

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 13
REPEAT-UNTIL**



Disusun oleh:

Farrasya Lazuardi

109082500040

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.Print("Masukkan Kata dan Jumlah Kata: ")

    fmt.Scan(&kata, &jumlahKata)

    for kondisi = false; !kondisi; {

        fmt.Println(kata)

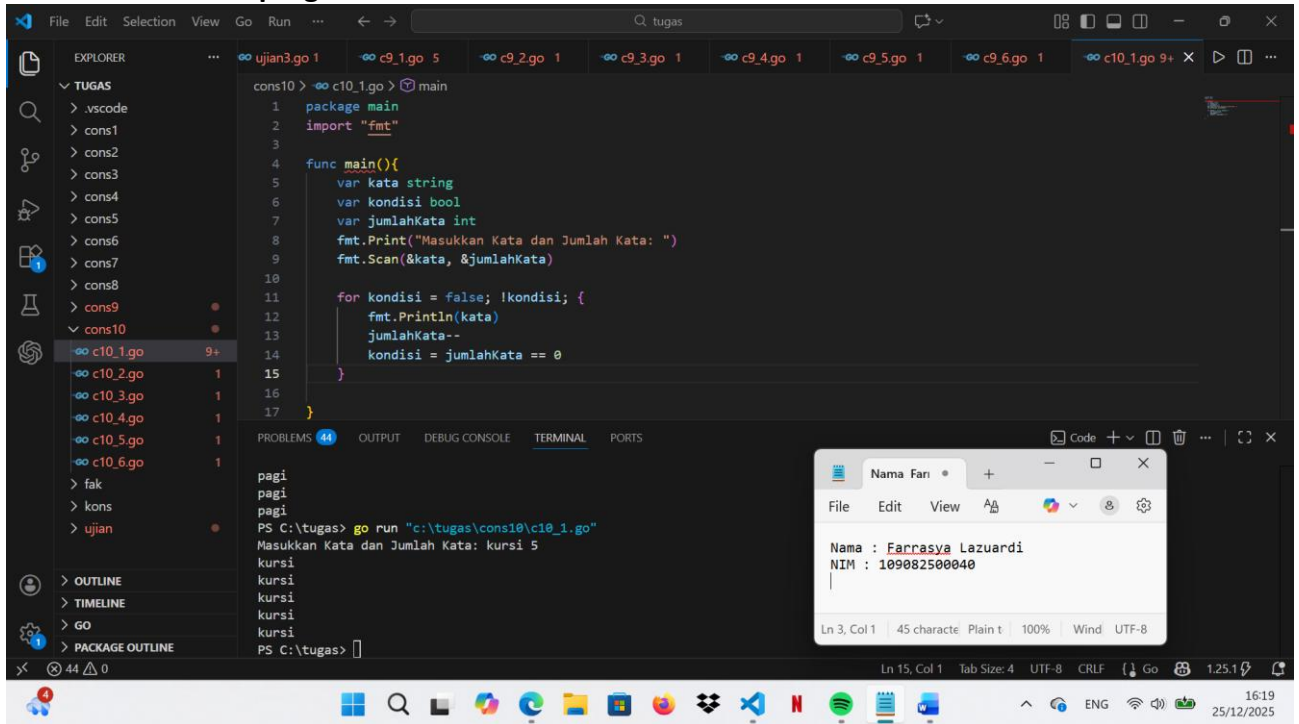
        jumlahKata--

        kondisi = jumlahKata == 0

    }

}
```

Screenshoot program



Deskripsi program

Program ini berfungsi untuk menampilkan sebuah kata secara berulang-ulang sesuai dengan jumlah yang diinginkan pengguna. Program meminta dua input: sebuah kata (string) dan jumlah perulangan (integer). Menggunakan struktur perulangan (loop), program akan mencetak kata tersebut baris demi baris dan mengurangi nilai penghitung (jumlahKata) satu per satu hingga mencapai nol, di mana perulangan akan berhenti.

2. Guided 2

Source Code

```
package main

import "fmt"

func main(){

    var a int

    var kondisi bool
```

```

        for kondisi = false; !kondisi;{

            fmt.Print("Masukkan Angka: ")

            fmt.Scan(&a)

            kondisi = (a * -2) < 0

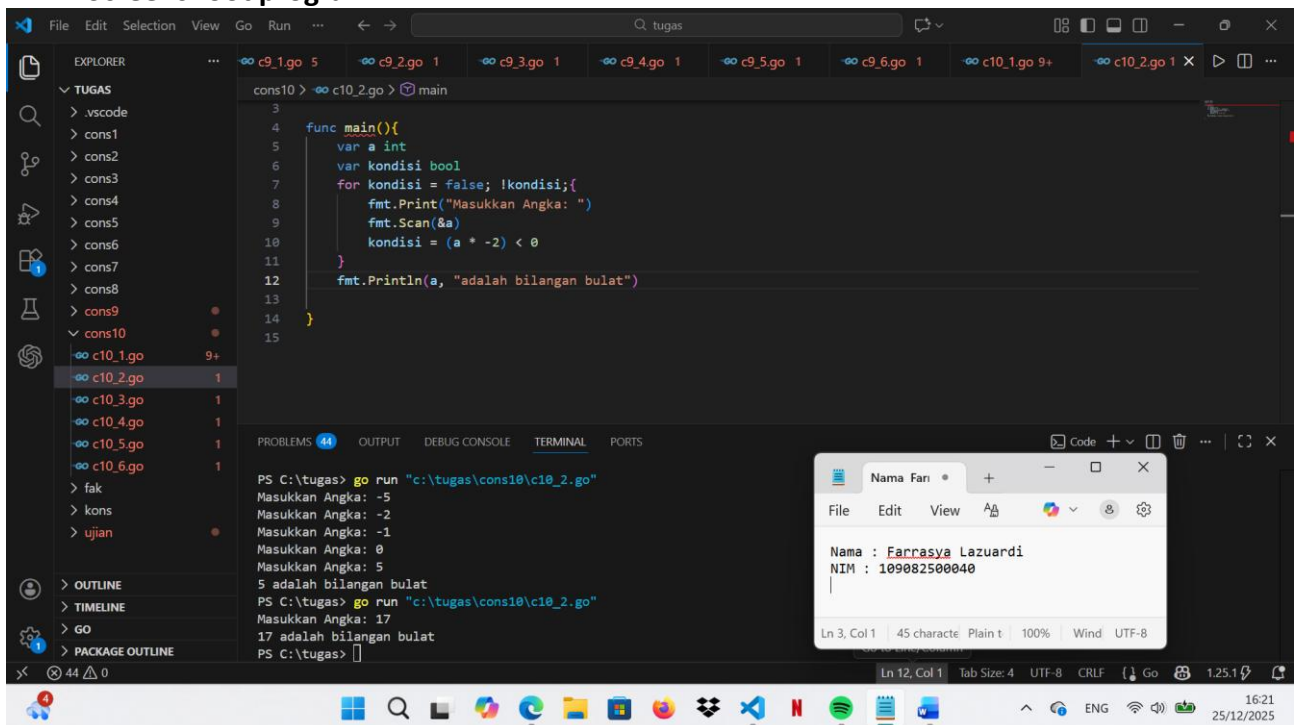
        }

        fmt.Println(a, "adalah bilangan bulat")

    }

```

Screenshoot program



Deskripsi program

Program ini untuk memvalidasi input agar hanya menerima bilangan bulat positif untuk menghentikan program. Program akan terus meminta pengguna memasukkan angka selama kondisi $(a * -2) < 0$ bernilai *false*. Secara matematis, kondisi ini memaksa pengguna memasukkan angka positif (karena positif dikali negatif hasilnya negatif, yang kurang dari 0), sehingga jika pengguna memasukkan angka negatif atau nol, program akan terus melakukan *looping*.

3. Guided 3

Source Code

```
package main

import "fmt"

func main(){

    var a, b int

    fmt.Scan(&a)

    fmt.Scan(&b)


    for kondisi := false; !kondisi; {

        a = a - b

        fmt.Println(a)


        if a <= 0{

            kondisi = a == 0

            fmt.Print(kondisi)

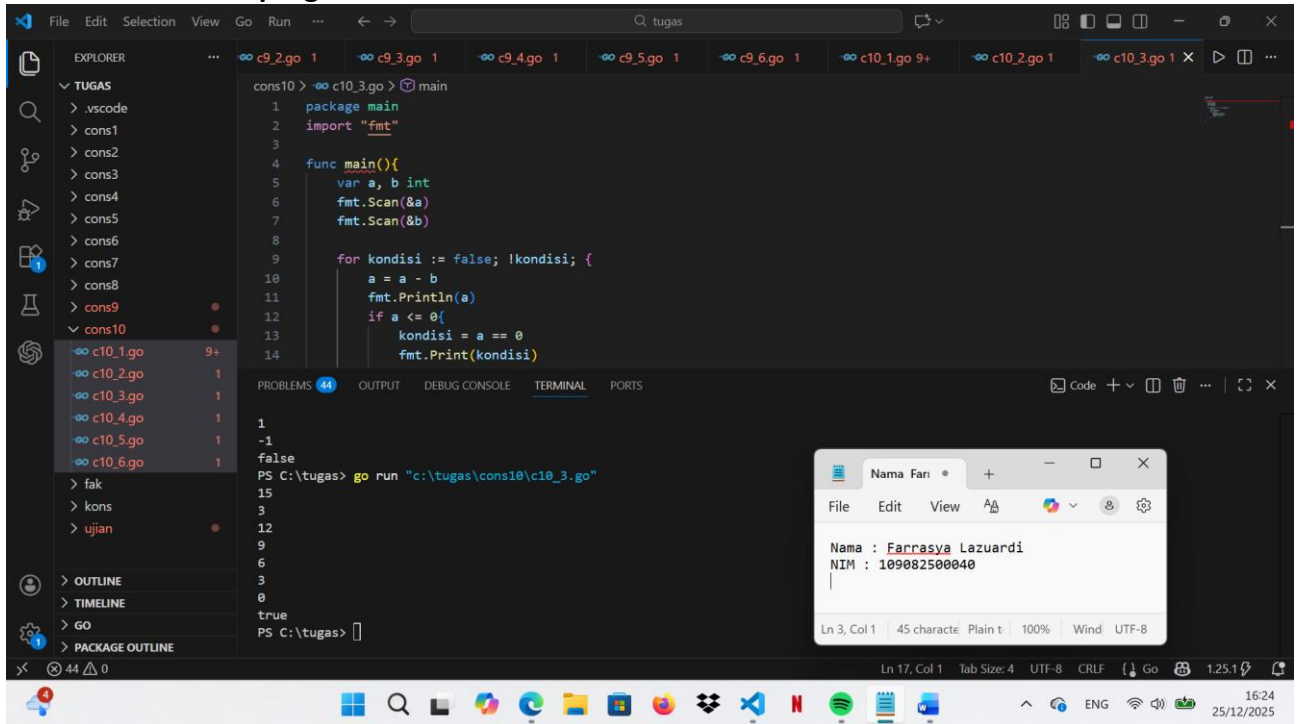

            kondisi = true

        }

    }

}
```

Screenshoot program



The screenshot shows the Visual Studio Code interface with a Go program open in the editor. The Explorer panel on the left shows a project structure with a 'TUGAS' folder containing several Go files. The main editor displays the code for 'c10_3.go', which implements a loop to calculate the remainder of a division. The terminal at the bottom shows the command 'go run "c:\tugas\cons10\c10_3.go"' being executed, and the output shows the program's execution results, including the input values and the final result.

```
cons10 > c10_3.go > main
1 package main
2 import "fmt"
3
4 func main(){
5     var a, b int
6     fmt.Scan(&a)
7     fmt.Scan(&b)
8
9     for kondisi := false; !kondisi; {
10        a = a - b
11        fmt.Println(a)
12        if a <= 0{
13            kondisi = a == 0
14            fmt.Print(kondisi)
15        }
16    }
17 }
```

Terminal Output:

```
PS C:\tugas> go run "c:\tugas\cons10\c10_3.go"
1
-1
false
15
3
12
9
6
3
0
true
PS C:\tugas>
```

Deskripsi program

Program ini mensimulasikan operasi pembagian atau pencarian sisa bagi (modulo) menggunakan metode pengurangan berulang. Program menerima dua angka input, lalu angka pertama dikurangi dengan angka kedua secara terus-menerus dalam sebuah loop hingga hasilnya menjadi nol atau negatif. Pada akhirnya, program menampilkan nilai true jika hasil akhirnya tepat nol (artinya habis dibagi) atau false jika hasilnya negatif (artinya memiliki sisa bagi).

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var bilangan int

    fmt.Print("Masukkan bilangan: ")

    fmt.Scan(&bilangan)

    jumlahdigit := 0

    for kondisi := false; !kondisi; {

        bilangan = bilangan / 10

        jumlahdigit++

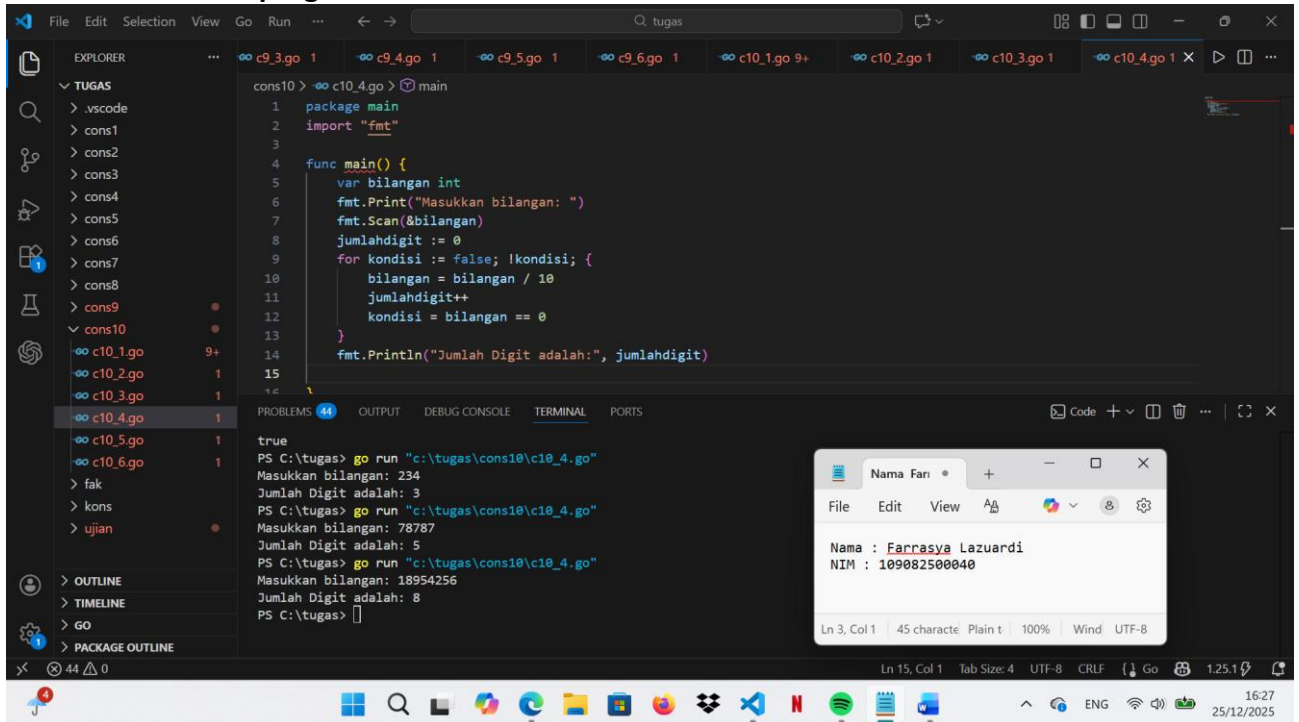
        kondisi = bilangan == 0

    }

    fmt.Println("Jumlah Digit adalah:", jumlahdigit)

}
```

Screenshoot program



Deskripsi program

Program ini untuk menghitung jumlah digit yang terdapat dalam sebuah bilangan bulat yang diinputkan pengguna. Logika utamanya adalah menggunakan perulangan untuk membagi bilangan input dengan angka 10 secara terus-menerus. Setiap kali pembagian terjadi (yang secara efektif membuang digit terakhir), variabel penghitung (jumlahdigit) akan bertambah satu hingga bilangan tersebut habis (menjadi 0).

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var n float64

    fmt.Print("Masukkan sebuah bilangan desimal: ")

    fmt.Scan(&n)
```

```

bilBul := int(n)

target := bilBul

if n != float64(bilBul) {

    target = bilBul + 1

}

for kondisi:= false; !kondisi; {

    n = n + 0.1

    fmt.Printf("%.1f\n", n)

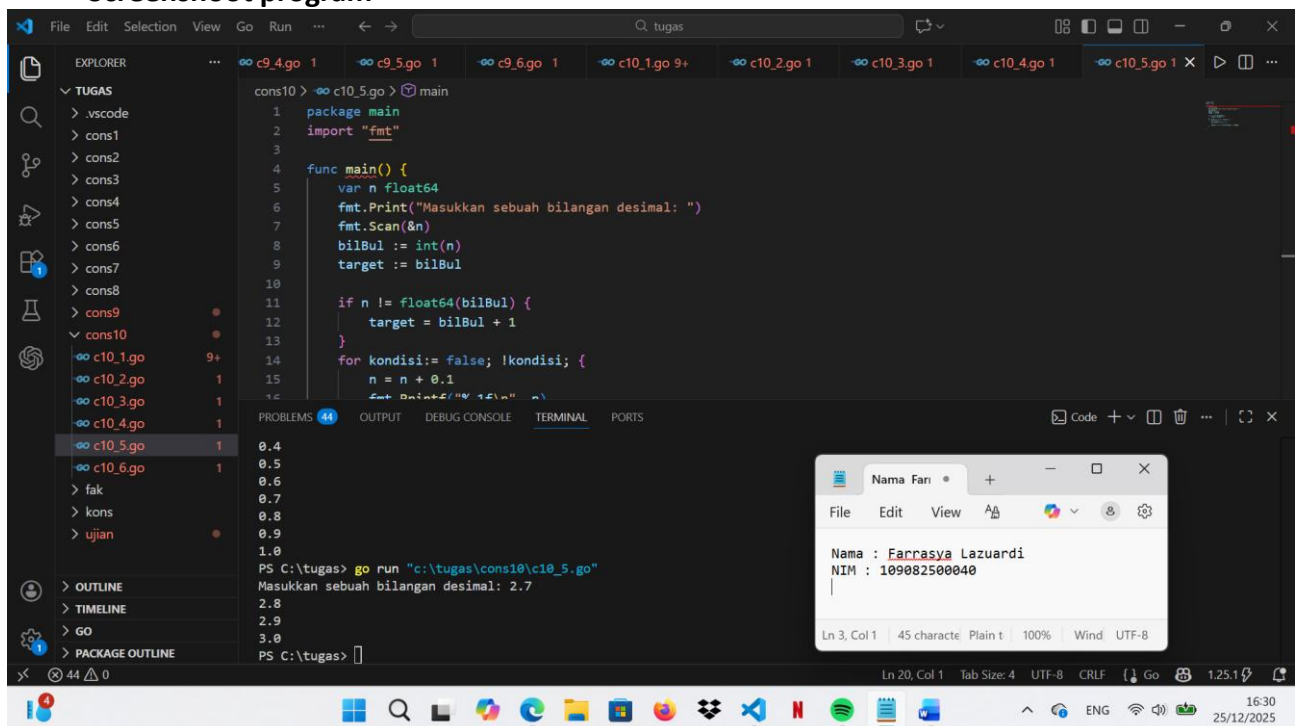
    kondisi = n >= float64(target) - 0.000001

}

}

```

Screenshoot program



Deskripsi program

Program ini buat melakukan pembulatan ke atas (ceiling) dari sebuah bilangan desimal secara manual dengan iterasi penambahan 0.1. Setelah pengguna memasukkan bilangan desimal (float), program menentukan target bilangan bulat terdekat di atasnya. Kemudian, program melakukan *loop* untuk menambahkan 0.1 pada bilangan awal dan mencetak hasilnya langkah demi langkah hingga nilainya mencapai target bilangan bulat tersebut.

3. Tugas 3

Source code

```
package main

import "fmt"

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

    jumlah = 0
    donatur := 0

    for kondisi := false; !kondisi; {
        fmt.Print("Masukkan donasi: ")
        fmt.Scan(&donasi)

        jumlah += donasi
        donatur++

        fmt.Printf("Donatur %d: Menyumbang %d. Total  
Terkumpul: %d\n", donatur, donasi, jumlah)
        kondisi = jumlah >= target
    }
    fmt.Printf("Target tercapai total donasi: %d dari  
%d donatur\n", jumlah, donatur)
}
```

Screenshoot program

The screenshot shows a Go program in VS Code. The Explorer panel on the left shows a project named 'TUGAS' with files 'cons1' through 'cons10'. The main editor shows the code for 'cons10_6.go'. The code defines a 'main' function that takes 'target', 'jumlah', and 'donasi' as integers. It uses 'fmt' for printing and scanning. The program prompts the user for a target donation, then enters a loop where it repeatedly asks for donation amounts from donors until the total reaches or exceeds the target. The terminal output shows the execution of the program with three donors contributing 100, 50, and 200 respectively, reaching a total of 350.

```
cons10 > go run c10_6.go
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var target, jumlah, donasi int
9     fmt.Print("Masukkan target donasi: ")
10    fmt.Scan(&target)
11    jumlah = 0
12    donatur := 0
13
14    for kondisi := false; !kondisi; {
15        fmt.Print("Masukkan donasi: ")
16        fmt.Scan(&donasi)
17        jumlah += donasi
18        donatur++
19    }
20
21    fmt.Println("Target tercapai total donasi: ", jumlah, "dari", donatur, "donatur")
22}
```

Terminal Output:

```
PS C:\tugas> go run "c:\tugas\cons10\c10_6.go"
Masukkan target donasi: 300
Masukkan donasi: 100
Donatur 1: Menyumbang 100. Total Terkumpul: 100
Masukkan donasi: 50
Donatur 2: Menyumbang 50. Total Terkumpul: 150
Masukkan donasi: 200
Donatur 3: Menyumbang 200. Total Terkumpul: 350
Target tercapai total donasi: 350 dari 3 donatur
PS C:\tugas>
```

Deskripsi program

Program ini buat sistem pencatatan donasi sederhana untuk mencapai target dana tertentu. Pengguna diminta memasukkan target total donasi, kemudian program akan masuk ke dalam perulangan untuk meminta input nominal donasi dari para donatur satu per satu. Loop akan terus berjalan dan mengakumulasi jumlah dana serta jumlah donatur hingga total uang yang terkumpul setara atau melebihi target yang ditetapkan.