bernilai true dan perulangannya berhenti.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Scan(&x, &y)

    for kondisi := false; !kondisi; {
        x = x - y
        fmt.Println(x)
        if x <= 0 {
            kondisi = x == 0
            fmt.Print(kondisi)
        }
        kondisi = true
    }
}
```

Screenshot program

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

- Explorer View:** Shows files in the Golang workspace, including guided1.go, guided2.go, guided3.go, soal1.go, soal2.go, and soal3.go.
- Editor View:** Displays the code for `guided3.go`. The code defines a function `main()` that takes two integers `x` and `y`, calculates their difference, and prints the result. It includes logic to determine if the result is zero and prints "true" or "false".

```
package main
import "fmt"

func main() {
    var x, y int
    fmt.Scan(&x, &y)

    for kondisi := false; !kondisi; {
        x = x - y
        fmt.Println(x)
        if x <= 0 {
            kondisi = x == 0
            fmt.Println(kondisi)
        }
        kondisi = true
    }
}
```
- Terminal View:** Shows the command `go run guided3.go` being run in a PowerShell terminal. The output shows various test cases: 5, 2, 3, 1, -1, false; 15, 3, 12, 2, 9, true; and 25, 5, 20, 15, 10, 5, 3, true.
- Output View:** A modal window titled "Nan" displays the user's information: Nama: Mohamad Erlangga Zein, NIM: 109082500020, Kelas: S1IF-13-07.

Deskripsi program

Program ini adalah program untuk mengetahui apakah bilangan(x & y) yang diinputkan oleh user merupakan bilangan kelipatannya atau bukan, disini seperti biasa menggunakan 2 tipe data yang pertama int yang kedua Boolean, dan sama seperti pada program sebelumnya, program ini akan melakukan proses dimana user menginputkan kedua bilangan, kemudian kedua bilangan tersebut di selisihkan atau dikurangi hingga hasilnya 0, apabila hasilnya bukan 0, maka output programnya akan false/tidak memenuhi bilangan kelipatan, dan ketika dikurangi hasilnya adalah 0, maka output programnya adalah true/memenuhi bilangan tersebut adalah bilangan kelipatan, sebagai contoh disini $x=5$ dan $y=2$, maka $5-2=3$ apakah $= 0$? Tidak, maka di selisihkan lagi $3-2 = 1$ apakah $= 0$? Tidak, maka diselisihkan lagi $1-2 = -1$ apakah $= 0$? Tidak, nah disini program akan berhenti dan menampilkan output false, karena disini algoritma if nya adalah ketika $x \leq 0$, sehingga langsung dinyatakan false, berbeda dengan ketika user menginputkan $x=15$ dan $y=3$, maka ketika diproses $15-3=12$ apakah $= 0$? Tidak, selisihkan lagi $12-3=9$ apakah $= 0$? Tidak, selisihkan lagi $9-3=6$ apakah $= 0$? Tidak, selisihkan lagi $6-3=3$ apakah $= 0$? Tidak, selisihkan lagi $3-3 = 0$ apakah $= 0$? Iya, dan output programnya true.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var n, digit int
    digit = 0

    fmt.Print("Masukkan n: ")

    fmt.Scan(&n)

    for n > 0 {

        n = n / 10

        digit++

    }

    fmt.Print(digit)

}
```

Screenshot program

The screenshot shows the Visual Studio Code interface with a dark theme. In the Explorer sidebar, there are several Go files: guided1.go, guided2.go, guided3.go, soal1.go, soal2.go, and soal3.go. The soal1.go file is open in the editor, displaying the provided Go code. A terminal window is open at the bottom, showing the execution of the program. The terminal output is as follows:

```
PS C:\GoLang> go run soal1.go
Masukkan n: 5
1
PS C:\GoLang> go run soal1.go
Masukkan n: 234
3
PS C:\GoLang> go run soal1.go
Masukkan n: 78787
5
PS C:\GoLang> go run soal1.go
Masukkan n: 1894256
7
```

Deskripsi program

Program ini adalah program untuk mengetahui jumlah digit pada suatu bilangan, program ini menggunakan satu tipe data saja yaitu integer, dimana disini menggunakan konsep iterasi dengan membagi bilangan sampai hasilnya 0, untuk mempermudah, disini pembagiannya dengan angka 10, dimana angka/nilai yang di input user akan dibagi 10 dan setiap pembagian itu iterasinya akan berjumlah +1 dengan nilai dari iterasinya adalah digit = 0, misal user menginputkan nilai 5, maka $5 / 10 = 0.5$, apakah == 0? Iya, sehingga iterasi dihitung +1, dan outputnya adalah 1, sesuai dengan jumlah digit pada nilai tersebut, kemudian ketika user menginputkan nilai 234, maka akan dibagi dengan 10, $234 / 10 = 23.4$, apakah == 0? Tidak, maka dibagi lagi dengan 10, $23.4 / 10 = 2.34$ apakah == 0? Tidak, maka dibagi lagi dengan 10, $2.34 / 10 = 0.234$ apakah == 0? Ya, sehingga output programnya ada 3, karena iterasinya dilakukan sebanyak 3 kali hingga hasilnya 0.

2. Tugas 2

Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var n float64
    fmt.Print("Masukkan n: ")
    fmt.Scan(&n)

    batas := math.Ceil(n)

    saatIni := int(n * 10)
    nLanjutan := int(batas * 10)
```

```

for i := saatIni + 1; i <= nLanjutan; i++{

    if i%10 == 0 {

        fmt.Println(i / 10)

    } else {

        fmt.Printf("%.1f\n", float64(i) / 10.0)

    }

}

}

```

Screenshot program

The screenshot shows a Go development environment in VS Code. The code editor displays a file named `soal2.go` with the following content:

```

package main
import (
    "fmt"
    "math"
)
func main() {
    var n float64
    fmt.Print("Masukkan n: ")
    fmt.Scan(&n)

    batas := math.Ceil(n)

    saatIni := int(n * 10)
    nLanjutan := int(batas * 10)

    for i := saatIni + 1; i <= nLanjutan; i++{
        if i%10 == 0 {
            fmt.Println(i / 10)
        } else {
            fmt.Printf("%.1f\n", float64(i) / 10.0)
        }
    }
}

```

The terminal pane shows the command `go run soal2.go` being run, followed by the input `Masukkan n: 0.2` and the output `0.3
0.4
0.5
0.6
0.7
0.8
0.9
1`. The output pane shows the program's output for `n = 0.2`:

```

Nama: Mohamad Erlangga Zein
NIM: 109082500020
Kelas: S1IF-13-07

```

Deskripsi program

Program ini adalah program untuk mencetak nilai dari suatu bilangan, dan akan terus bertambah 1 apabila ketika di modulus 10 tidak == 0, dan perulangannya akan berhenti apabila n yang di inputkan user bertambah 1 dan ketika di modulus 10 hasilnya == 0, disini menggunakan math.Ceil() untuk membulatkan n/bilangan yang di input oleh user dibulatkan ke atas, dan untuk perulangannya, nilai dari n nya + 1 disini menggunakan variabel i(i := saatIni + 1), dan n <= n yang telah di math.ceil(n), kemudian setelah itu nilai dari n nya akan diproses menggunakan pengkondision if else, dimana ketika i%10 == 0 maka nilai i nya akan dibagi 10 (i / 10) lalu untuk kondisi else sama seperti pada if yakni i/10, untuk contohnya seperti ini, ketika user menginputkan 0.2, maka nilai 0.2 akan di kalikan dengan 10, maka untuk nilai pada saatIni adalah $0.2 \times 10 = 2$, maka nilai variabel saatIni = 2, kemudian nilai 0.2 di math.ceil sehingga akan menjadi 0.3, kemudian dari hasil itu di kali dengan 10, $0.3 \times 10 = 3$ nilai ini akan masuk ke dalam variabel nLanjutan, sehingga nilai variabel nLanjutan = 3, kemudian akan di proses pada algoritma for, dimana nilai dari variabel i := saatIni + 1; i <= nLanjutan; i++, maka perhitungannya akan seperti ini, nilai dari variabel i = 3, kemudian nilai nLanjutan = 3, apakah $i \leq nLanjutan$? Ya, kemudian akan di proses lagi dengan me moduluskam nilai dari i, ketika i%10 == 0 maka akan berhenti, dan apabila i%10 tidak sama dengan 0, maka perulangan akan terus berlanjut, nilai dari i akan terus bertambah(saatIni + 1), sampai hasil dari i%10 == 0, maka program akan berhenti dan output programnya seperti penambahan nilai secara berurutan.

3. Tugas 3

Source code

```
package main
import "fmt"

func main() {
    var target, total, donatur int
    fmt.Print("Masukkan target donasi: ")
    fmt.Scan(&target)

    total = 0
    donatur = 0

    fmt.Println("Masukkan jumlah uang untuk di
donasikan: ")

    for total <= target {
```

```

var donasi int
fmt.Scan(&donasi)
donatur += 1

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 GoLang IDE interface with several tabs open in the background. In the foreground, a terminal window displays the execution of a Go program named `soal3.go`. The program prompts the user for a target donation amount and then iterates through three donors, each contributing 100 units. The total donation reaches 300, which exceeds the target of 200, so the loop terminates. The terminal output is as follows:

```

PS C:\GoLang> go run soal3.go
Masukkan target donasi: 200
Masukkan jumlah uang untuk di donasikan:
100
Donatur 1: menyumbang 100. Total terkumpul: 100
50
Donatur 2: menyumbang 50. Total terkumpul: 150
50
Donatur 3: menyumbang 50. Total terkumpul: 200
Target tercapai! total donasi: 200 dari 200 donatur.

PS C:\GoLang> go run soal3.go
Masukkan target donasi: 500
Masukkan jumlah uang untuk di donasikan:
100
Donatur 1: menyumbang 100. Total terkumpul: 100
50
Donatur 2: menyumbang 50. Total terkumpul: 150
50
Donatur 3: menyumbang 50. Total terkumpul: 200
100
Donatur 4: menyumbang 100. Total terkumpul: 300
50
Target tercapai! total donasi: 300 dari 4 donatur.

PS C:\GoLang> go run soal3.go
Masukkan target donasi: 200

```

Deskripsi program

Program ini adalah program untuk mengumpulkan target donasi, jika jumlah donasi sudah melebihi dari target donasi, maka program akan berhenti, disini terdapat 3 variabel dengan variabel integer, dimana 3 variabel itu adalah target, total, donatur, dimana nilai default dari total dan donatur adalah 0 sebelum dilakukannya proses dari perulangannya, di dalam perulangannya dimana ketika total \leq target, maka user akan memasukkan jumlah donasi, kemudian nilai dari donaturnya +1, kemudian bila nilai dari donasi nya masih kurang, maka perulangannya tidak akan berhenti, lalu nilai dari donasi itu akan masuk ke dalam variabel total, dan variabel total itu nilainya akan

bertambah mengikuti dengan user yang menginput jumlah donasi, lalu Ketika jumlah total sudah melebihi target, maka program akan berhenti, contoh misal user menginputkan target sebanyak 500, maka user juga akan memasukkan donasinya, misal pada inputan pertama 150, maka apakah $150 \leq 500$? Ya, maka program terus berlanjut, dan iterasi dari donatur akan $+1$, kemudian user menginputkan lagi 100, maka nilai dari total nya itu akan bertambah $150 + 100 = 250$, apakah $250 \leq 500$? Ya maka program akan terus berlanjut dan nilai dari donatur menjadi 2, lalu karena masih belum memenuhi target, maka user akan memasukkan donasi lagi, contoh 50, maka total yang terkumpul = 300, apakah $300 \leq 500$? Ya, masih belum memenuhi, dan nilai iterasi dari donatur menjadi 3, kemudian user memasukkan jumlah donasi lagi karena nilai nya masih belum memenuhi 500, user memasukkan 300, maka nilai total = 600, apakah $600 \leq 500$? Tidak, nilai dari total sudah memenuhi dari nilai target, dan nilai iterasi dari donatur menjadi 4, maka output nya akan “Target tercapai! total donasi: 600 dari 4 donatur.”.