

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 13
REPEAT-UNTIL



Disusun oleh:

MUHAMMAD FAIZ MAULANA

109082500124

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1

Source Code

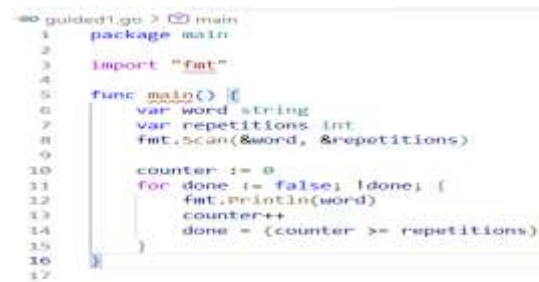
```
package main

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)
    }
}
```

Screenshoot program



```
<=0 guided1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var word string
7     var repetitions int
8     fmt.Scan(&word, &repetitions)
9
10    counter := 0
11    for done := false; !done; {
12        fmt.Println(word)
13        counter++
14        done = (counter >= repetitions)
15    }
16 }
17
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
pagi 3
pagi
pagi
pagi
PS C:\Users\User-PC\Documents\Semester 1\apro\khusus_laprak\mingguli3\guided> go run "C:\Users\User-PC\Documents\Semester 1\apro\khusus_laprak\mingguli3\guided\guided1.go"
pagi 5
pagi
pagi
pagi
pagi
```

Deskripsi program

Program Go ini membaca dua input dari pengguna: sebuah kata (string) dan sebuah angka repetitions (int) lewat `fmt.Scan(&word, &repetitions)`, lalu mencetak kata tersebut berulang-ulang sebanyak nilai repetitions. Caranya, program membuat penghitung counter mulai dari 0, kemudian menjalankan perulangan for dengan kondisi berhenti yang diatur oleh variabel boolean done. Di setiap iterasi, program menampilkan word dengan `fmt.Println(word)`, menaikkan counter, lalu, mengubah done menjadi true kalau counter \geq repetitions, sehingga perulangan berhenti tepat setelah jumlah cetakkannya sesuai angka yang dimasukkan.

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)
}
```

Screenshoot program



Deskripsi program

Kode Go ini meminta pengguna memasukkan sebuah angka terus-menerus di dalam perulangan sampai pengguna memberikan bilangan bulat positif. Setiap kali input dibaca, program mengecek apakah angkanya masih kurang dari atau sama dengan nol; kalau iya, perulangan lanjut (minta input lagi), dan kalau sudah lebih dari nol, perulangan berhenti. Setelah mendapatkan angka yang valid, program menampilkan pesan bahwa angka tersebut adalah bilangan bulat positif dengan format %d.

3. Guided 3

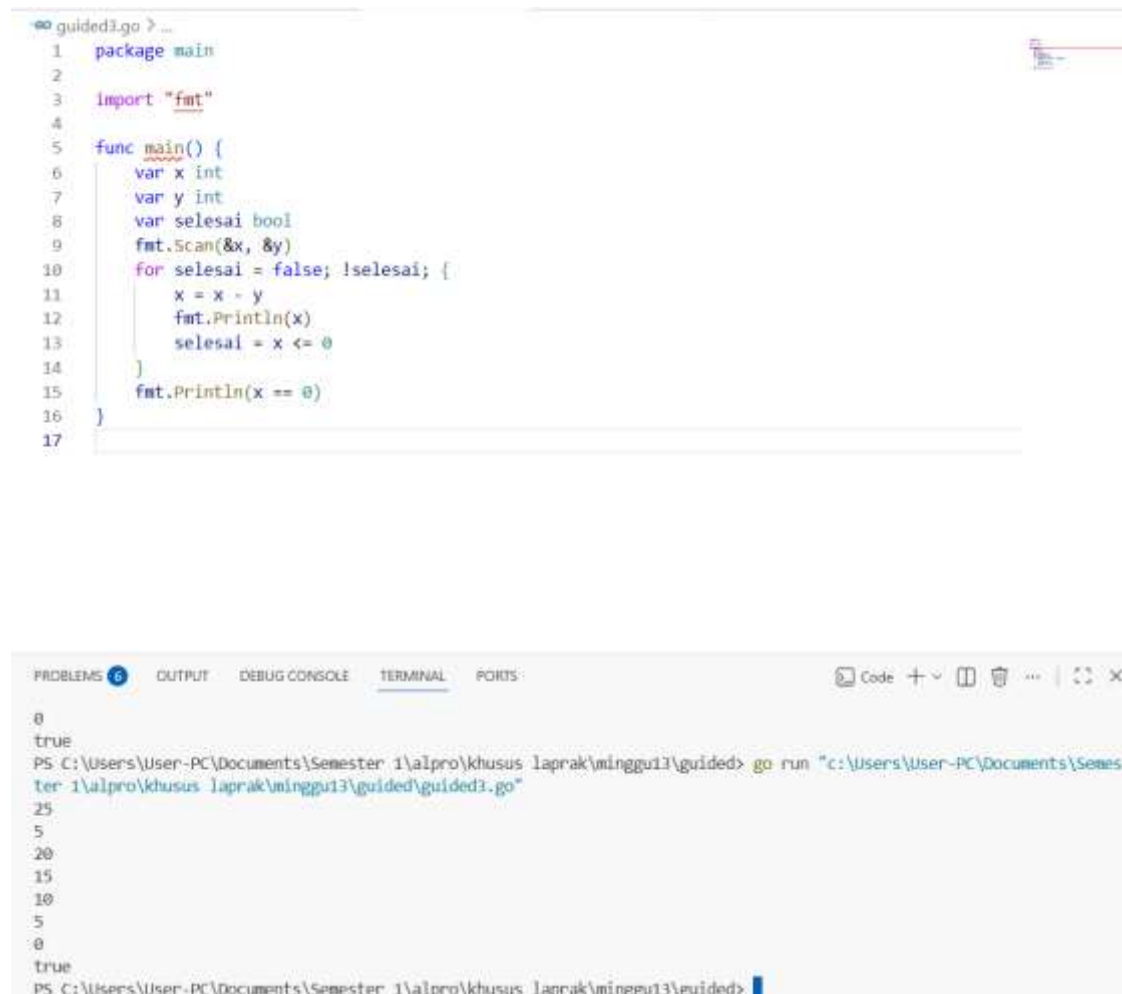
Source Code

```
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 image shows two screenshots from a code editor. The top screenshot displays the source code of a Go program named `guided3.go`. The code defines a `main` function that declares variables `x` (int), `y` (int), and `selesai` (bool). It uses `fmt.Scan(&x, &y)` to read input. A `for` loop runs while `selesai` is false, decrementing `x` by `y` and printing `x` until `x` is less than or equal to 0. Finally, it prints `x == 0`. The bottom screenshot shows the terminal output of running the program. It shows the input values 0 and 25, followed by the output of the loop (0, 5, 10, 15, 20) and the final result `true` because `x` reached 0.

```
guided3.go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      var x int
7      var y int
8      var selesai bool
9      fmt.Scan(&x, &y)
10     for selesai = false; !selesai; {
11         x = x - y
12         fmt.Println(x)
13         selesai = x <= 0
14     }
15     fmt.Println(x == 0)
16 }
17
```

```
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS
0
true
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus laprak\minggui3\guided> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus laprak\minggui3\guided\guided3.go"
25
5
20
15
10
5
0
true
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus laprak\minggui3\guided>
```

Deskripsi program

Kode Go ini membaca dua bilangan bulat `x` dan `y`, lalu menjalankan perulangan yang terus mengurangi nilai `x` dengan `y` (jadi `x = x - y`) sambil mencetak nilai `x` setiap kali pengurangan dilakukan. Perulangan berhenti ketika `x` sudah kurang dari atau sama dengan 0 (ditandai dengan variabel boolean `selesai` menjadi `true`). Setelah keluar dari loop, program mencetak hasil pengecekan `x == 0`, jadi output akhirnya adalah `true` kalau `x` pas menjadi 0 (artinya habis dibagi/pengurangan berulangannya tepat), dan `false` kalau `x` lewat menjadi negatif.

TUGAS

1. Tugas 1

Source code

```
package main

import ("fmt")

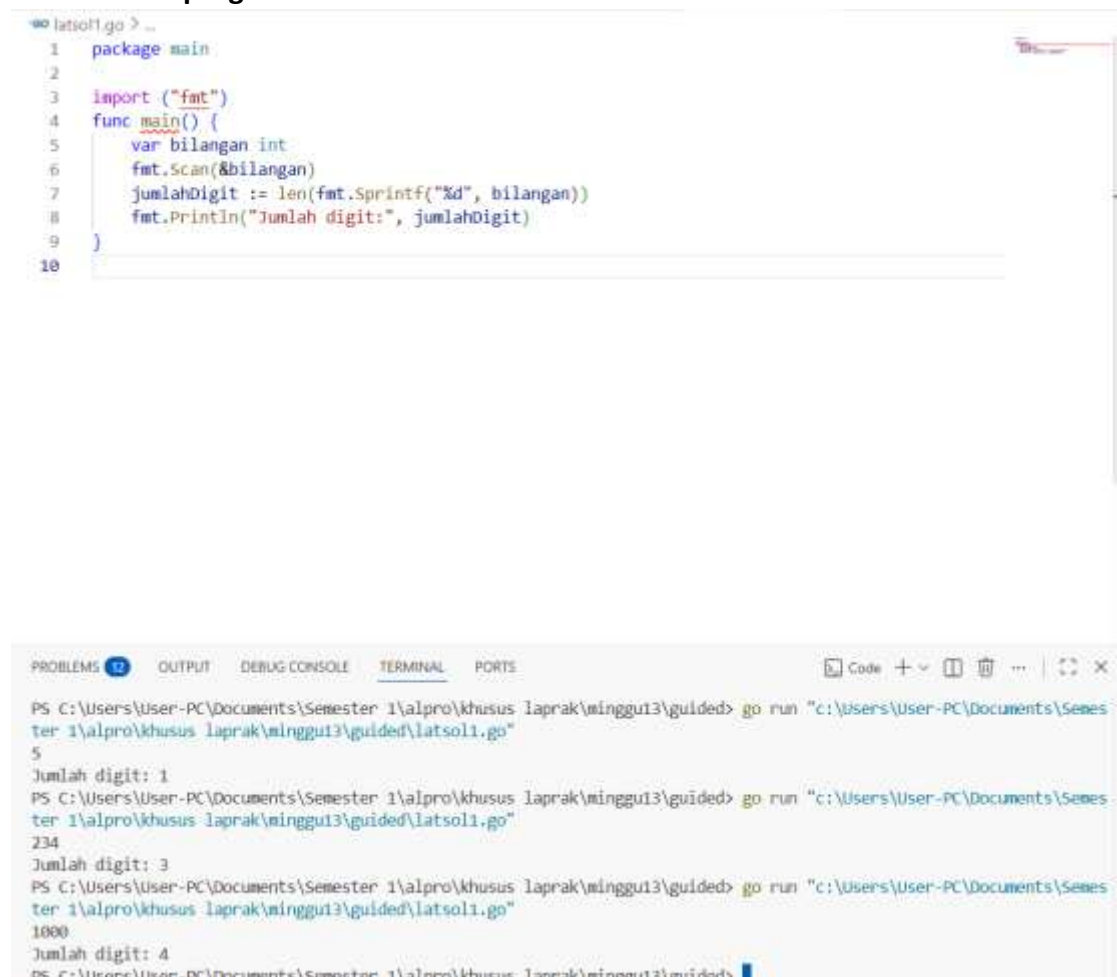
func main() {
    var bilangan int

    fmt.Scan(&bilangan)

    jumlahDigit := len(fmt.Sprintf("%d", bilangan))

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

Screenshoot program



Deskripsi program

Program Go ini membaca satu bilangan bulat dari input, lalu menghitung berapa banyak digit angka tersebut dengan cara mengubahnya dulu menjadi teks (string). Setelah bilangan diubah menjadi string memakai format angka bulat, program mengambil panjang teks itu (jumlah karakternya) sebagai jumlah digit. Terakhir, hasilnya ditampilkan ke layar dengan tulisan "Jumlah digit:" diikuti angka jumlah digitnya.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var x float64
    fmt.Scan(&x)
    batasAtas := float64(int(x)) + 1
    for v := x + 0.1; v <= batasAtas; v += 0.1 {
        fmt.Printf("%.1f\n", v)
    }
}
```

Screenshoot program



Deskripsi program

Kode Go ini membaca sebuah angka desimal x (tipe float64), lalu menentukan batas atas batasAtas sebagai bilangan bulat dari x ditambah 1 (misalnya jika x = 2.7, maka int(x)=2 dan batasAtas=3.0). Setelah itu program melakukan perulangan dari v = x + 0.1 sampai v <= batasAtas dengan kenaikan 0.1 setiap langkah, dan setiap nilai v dicetak dengan satu angka di belakang koma menggunakan format %.1f. Jadi, outputnya adalah deretan angka yang naik 0.1 mulai sedikit di atas x sampai mendekati/tepat di batas atas tersebut.

3. Tugas 3

Source code

```
package main

import "fmt"

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

    total := 0
    donatur := 1

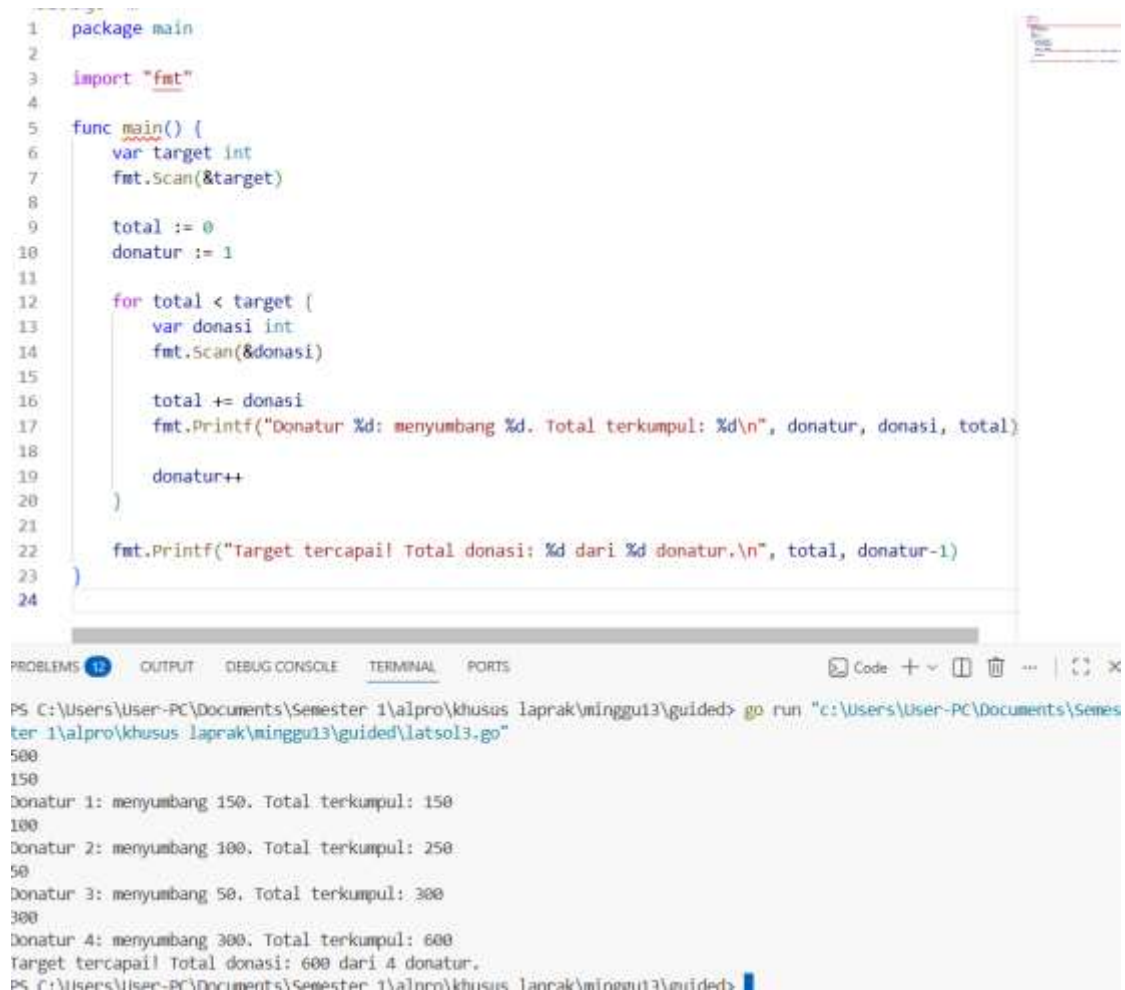
    for total < target {
        var donasi int
        fmt.Scan(&donasi)

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

        donatur++
    }

    fmt.Printf("Target tercapai! Total donasi: %d dari %d donatur.\n", total, donatur-1)
}
```

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var target int
7     fmt.Scan(&target)
8
9     total := 0
10    donatur := 1
11
12    for total < target {
13        var donasi int
14        fmt.Scan(&donasi)
15
16        total += donasi
17        fmt.Printf("Donatur %d: menyumbang %d. Total terkumpul: %d\n", donatur, donasi, total)
18
19        donatur++
20    }
21
22    fmt.Printf("Target tercapai! Total donasi: %d dari %d donatur.\n", total, donatur-1)
23 }
24
```

PROBLEMS 12 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus laprak\minggu13\guided> go run "c:\Users\User-PC\Documents\Semester 1\alpro\khusus laprak\minggu13\guided\latsol3.go"
500
150
Donatur 1: menyumbang 150. Total terkumpul: 150
100
Donatur 2: menyumbang 100. Total terkumpul: 250
50
Donatur 3: menyumbang 50. Total terkumpul: 300
300
Donatur 4: menyumbang 300. Total terkumpul: 600
Target tercapai! Total donasi: 600 dari 4 donatur.
PS C:\Users\User-PC\Documents\Semester 1\alpro\khusus laprak\minggu13\guided>
```

Deskripsi program

Program Go ini meminta pengguna memasukkan angka target donasi, lalu menerima input donasi satu per satu dari para donatur sampai total donasi yang terkumpul mencapai atau melewati target. Setiap kali ada donasi masuk, program menambahkan nilainya ke total dan langsung menampilkan laporan: donatur ke berapa, menyumbang berapa, dan total sementara yang sudah terkumpul. Setelah target tercapai, perulangan berhenti dan program menampilkan pesan bahwa target sudah terpenuhi, beserta total donasi akhir dan jumlah donatur yang berpartisipasi