

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 13  
REPEAT-UNTIL**



**Disusun oleh:**

**FIRDAUS RAMADHANA**

**109082500151**

**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 teks string

    var cetak int

    fmt.Print("Masukkan teks dan berapakai akan dicetak: ")

    fmt.Scanln(&teks, &cetak)

    for i := 0; i < cetak; i++ {

        fmt.Println(teks)

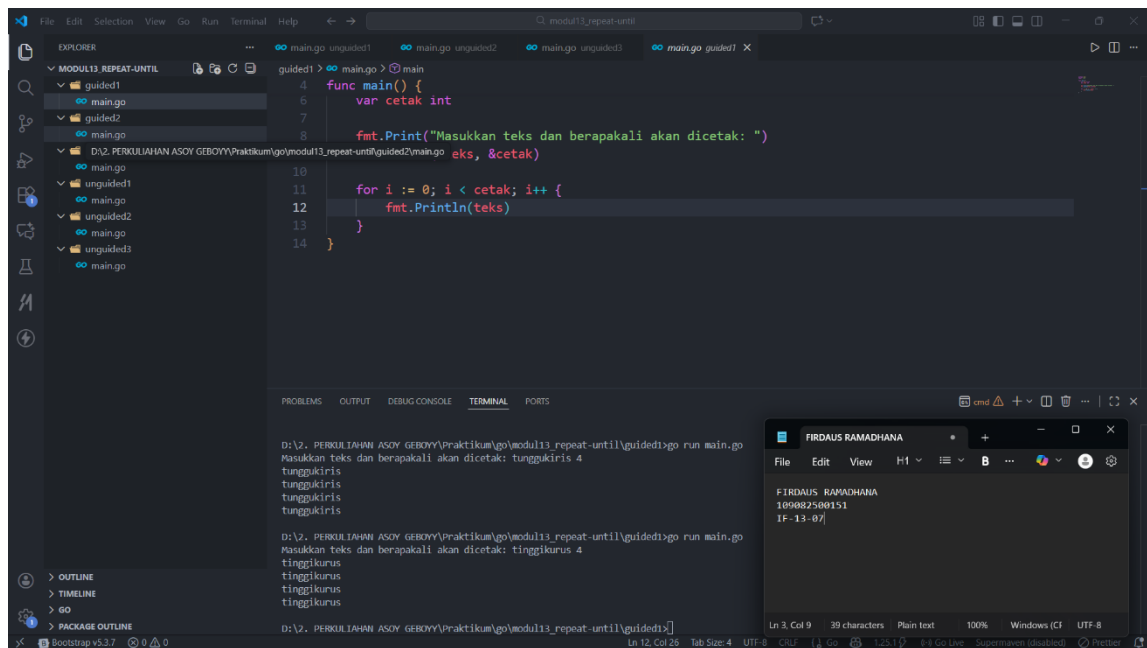
    }

}
```

#### Screenshoot program

//tambahkan tangkapan layar dari program (boleh lebih dari 1 jika diperlukan)

#### CONTOH TANGKAPAN LAYAR:



### Deskripsi program

Program ini meminta kita memasukkan sebuah teks dan sebuah angka yang menunjukkan berapa kali teks tersebut akan dicetak, kemudian program menyimpan teks ke dalam variabel teks dan jumlah pengulangan ke dalam variabel cetak, setelah itu program menggunakan perulangan for untuk menampilkan teks yang dimasukkan oleh kita ke layar secara berulang-ulang sesuai dengan jumlah cetak yang diminta.

## 2. Guided 2

### Source Code

```
package main

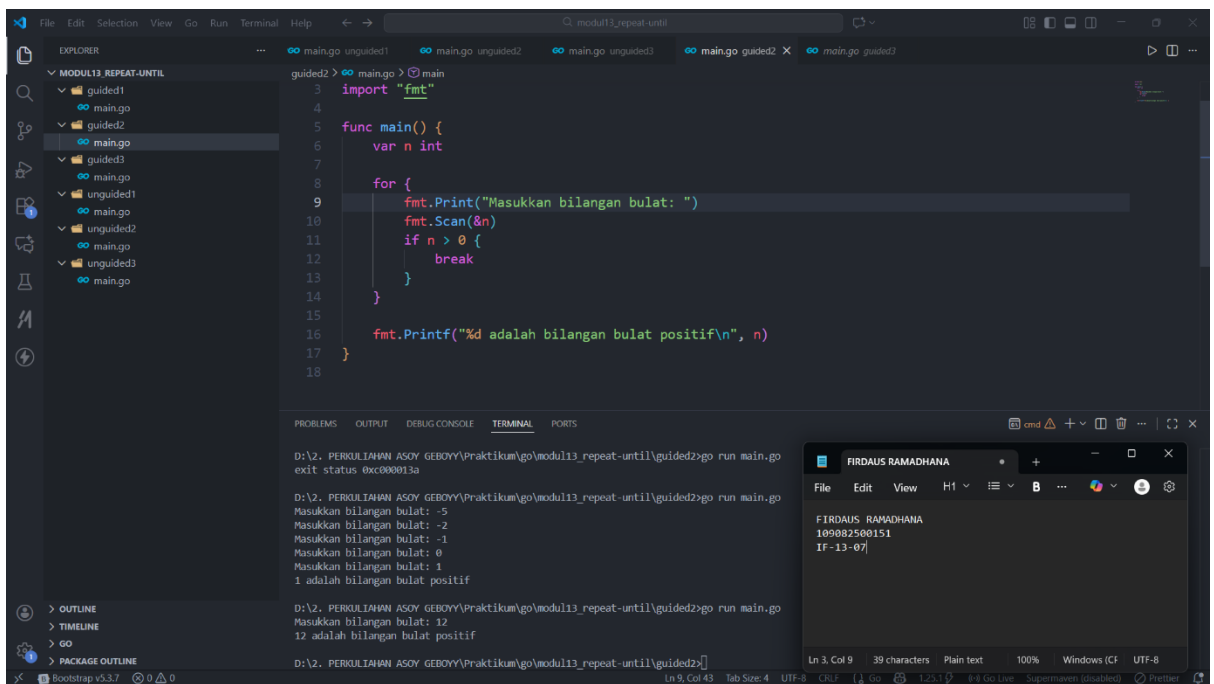
import "fmt"

func main() {
    var n int

    for {
        fmt.Print("Masukkan bilangan bulat: ")
        fmt.Scan(&n)
        if n > 0 {
            break
        }
    }

    fmt.Printf("%d adalah bilangan bulat positif\n", n)
}
```

## Screenshoot program



## Deskripsi program

Program ini bekerja dengan cara terus meminta kita memasukkan sebuah bilangan bulat, kemudian mengecek setiap input yang diberikan, dan selama bilangan tersebut belum bernilai positif (lebih dari nol) program akan terus mengulang permintaan input, hingga akhirnya ketika kita memasukkan bilangan bulat positif, program berhenti dan menampilkan pesan bahwa bilangan tersebut adalah bilangan bulat positif.

### 3. Guided 3

#### Source Code

```
package main

import "fmt"

func main() {

    var x, y int

    var kondisi bool

    fmt.Print("Masukkan dua angka: ")

    fmt.Scanln(&x, &y)

    for kondisi = false; !kondisi; {
```

```

        x -= y

        fmt.Println(x)

        if x <= 0 {

            kondisi = x == 0

            fmt.Println(kondisi)

            kondisi = true

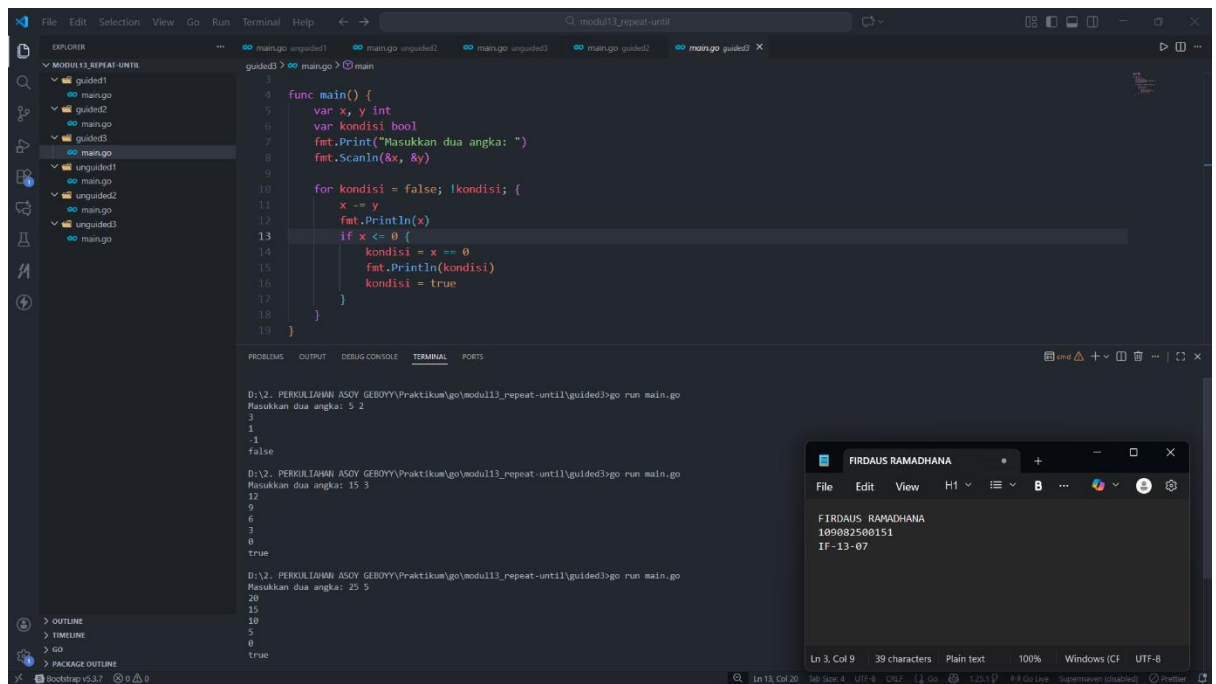
        }

    }

}

```

## Screenshoot program



## Deskripsi program

Program ini bekerja dengan cara meminta kita memasukkan dua bilangan bulat, yaitu x dan y, kemudian program mengurangi nilai x dengan y secara berulang-ulang menggunakan perulangan for, di mana setiap hasil pengurangan langsung ditampilkan ke layar, dan perulangan tersebut akan terus berjalan sampai nilai x menjadi kurang dari atau sama dengan nol, lalu program menampilkan nilai kondisi (true) dan menghentikan perulangan.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {

    var n int

    fmt.Print("Masukkan angka: ")

    fmt.Scan(&n)

    count := 0

    for n > 0 {

        n /= 10

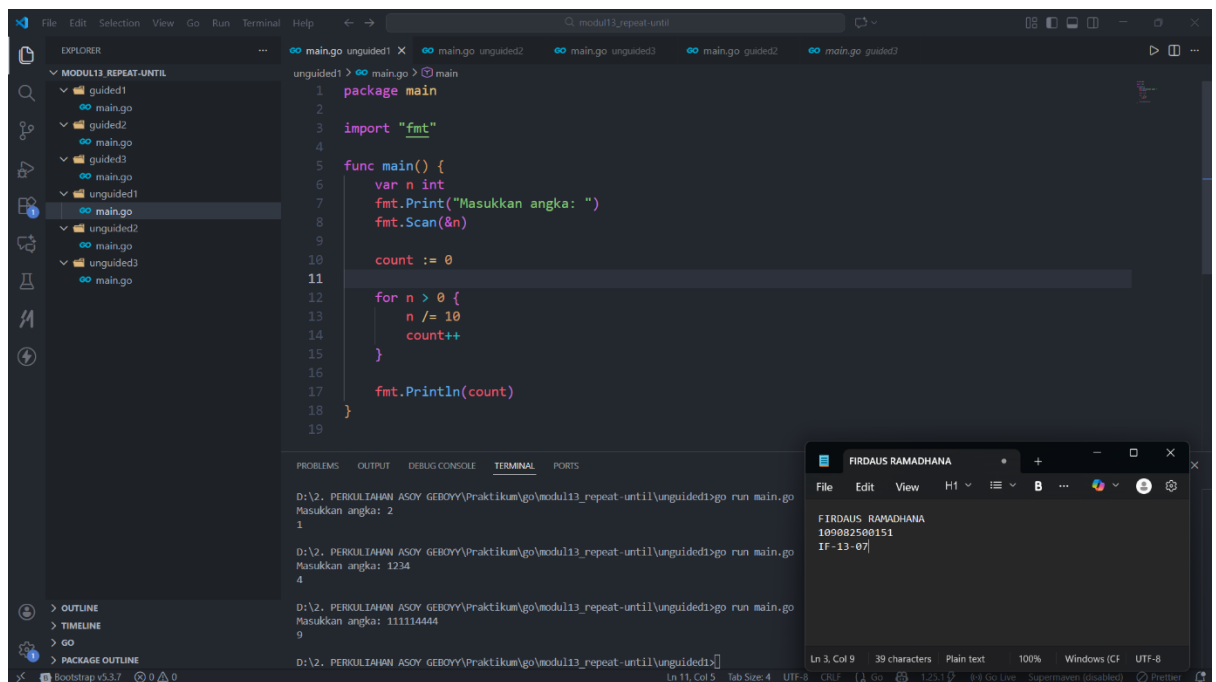
        count++

    }

    fmt.Println(count)

}
```

#### Screenshoot program



## Deskripsi program

Program ini bekerja dengan cara meminta kita memasukkan sebuah angka bulat, kemudian program menyimpan angka tersebut ke dalam variabel `n`, setelah itu program menggunakan perulangan `for` untuk membagi angka tersebut dengan 10 secara berulang-ulang, di mana setiap pembagian menghilangkan satu digit dari belakang angka, sambil menambah nilai variabel `count`, sehingga pada akhirnya nilai `count` menunjukkan banyaknya digit dari angka yang kita masukkan dan hasil tersebut ditampilkan ke layar.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {

    var a float64

    fmt.Print("Masukkan angka desimal: ")

    fmt.Scan(&a)
```

```

b := int(a * 10)

limit := ((b / 10) + 1) * 10

for b < limit {

    b++

    fmt.Printf("%.1f\n", float64(b)/10)

}

```

## Screenshoot program

The screenshot shows a Go program in VS Code. The source code is as follows:

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var a float64
7     fmt.Print("Masukkan angka desimal: ")
8     fmt.Scan(&a)
9
10    b := int(a * 10)
11    limit := ((b / 10) + 1) * 10
12
13    for b < limit {
14        b++
15        fmt.Printf("%.1f\n", float64(b)/10)
16    }
17 }
18

```

The terminal output shows the program running twice. In the first run, the input is 0.2, and the output is a sequence of numbers from 0.3 to 1.0. In the second run, the input is 2.6, and the output is a sequence of numbers from 2.7 to 3.0.

```

D:\2. PERKULIAHAN ASOY GERONY\Praktikum\go\modul13_repeat-until\unguided2>go run main.go
Masukkan angka desimal: 0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0

D:\2. PERKULIAHAN ASOY GERONY\Praktikum\go\modul13_repeat-until\unguided2>go run main.go
Masukkan angka desimal: 2.6
2.7
2.8
2.9
3.0

```

## Deskripsi program

Program ini bekerja dengan cara meminta kita memasukkan sebuah angka desimal, kemudian program mengalikan angka tersebut dengan 10 dan mengubahnya menjadi bilangan bulat untuk menghindari kesalahan desimal, menentukan batas pembulatan ke atas, lalu menggunakan perulangan untuk menambahkan nilai tersebut sedikit



demi sedikit sambil menampilkan hasilnya hingga mencapai nilai pembulatan ke atas dari angka yang kita masukkan.

### 3. Tugas 3

#### Source code

```
package main

import "fmt"

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

    total := 0
    donatur := 0

    for total <= target {
        var donasi int
        fmt.Print("Masukkan jumlah donasi: ")
        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 an IDE with a Go file named `modul13_repeat-until.go`. The code implements a loop that asks for a target donation and then repeatedly asks for individual donations until the total reaches or exceeds the target. The terminal output shows two runs of the program. In the first run, the target is 200, and four donations of 50, 50, 50, and 50 are entered, reaching the target. In the second run, the target is 100, and three donations of 30, 40, and 30 are entered, reaching the target.

```

1  func main() {
2      fmt.Println("Masukkan target donasi: ")
3      fmt.Scan(&target)
4
5      total := 0
6      donatur := 0
7
8      for total <= target {
9          var donasi int
10         fmt.Println("Masukkan jumlah donasi: ")
11         fmt.Scan(&donasi)
12
13         donatur++
14         total += donasi
15
16         fmt.Printf(
17             "Donatur %d: Menyumbang %d. Total terkumpul: %d\n",
18             donatur, donasi, total,
19         )
20
21         fmt.Printf(
22             "Target tercapai! Total donasi: %d dari %d donatur.\n",
23             total, donatur,
24         )
25     }
26 }

```

```

D:\J. PERULIAHWA ASOY\GEOHY\Praktikum\go\modul13_repeat-until\modul13_repeat-until.go run main.go
Masukkan target donasi: 200
Masukkan jumlah donasi: 50
Donatur 1: Menyumbang 50. Total terkumpul: 50
Masukkan jumlah donasi: 50
Donatur 2: Menyumbang 50. Total terkumpul: 100
Masukkan jumlah donasi: 50
Donatur 3: Menyumbang 50. Total terkumpul: 150
Masukkan jumlah donasi: 50
Donatur 4: Menyumbang 50. Total terkumpul: 200
Target tercapai! Total donasi: 200 dari 4 donatur.

D:\J. PERULIAHWA ASOY\GEOHY\Praktikum\go\modul13_repeat-until\modul13_repeat-until.go run main.go
Masukkan target donasi: 100
Masukkan jumlah donasi: 30
Donatur 1: Menyumbang 30. Total terkumpul: 30
Masukkan jumlah donasi: 40
Donatur 2: Menyumbang 40. Total terkumpul: 70
Masukkan jumlah donasi: 30
Donatur 3: Menyumbang 30. Total terkumpul: 100
Target tercapai! Total donasi: 100 dari 3 donatur.

```

## Deskripsi program

Program ini bekerja dengan cara meminta kita memasukkan target donasi, kemudian menerima donasi satu per satu dari kita, menambahkan setiap donasi ke total dan menghitung jumlah donatur, menampilkan perkembangan total donasi setiap kali perulangan, dan terus mengulang proses tersebut hingga total donasi mencapai atau melebihi target, lalu menampilkan pesan bahwa target donasi telah tercapai beserta total donasi dan jumlah donatur.