

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 13  
REPEAT-UNTIL**



**Disusun oleh:**

**BAGUS IRSYAD KAMAL**

**109082500215**

**S1IF-13-02**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**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("Masukan kata :")

    fmt.Scan(&word)

    fmt.Print("Masukan berapa kali pengulangan :")

    fmt.Scan(&repetitions)


    counter := 0

    for done := false; !done; {

        fmt.Println(word)

        counter++

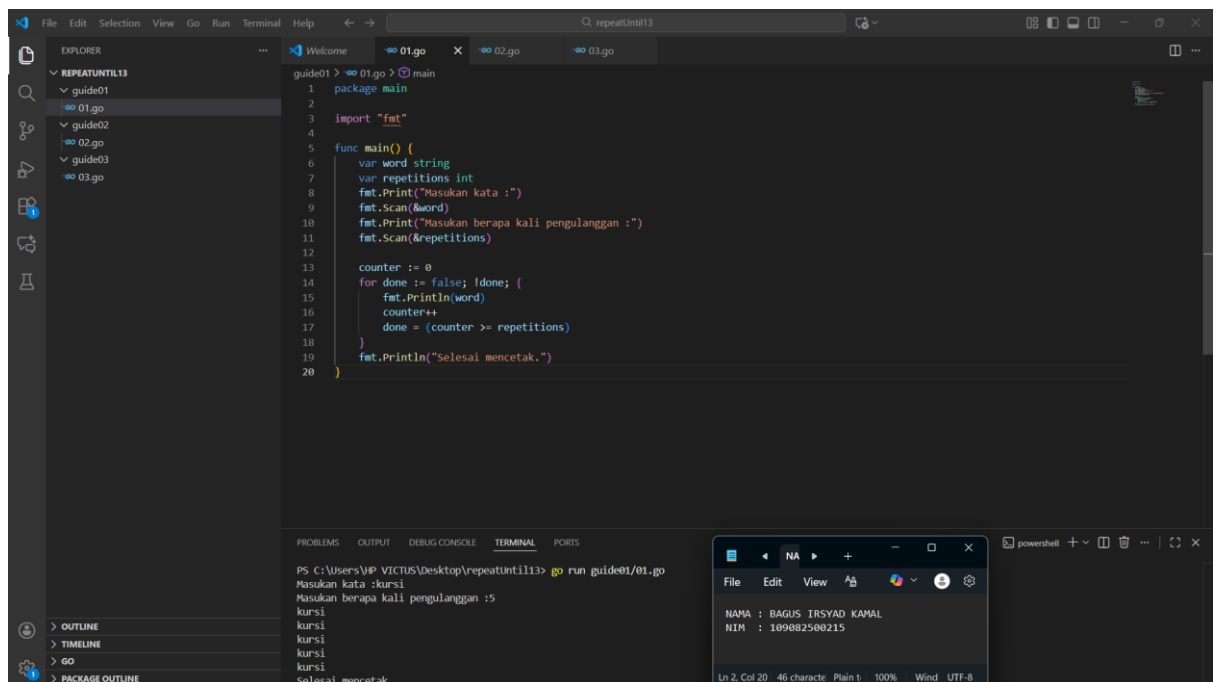
        done = (counter >= repetitions)

    }

    fmt.Println("Selesai mencetak.")

}
```

## Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var word string
7     var repetitions int
8     fmt.Print("Masukan kata :")
9     fmt.Scan(&word)
10    fmt.Print("Masukan berapa kali pengulangan :")
11    fmt.Scan(&repetitions)
12
13    counter := 0
14    for done := false; !done; {
15        fmt.Println(word)
16        counter++
17        done = (counter >= repetitions)
18    }
19    fmt.Println("Selesai mencetak.")
20 }
```

PS C:\Users\HP\_VICTUS\Desktop\repeatUntil13> go run guide01/01.go  
Masukan kata :kursi  
Masukan berapa kali pengulangan :5  
kursi  
kursi  
kursi  
kursi  
kursi  
Selesai mencetak.

## Deskripsi program

Program ini digunakan untuk mencetak sebuah kata sebanyak jumlah pengulangan yang ditentukan oleh pengguna. Program menerima dua input, yaitu sebuah kata (string) dan sebuah bilangan bulat positif yang menyatakan berapa kali kata tersebut akan dicetak ke layar.

Ketika program dijalankan, pertama-tama program meminta pengguna untuk memasukkan sebuah kata. Setelah itu, program meminta pengguna memasukkan jumlah pengulangan. Nilai pengulangan ini menentukan berapa kali kata akan dicetak.

Selanjutnya, program menggunakan perulangan for dengan kondisi berbasis boolean (done) dan sebuah variabel penghitung (counter). Pada setiap iterasi, program mencetak kata yang telah dimasukkan, lalu menaikkan nilai penghitung. Proses ini terus berjalan sampai jumlah cetakan sama dengan jumlah pengulangan yang diminta. Setelah kondisi terpenuhi, perulangan dihentikan dan program mencetak pesan penutup "Selesai mencetak."

Tujuan dari program ini adalah untuk melatih penggunaan perulangan pada bahasa Go serta memahami mekanisme input, kontrol kondisi, dan pencetakan output secara berulang.

**Contoh 1**

Input:

kata = pagi

pengulangan = 3

Output:

pagi

pagi

pagi

Selesai mencetak.

**Contoh 2**

Input:

kata = kursi

pengulangan = 5

Output:

kursi

kursi

kursi

kursi

kursi

Selesai mencetak.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {

    var n int

    for {

        fmt.Print("Masukkan bilangan bulat positif: ")

        fmt.Scan(&n)

        if n > 0 {

            break

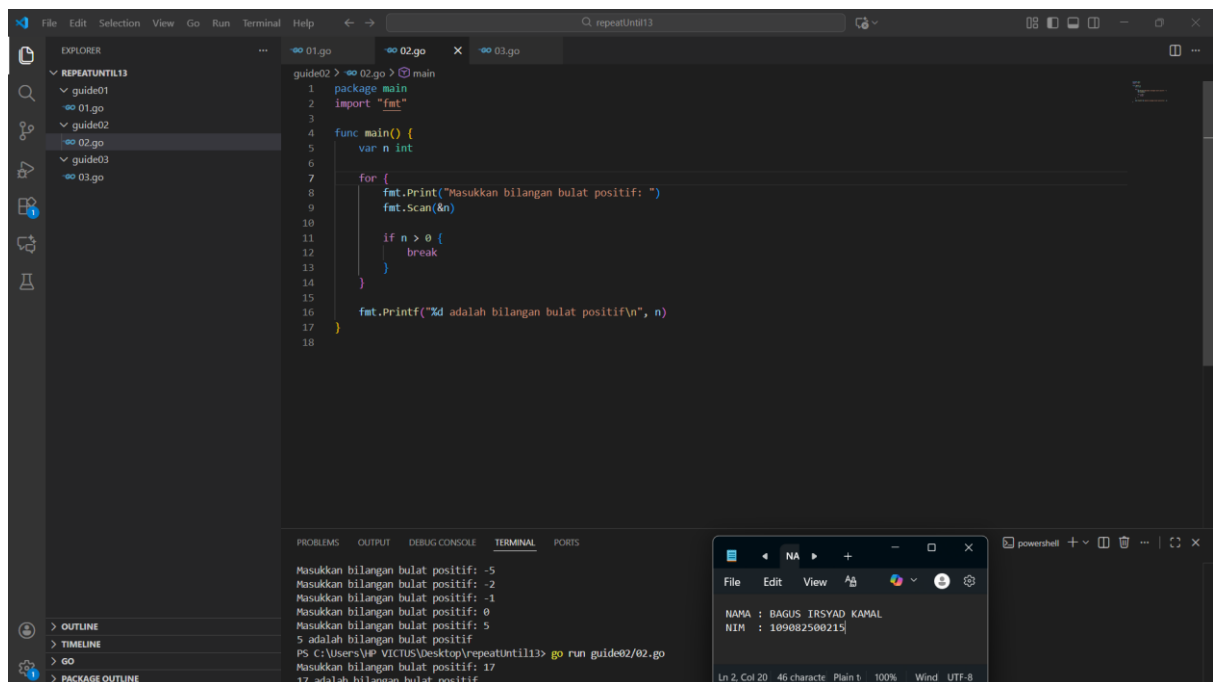
        }

    }

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

}
```

## Screenshoot program



```
1 package main
2 import "fmt"
3
4 func main() {
5     var n int
6
7     for {
8         fmt.Print("Masukkan bilangan bulat positif: ")
9         fmt.Scan(&n)
10
11         if n > 0 {
12             break
13         }
14     }
15
16     fmt.Printf("%d adalah bilangan bulat positif\n", n)
17 }
18
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Masukkan bilangan bulat positif: -5  
Masukkan bilangan bulat positif: -2  
Masukkan bilangan bulat positif: -1  
Masukkan bilangan bulat positif: 0  
Masukkan bilangan bulat positif: 5  
5 adalah bilangan bulat positif  
PS C:\Users\VIP\OneDrive\Desktop\repeatuntil11> go run guide02/02.go  
Masukkan bilangan bulat positif: 17  
17 adalah bilangan bulat positif

## Deskripsi program

Program ini digunakan untuk memastikan bahwa pengguna memasukkan sebuah bilangan bulat positif. Program akan terus meminta input sampai pengguna memasukkan nilai yang lebih besar dari nol.

Ketika program dijalankan, pertama-tama program menampilkan pesan untuk meminta pengguna memasukkan sebuah bilangan bulat positif. Input tersebut kemudian dibaca dan disimpan ke dalam variabel `n`. Program menggunakan perulangan `for` tanpa kondisi awal, sehingga perulangan akan terus berjalan sampai dihentikan secara eksplisit.

Di dalam perulangan, program memeriksa apakah nilai `n` lebih besar dari 0. Jika kondisi ini terpenuhi, perintah `break` dijalankan untuk menghentikan perulangan. Jika nilai yang dimasukkan kurang dari atau sama dengan 0, program akan kembali meminta input hingga pengguna memasukkan bilangan yang valid. Setelah perulangan selesai, program mencetak pesan yang menyatakan bahwa nilai yang dimasukkan adalah bilangan bulat positif.

Tujuan dari program ini adalah untuk melatih penggunaan validasi input menggunakan perulangan pada bahasa Go, sehingga program hanya menerima data yang sesuai dengan kriteria yang ditentukan.

**Contoh 1**

Input:

-5

-2

0

5

Output:

5 adalah bilangan bulat positif

**Contoh 2**

Input:

-1

-10

17

Output:

17 adalah bilangan bulat positif

### 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)

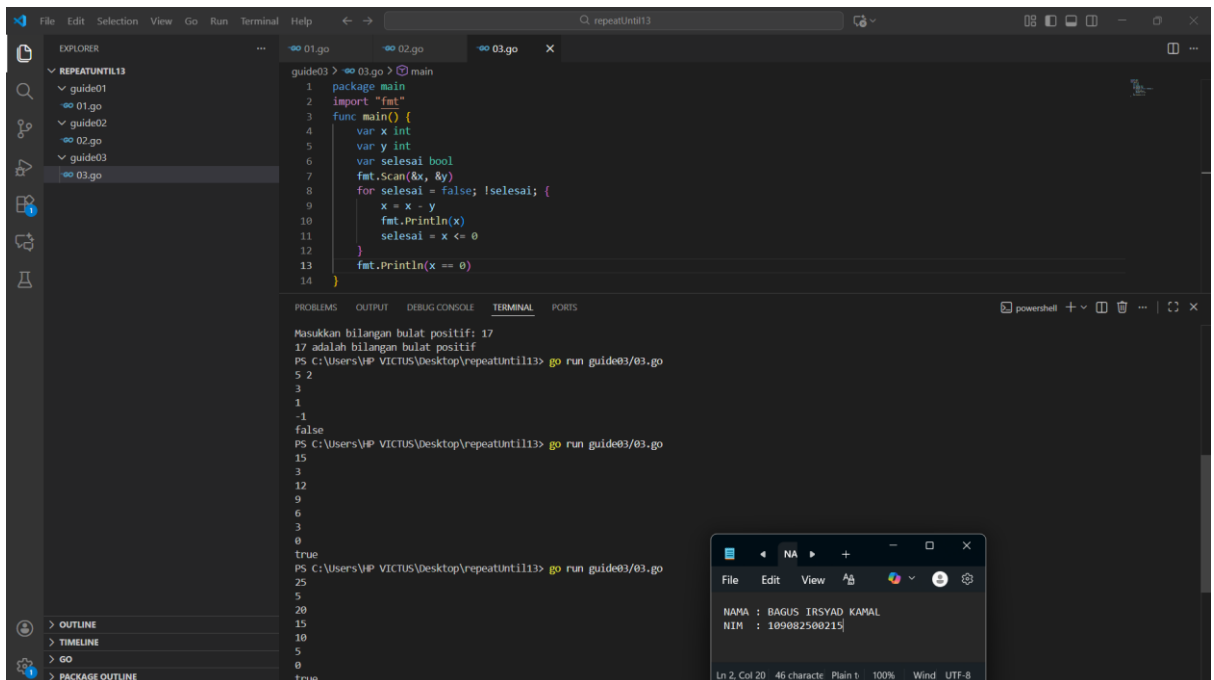
        selesai = x <= 0

    }

    fmt.Println(x == 0)

}
```

#### Screenshoot program



```
guide03 > 03.go > main
1 package main
2 import "fmt"
3 func main() {
4     var x int
5     var y int
6     var selesai bool
7     fmt.Scan(&x, &y)
8     for selesai = false; !selesai; {
9         x = x - y
10        fmt.Println(x)
11        selesai = x <= 0
12    }
13    fmt.Println(x == 0)
14 }
```

Masukkan bilangan bulat positif: 17  
17 adalah bilangan bulat positif  
PS C:\Users\VP VICTUS\Desktop\repeatuntil13> go run guide03/03.go  
5 2  
3  
1  
-1  
false  
PS C:\Users\VP VICTUS\Desktop\repeatuntil13> go run guide03/03.go  
15  
3  
12  
9  
6  
3  
0  
true  
PS C:\Users\VP VICTUS\Desktop\repeatuntil13> go run guide03/03.go  
25  
5  
20  
10  
5  
0  
true

File Edit View NA + - □ ×  
File Edit View + - □ ×  
NAMA : BAGUS IRSYAD KAMAL  
NIM : 109082500215  
Ln 2, Col 20 46 character Plain 100% Wind UTF-8



## Deskripsi program

Program ini digunakan untuk mencetak deret bilangan menurun berdasarkan dua buah bilangan bulat yang dimasukkan oleh pengguna. Program menerima dua input, yaitu bilangan awal (x) dan bilangan pengurang (y).

Ketika program dijalankan, pertama-tama program membaca dua buah bilangan bulat dari input dan menyimpannya ke dalam variabel x dan y. Setelah itu, program menjalankan perulangan for dengan kondisi berbasis boolean menggunakan variabel selesai.

Di dalam perulangan, nilai x akan dikurangi dengan nilai y pada setiap iterasi. Setelah pengurangan dilakukan, nilai x langsung dicetak ke layar. Perulangan akan terus berjalan selama nilai x masih lebih besar dari 0. Jika nilai x sudah kurang dari atau sama dengan 0, variabel selesai akan bernilai benar dan perulangan dihentikan.

Setelah perulangan selesai, program mencetak nilai boolean hasil dari pengecekan  $x == 0$ , yang menunjukkan apakah nilai akhir x tepat sama dengan nol atau tidak.

Tujuan dari program ini adalah untuk melatih penggunaan perulangan repeat-until (disimulasikan dengan for di Go), operasi aritmatika pengurangan, serta evaluasi kondisi logika.

### Contoh 1

Input:

x = 5

y = 2

Output:

3

1

-1

false

### Contoh 2

Input:

x = 15

y = 3

Output:

12

9

6

3

0  
true

### **Contoh 3**

Input:

x = 25

y = 5

Output:

20

15

10

5

0

true

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main() {

    var n int

    var jumlahDigit int

    var selesai bool

    fmt.Print("Masukkan bilangan bulat positif: ")

    fmt.Scan(&n)

    jumlahDigit = 0

    for selesai = false; !selesai; {

        jumlahDigit++

        n = n / 10

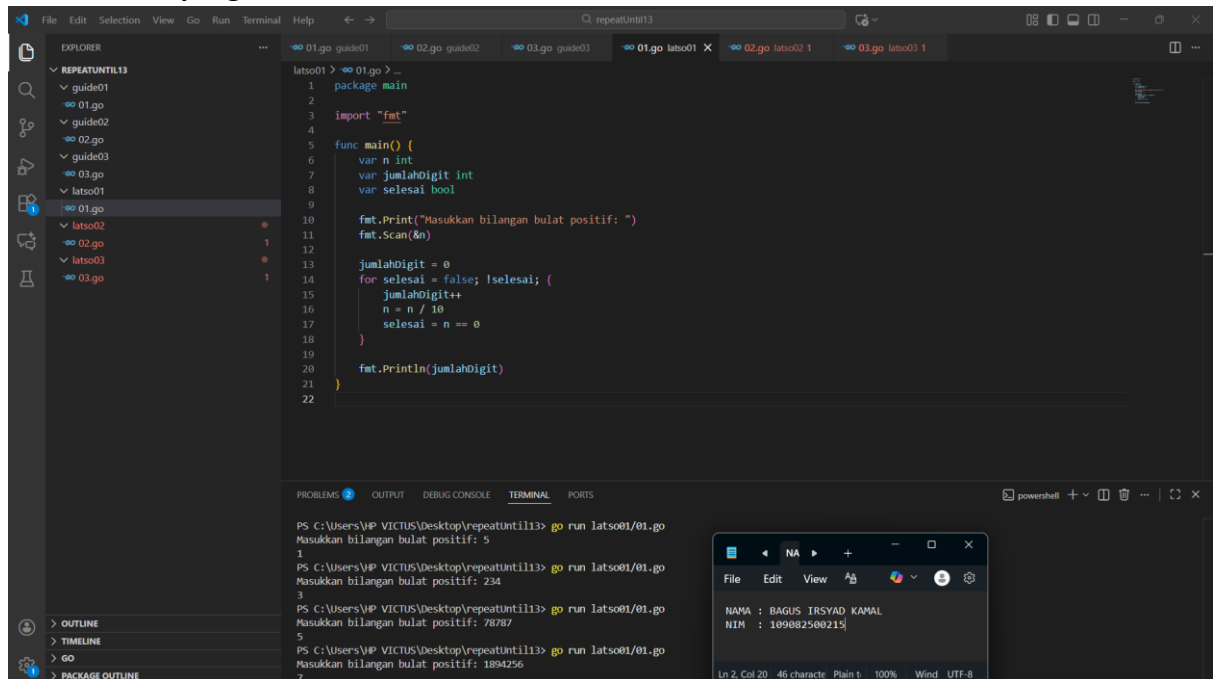
        selesai = n == 0

    }

    fmt.Println(jumlahDigit)

}
```

## Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     var jumlahDigit int
8     var selesai bool
9
10    fmt.Print("Masukkan bilangan bulat positif: ")
11    fmt.Scan(&n)
12
13    jumlahDigit = 0
14    for selesai = false; !selesai; {
15        jumlahDigit++
16        n = n / 10
17        selesai = n == 0
18    }
19
20    fmt.Println(jumlahDigit)
21 }
22
```

```
PS C:\Users\VP VICTUS\Desktop\repeatUntil13> go run latso01/01.go
Masukkan bilangan bulat positif: 5
1
PS C:\Users\VP VICTUS\Desktop\repeatUntil13> go run latso01/01.go
Masukkan bilangan bulat positif: 234
3
PS C:\Users\VP VICTUS\Desktop\repeatUntil13> go run latso01/01.go
Masukkan bilangan bulat positif: 78787
5
PS C:\Users\VP VICTUS\Desktop\repeatUntil13> go run latso01/01.go
Masukkan bilangan bulat positif: 1894256
7
```

File Edit View NA + - x  
File Edit View  
NAMA : BAGUS IRSYAD KAMAL  
NIM : 109082500215  
Ln 2, Col 20 46 character Plain 100% Wind UTF-8

## Deskripsi program

Program ini digunakan untuk menghitung banyaknya digit dari suatu bilangan bulat positif. Program menerima satu input berupa bilangan bulat positif dan menghasilkan keluaran berupa jumlah digit dari bilangan tersebut.

Ketika program dijalankan, pertama-tama program membaca sebuah bilangan bulat positif dan menyimpannya ke dalam variabel `n`. Selanjutnya, program menggunakan perulangan `repeat-until` yang disimulasikan dengan perulangan `for` dan sebuah variabel boolean `selesai`.

Di dalam perulangan, program membagi nilai `n` dengan 10 untuk menghilangkan satu digit terakhir. Setiap kali pembagian dilakukan, variabel `jumlahDigit` akan ditambah satu sebagai penanda bahwa satu digit telah dihitung. Setelah itu, program memeriksa apakah nilai `n` sudah menjadi 0. Jika kondisi tersebut terpenuhi, variabel `selesai` akan bernilai benar dan perulangan dihentikan.

Proses ini akan terus berulang sampai seluruh digit pada bilangan telah habis dibagi. Setelah perulangan `selesai`, program mencetak jumlah digit yang telah dihitung.

Tujuan dari program ini adalah untuk melatih penggunaan perulangan `repeat-until`, operasi pembagian bilangan bulat, serta pemrosesan digit pada sebuah bilangan.

## **Contoh Masukan dan Keluaran**

### **Contoh 1**

Input:

5

Output:

1

### **Contoh 2**

Input:

234

Output:

3

### **Contoh 3**

Input:

78787

Output:

5

### **Contoh 4**

Input:

1894256

Output:

7

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {

    var n float64

    var nilai int

    var batas int

    var selesai bool


    fmt.Print("Masukkan bilangan desimal: ")

    fmt.Scan(&n)

    nilai = int(n * 10)


    if nilai%10 == 0 {

        batas = nilai

    } else {

        batas = (nilai/10 + 1) * 10

    }

    for selesai = false; !selesai; {

        nilai++

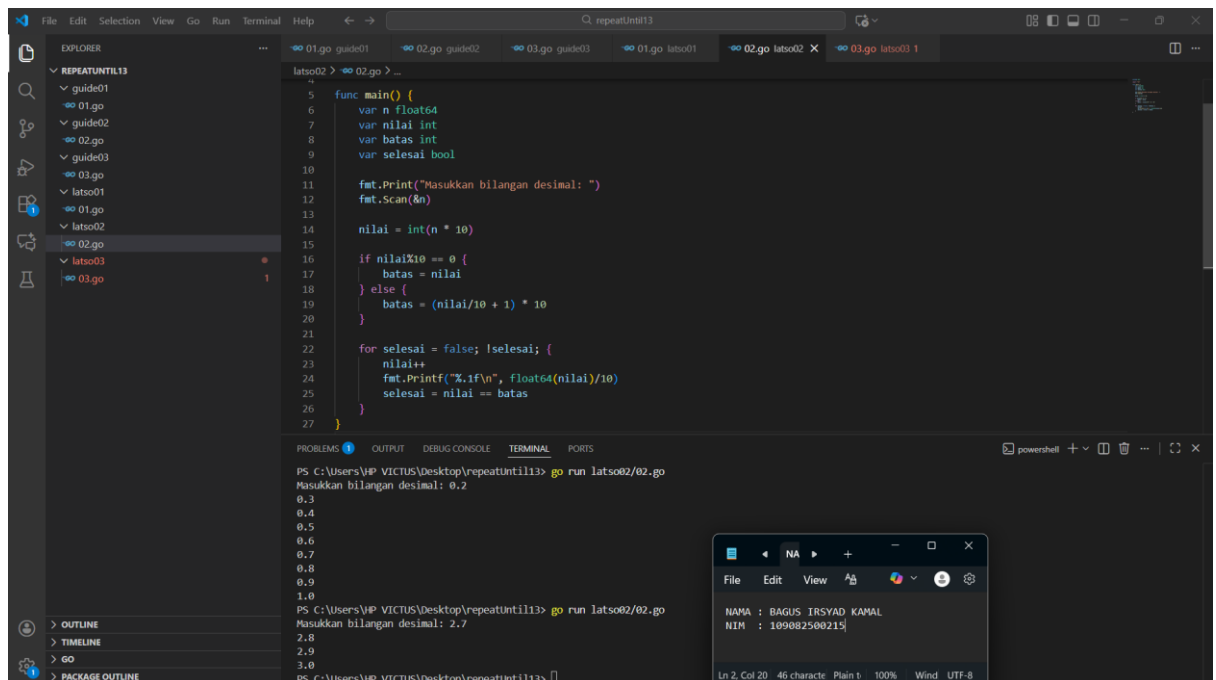
        fmt.Printf("%.1f\n", float64(nilai)/10)

        selesai = nilai == batas

    }

}
```

## Screenshoot program



```
5 func main() {
6     var n float64
7     var nilai int
8     var batas int
9     var selesai bool
10
11     fmt.Print("Masukkan bilangan desimal: ")
12     fmt.Scan(&n)
13
14     nilai = int(n * 10)
15
16     if nilai%10 == 0 {
17         batas = nilai
18     } else {
19         batas = (nilai/10 + 1) * 10
20     }
21
22     for selesai = false; !selesai; {
23         nilai++
24         fmt.Printf("%.1f\n", float64(nilai)/10)
25         selesai = nilai == batas
26     }
27 }
```

PS C:\Users\VIP VICTUS\Desktop\repeatUntil13> go run latso02/02.go  
Masukkan bilangan desimal: 0.2  
0.3  
0.4  
0.5  
0.6  
0.7  
0.8  
0.9  
1.0  
PS C:\Users\VIP VICTUS\Desktop\repeatUntil13> go run latso02/02.go  
Masukkan bilangan desimal: 2.7  
2.8  
2.9  
3.0  
PS C:\Users\VIP VICTUS\Desktop\repeatUntil13> |

NAMA : BAGUS IRSYAD KAMAL  
NIM : 109082500215

## Deskripsi program

Program ini digunakan untuk mendapatkan bilangan bulat optimal dari sebuah bilangan desimal dengan cara melakukan penjumlahan bertahap sampai mencapai pembulatan ke atas dari bilangan yang diinputkan. Program menerima satu input berupa bilangan desimal.

Ketika program dijalankan, pertama-tama program membaca sebuah bilangan desimal dan menyimpannya ke dalam variabel `n`. Nilai awal perulangan ditentukan dari hasil pembulatan ke bawah bilangan tersebut menggunakan fungsi `Floor`.

Selanjutnya, program menjalankan perulangan `repeat-until` yang disimulasikan menggunakan perulangan `for` dengan variabel boolean `selesai`. Pada setiap iterasi, nilai hasil akan ditambah sebesar 0.1 dan langsung dicetak ke layar dengan satu angka di belakang koma.

Setelah penambahan dilakukan, program memeriksa apakah nilai hasil sudah lebih besar atau sama dengan hasil pembulatan ke atas dari bilangan input (`Ceil(n)`). Jika kondisi tersebut terpenuhi, variabel `selesai` akan bernilai benar dan perulangan dihentikan.

Proses ini terus berulang hingga nilai hasil mencapai bilangan bulat optimal, yaitu pembulatan ke atas dari bilangan desimal yang dimasukkan.

Tujuan dari program ini adalah untuk melatih penggunaan perulangan repeat-until, operasi pembulatan bilangan desimal, serta proses iterasi bertahap menuju suatu kondisi tertentu.

### **Contoh Masukan dan Keluaran**

#### **Contoh 1**

Input:

0.2

Output:

0.3

0.4

0.5

0.6

0.7

0.8

0.9

1.0

#### **Contoh 2**

Input:

2.7

Output:

2.8

2.9

3.0



### Tugas 3

#### Source code

```
package main
import "fmt"

func main() {
    var target int
    var donasi int
    var total int
    var jumlahDonatur int
    var selesai bool

    fmt.Scan(&target)

    total = 0
    jumlahDonatur = 0

    for selesai = false; !selesai; {
        fmt.Scan(&donasi)
        jumlahDonatur++
        total = total + donasi

        fmt.Printf(
            "Donatur %d : Menyumbang %d. Total
terkumpul: %d\n",
            jumlahDonatur, donasi, total,
        )

        selesai = total >= target
    }

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

## Screenshoot program

The screenshot shows a Go IDE with a project named 'repeatuntil13'. The Explorer panel on the left shows a directory structure with files 'guide01', 'guide02', 'guide03', 'latso01', 'latso02', 'latso03', and '03.go'. The main editor displays the code for '03.go', which is a Go program that reads a target value and then repeatedly asks for donations until the total reaches or exceeds the target. The code uses a 'for' loop with a boolean 'selesai' variable to control the repetition. The output window at the bottom shows the program's execution, displaying the target (300) and the cumulative total of donations (300) after 4 donations. A small window in the bottom right corner shows a text editor with the name 'BAGUS IRSYAD KAMAL' and NIM '109082500215'.

```
03.go guide01 02.go guide02 03.go guide03 01.go latso01 02.go latso02 03.go latso03 X
EXPLORER
REPEATUNTIL13
guide01
guide02
guide03
latso01
latso02
latso03
03.go

10 func main() {
11     fmt.Scan(&target)
12     total = 0
13     jumlahDonatur = 0
14     for selesai = false; !selesai; {
15         fmt.Scan(&donasi)
16         jumlahDonatur++
17         total = total + donasi
18         fmt.Printf(
19             "Donatur %d : Menyumbang %d. Total terkumpul: %d\n",
20             jumlahDonatur, donasi, total,
21         )
22         selesai = total >= target
23     }
24     fmt.Printf(
25         "Target tercapai! Total donasi: %d dari %d donatur.\n",
26         total, jumlahDonatur,
27     )
28 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\VP VICTUS\Desktop\repeatuntil13> go run latso03/03.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\VP VICTUS\Desktop\repeatuntil13> go run latso03/03.go
200
300
Donatur 1 : Menyumbang 300. Total terkumpul: 300
Target tercapai! Total donasi: 300 dari 1 donatur.
```

## Deskripsi program

Program ini digunakan untuk menghitung total donasi yang dikumpulkan oleh sebuah organisasi amal hingga mencapai atau melebihi target yang telah ditentukan. Program menerima masukan berupa bilangan bulat, yaitu target donasi dan sumbangan dari setiap donatur.

Ketika program dijalankan, pertama-tama program membaca sebuah bilangan bulat yang menyatakan target donasi. Setelah itu, program akan terus menerima input berupa jumlah donasi dari setiap donatur.

Program menggunakan perulangan repeat-until yang disimulasikan dengan perulangan for dan sebuah variabel boolean selesai. Pada setiap perulangan, program membaca nilai donasi, menambahkan nilai tersebut ke total donasi, serta menghitung jumlah donatur. Setiap kali donasi diterima, program menampilkan informasi donatur ke berapa, jumlah sumbangan, dan total donasi yang telah terkumpul.

Perulangan akan terus berjalan sampai total donasi yang terkumpul mencapai atau melebihi target donasi. Ketika kondisi tersebut terpenuhi, variabel selesai bernilai benar dan perulangan dihentikan.

Setelah perulangan selesai, program menampilkan pesan bahwa target donasi telah tercapai, beserta total donasi yang terkumpul dan jumlah donatur yang berpartisipasi.

Tujuan dari program ini adalah untuk melatih penggunaan perulangan repeat-until, proses akumulasi nilai, serta pengolahan input secara berulang hingga suatu kondisi terpenuhi.

### **Contoh Masukan dan Keluaran**

#### **Contoh 1**

Masukan:

300

100

50

200

Keluaran:

Donatur 1 : Menyumbang 100. Total terkumpul: 100

Donatur 2 : Menyumbang 50. Total terkumpul: 150

Donatur 3 : Menyumbang 200. Total terkumpul: 350

Target tercapai! Total donasi: 350 dari 3 donatur.

#### **Contoh 2**

Masukan:

500

150

100

50

300

Keluaran:

Donatur 1 : Menyumbang 150. Total terkumpul: 150

Donatur 2 : Menyumbang 100. Total terkumpul: 250

Donatur 3 : Menyumbang 50. Total terkumpul: 300

Donatur 4 : Menyumbang 300. Total terkumpul: 600

Target tercapai! Total donasi: 600 dari 4 donatur.

#### **Contoh 3**

Masukan:

200

300

Keluaran:

Donatur 1 : Menyumbang 300. Total terkumpul: 300

Target tercapai! Total donasi: 300 dari 1 donatur.