

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 13  
REPEAT - UNTIL**



**Disusun oleh:**

**Janica Prima Ginting**

**109082500064**

**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 kata string

    var jmlh int

    fmt.Scan(&kata)

    fmt.Scan(&jmlh)

    for done := false ; !done;{

        fmt.Println(kata)

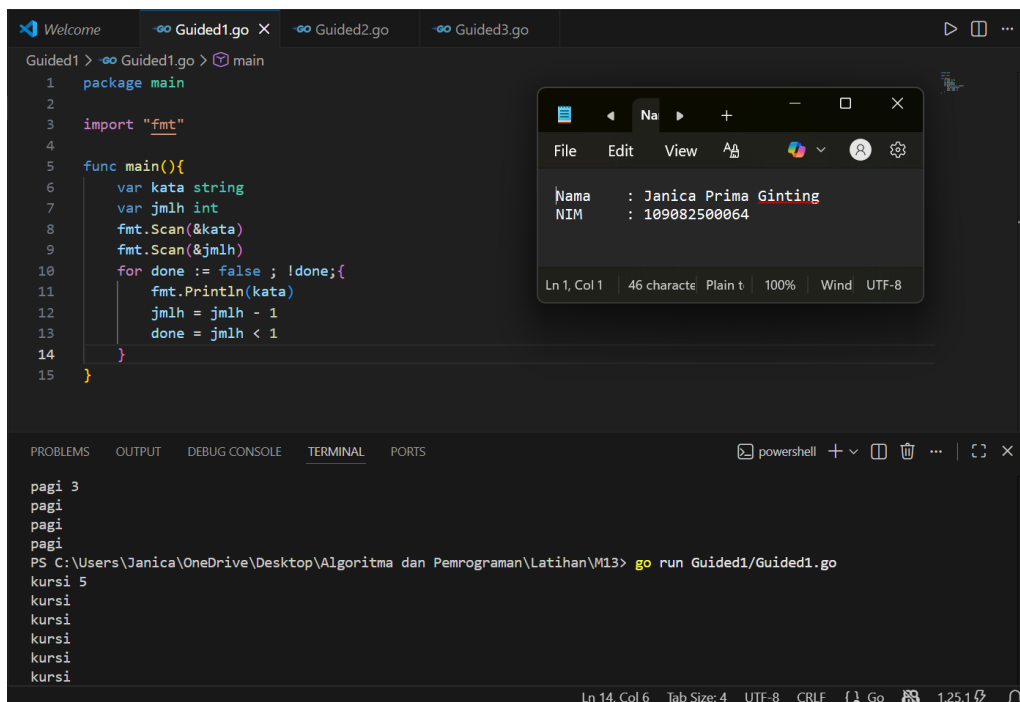
        jmlh = jmlh - 1

        done = jmlh < 1

    }

}
```

### Screenshoot program



**Deskripsi program**

Program golang ini berfungsi untuk menerima input kata dan mencetaknya sebanyak jumlah pengulangan yang diinginkan oleh pengguna. Program akan dihentikan ketika jumlah kata yang dicetak mencapai jumlah yang diinginkan oleh pengguna.

Cara Kerja Program :

Pertama variable kata dideklarasikan dengan tipe data string

Kedua variable jmlh dideklarasikan dengan tipe data integer

Ketiga fmt.Scan untuk pengguna menginputkan kata di variable kata dan bilangan bulat di variable jmlh

Keempat var done diberi nilai false

Terakhir masuk program perulangan / for jika tidak done atau negasi done maka,

- Mengeluarkan output dari variable kata
- Nilai variable jmlh dikurangi 1
- nilai done sama dengan var jmlh kurang dari 1.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main(){

    var n int

    for done := false ; !done ; {

        fmt.Scan(&n)

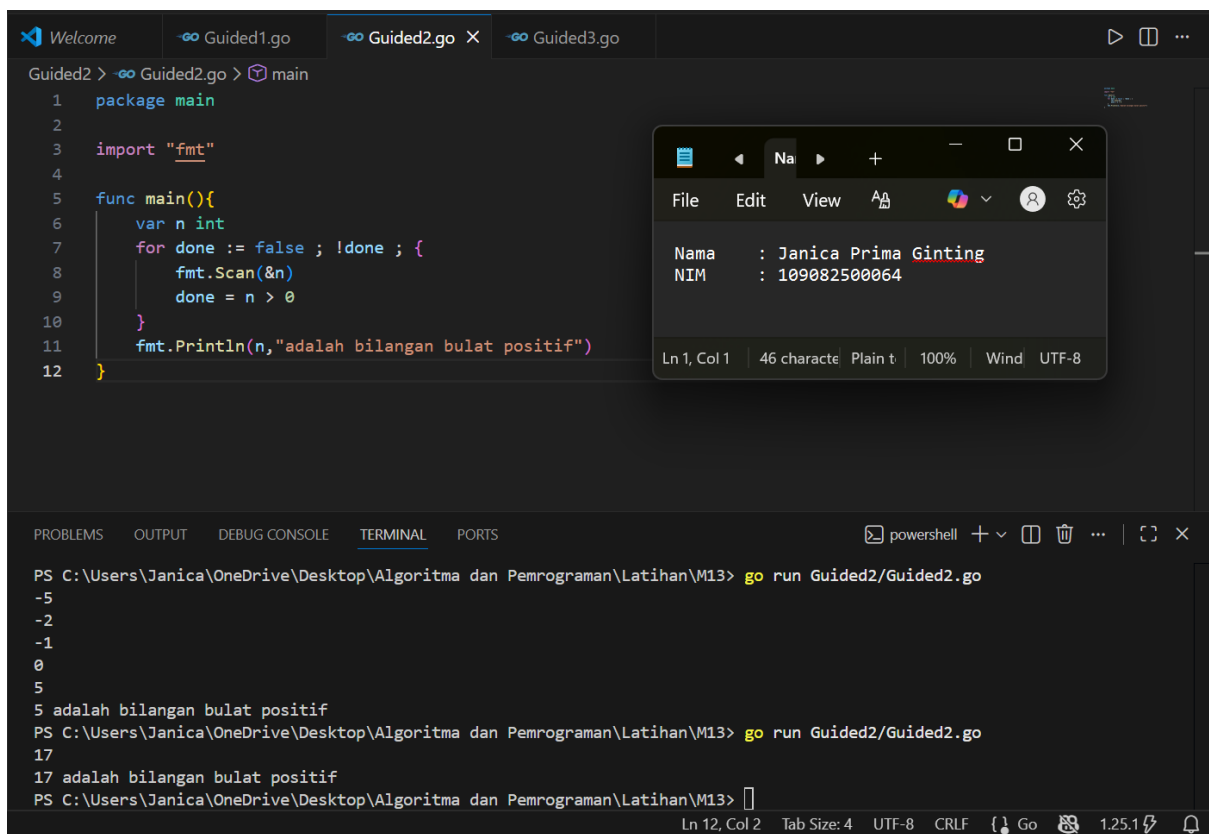
        done = n > 0

    }

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

}
```

### Screenshoot program



The screenshot shows a Go IDE with three tabs: Guided1.go, Guided2.go (active), and Guided3.go. The code in Guided2.go is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main(){
6     var n int
7     for done := false ; !done ; {
8         fmt.Scan(&n)
9         done = n > 0
10    }
11    fmt.Println(n,"adalah bilangan bulat positif")
12 }
```

A floating window titled 'Na' is open, displaying the following information:

```
File Edit View
Nama : Janica Prima Ginting
NIM : 109082500064
Ln 1, Col 1 | 46 character Plain t | 100% | Wind UTF-8
```

The terminal at the bottom shows the execution of the program:

```
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M13> go run Guided2/Guided2.go
-5
-2
-1
0
5
5 adalah bilangan bulat positif
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M13> go run Guided2/Guided2.go
17
17 adalah bilangan bulat positif
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M13>
```

**Deskripsi program**

Program Golang ini berfungsi untuk meminta pengguna untuk memasukkan bilangan bulat positif. Program akan terus meminta input hingga pengguna memasukkan bilangan bulat positif.

Cara kerja program :

Pertama variable n dideklarasikan dengan tipe data integer

Kedua program perulangan / for done diberi nilai false , jika tidak done maka program akan berhenti, jika kode masih done / bernilai false maka,

- Pengguna akan diminta menginputkan bilangan ke variable n
- Nilai done sama dengan var n lebih besar dari 0

Terakhir akan mengeluarkan output nilai dari n,"adalah bilangan bulat positif".

### 3. Guided 3

#### Source Code

```
package main

import "fmt"

func main(){

    var x, y int

    fmt.Scan(&x, &y)

    done := false

    for !done {

        x = x - y

        fmt.Println(x)

        done = x <= 0 ;

    }

    fmt.Println(x == 0)

}
```

#### Screenshoot program

The screenshot displays an IDE with a Go source code file named `Guided3.go` and a terminal window showing the program's execution. The code defines a `main` function that reads two integers `x` and `y` from standard input, then enters a loop where `x` is decremented by `y` until it is less than or equal to zero. The program prints the value of `x` at each iteration and finally checks if `x` is equal to zero.

The terminal output shows the program being run with the command `go run Guided3/Guided3.go`. The input values are `5` and `2`. The output shows the value of `x` decreasing from 5 to 0, with the final state being `x == 0`.

```
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M13> go run Guided3/Guided3.go
5 2
3
1
-1
false
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M13> go run Guided3/Guided3.go
15 3
12
9
6
3
0
true
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M13> go run Guided3/Guided3.go
25 5
20
15
10
5
0
true
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\Latihan\M13>
```

**Deskripsi program**

Program Golang ini berfungsi untuk melakukan pengecekan apakah suatu bilangan merupakan kelipatan dari bilangan lainnya.

Cara kerja program :

Pertama variable x dan y dideklarasikan dengan tipe data integer

Kedua Scan untuk menginputkan bilangan bulat positif ke variable x dan y

Ketiga variable done diberi nilai false

Keempat masuk program perulangan / for tidak done / negasi done maka,

- Nilai var x sama dengan var x dikurang var y
- Output hasil nilai var x keluar
- Nilai done sama dengan x lebih kecil sama dengan dari 0

Terakhir akan mengeluarkan output dari kondisi ( var x sama dengan 0 ) jika nilai x sama dengan 0 maka outputnya akan True, jika nilai lewat dari 0 atau tidak sama dengan 0 maka outputnya False.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

func main(){

    var bilangan int

    fmt.Scan(&bilangan)

    digit := 0

    habis := false

    for !habis{

        bilangan = bilangan / 10

        digit ++

        habis = bilangan <= 0

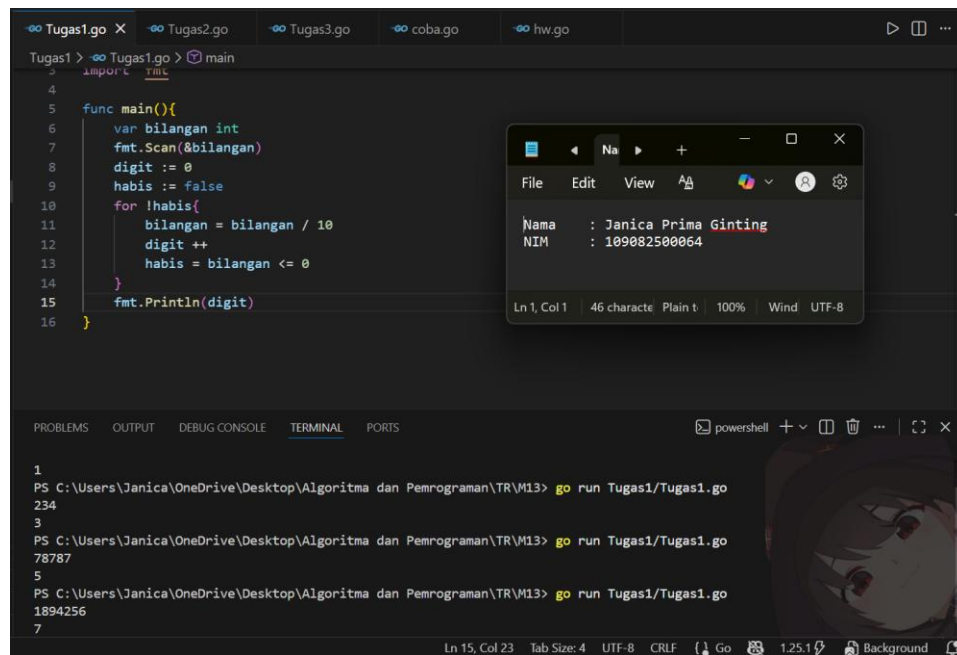
    }

    fmt.Println(digit)

}
```



## Screenshoot program



The screenshot shows a Go program in a text editor and its execution in a terminal. The program is a simple loop that calculates the number of digits in a given integer by repeatedly dividing it by 10 until it becomes zero.

```
1 import "fmt"
2
3 func main(){
4     var bilangan int
5     fmt.Scan(&bilangan)
6     digit := 0
7     habis := false
8     for !habis{
9         bilangan = bilangan / 10
10        digit ++
11        habis = bilangan <= 0
12    }
13    fmt.Println(digit)
14 }
```

The terminal output shows the program being run three times with different inputs, each time correctly outputting the number of digits:

```
1 PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M13> go run Tugas1/Tugas1.go
2 234
3
4 PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M13> go run Tugas1/Tugas1.go
5 78787
6
7 PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M13> go run Tugas1/Tugas1.go
8 1894256
```

## Deskripsi program

Program Golang ini berfungsi untuk menghitung banyaknya digit dari suatu Bilangan.

Cara kerja program :

Pertama variable bilangan dideklarasikan dengan tipe data integer

Kedua `fmt.Scan(&bilangan)` untuk pengguna menginputkan bilangan bulat

Ketiga variable digit diberi nilai 0

Keempat variable habis diberi nilai false

Kelima program perulangan / for tidak habis atau negasi habis maka,

- Nilai bilangan sama dengan bilangan dibagi 10
- Nilai digit sama dengan digit ditambah 1
- Nilai habis sama dengan bilangan lebih kecil sama dengan dari 0

Terakhir mengeluarkan output nilai var digit.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {

    var bilangan float64

    fmt.Scan(&bilangan)


    uNilai := int(bilangan * 10)

    nDepan := uNilai / 10

    nAkhir := (nDepan + 1) * 10


    kondisi := false


    for !kondisi {

        uNilai = uNilai + 1

        if uNilai % 10 == 0 {

            fmt.Println(uNilai / 10)

        }else{

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

        }

        if uNilai >= nAkhir {

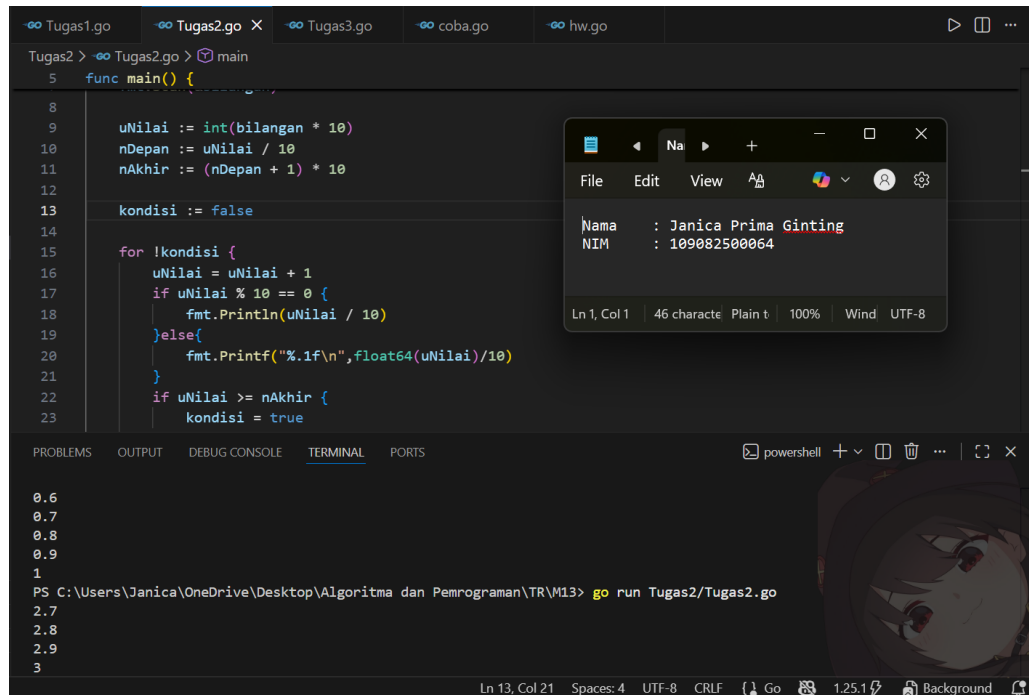
            kondisi = true

        }

    }

}
```

## Screenshoot program



The screenshot shows a Go IDE with the following code in `Tugas2.go`:

```
5 func main() {  
8  
9     uNilai := int(bilangan * 10)  
10    nDepan := uNilai / 10  
11    nAkhir := (nDepan + 1) * 10  
12  
13    kondisi := false  
14  
15    for !kondisi {  
16        uNilai = uNilai + 1  
17        if uNilai % 10 == 0 {  
18            fmt.Println(uNilai / 10)  
19        } else {  
20            fmt.Printf("%.1f\n", float64(uNilai)/10)  
21        }  
22        if uNilai >= nAkhir {  
23            kondisi = true  
24        }  
25    }  
26 }
```

The terminal output shows the following sequence of values:

```
0.6  
0.7  
0.8  
0.9  
1  
2.7  
2.8  
2.9  
3
```

The terminal command executed was `go run Tugas2/Tugas2.go`.

## Deskripsi program

Program Golang ini berfungsi untuk mendapatkan bilangan bulat optimal dari bilangan yang telah diinputkan. Melakukan penjumlahan tiap perulangan mencapai pembulatan keatas dari bilangan yang diinputkan.

Cara kerja Program :

Pertama variable bilangan dideklarasikan dengan tipe data float64

Kedua fmt.Scan untuk pengguna menginputkan bilangan decimal

Ketiga variable uNilai diberi nilai dari var bilangan dikali 10 lalu hasilnya dijadikan integer

Keempat variable nDepan diberi nilai dari var uNilai dibagi 10

Kelima variable nAkhir diberi nilai dari var nDepan ditambah 1 lalu dikali 10

Keenam variable kondisi diberi nilai false

Terakhir program perulangan / for tidak kondisi atau negasi kondisi maka,

- Nilai var uNilai sama dengan uNilai ditambah 1
- Program if kondisi( uNilai dimodulus atau dibagi sisa 10 sama dengan 0 ) maka,
  - o Akan mengeluarkan output hasil dari var uNilai dibagi 10
  - o Jika tidak maka masuk ke else
    - Akan mengeluarkan output dari uNilai yang diubah menjadi float64 dan dibagi 10 dengan 1 angka dibelakang koma.
- Program if kondisi( var uNilai lebih besar sama dengan dari var nAkhir ) maka,
  - o Nilai kondisi sama dengan true.

### 3. Tugas 3

#### Source code

```
package main

import "fmt"

func main(){

    var tDonasi, donasi int

    fmt.Scan(&tDonasi)

    donatur := 0

    tTerkumpul := 0

    tTercapai := false

    for !tTercapai {

        fmt.Scan(&donasi)

        donatur = donatur + 1

        tTerkumpul = tTerkumpul + donasi

        fmt.Println("Donatur", donatur, ": Menyumbang",
donasi, ". Total terkumpul:", tTerkumpul)

        tTercapai = tTerkumpul >= tDonasi

    }

    fmt.Println("Target tercapai! Total donasi:",
tTerkumpul, "dari", donatur, "donatur.")

}
```

## Screenshoot program

The screenshot displays a Go development environment with two windows. The top window shows the source code for `Tugas3.go`, which implements a donation collection program. The code defines variables for total donation (`tDonasi`), current donation (`donasi`), total collected (`tTerkumpul`), and whether the target is reached (`tTercapai`). It uses a `for` loop to repeatedly prompt for donations until the target is reached.

```
1 package main
2
3 import "fmt"
4
5 func main(){
6     var tDonasi, donasi int
7     fmt.Scan(&tDonasi)
8     donatur := 0
9     tTerkumpul := 0
10    tTercapai := false
11    for !tTercapai {
12        fmt.Scan(&donasi)
13        donatur = donatur + 1
14        tTerkumpul = tTerkumpul + donasi
15        fmt.Println("Donatur", donatur, ": Menyumbang", donasi, ". Total terkumpul:", tTerkumpul)
16        tTercapai = tTerkumpul >= tDonasi
17    }
18    fmt.Println("Target tercapai! Total donasi:", tTerkumpul, "dari", donatur, "donatur.")
19 }
```

The bottom window shows the terminal output of running the program. It demonstrates three successful runs where the target donation amount is reached. Each run shows the cumulative total of donations and the number of donors.

```
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M13> go run Tugas3/Tugas3.go
300
100
Donatur 1 : Menyumbang 100 . Total terkumpul: 100
50
Donatur 2 : Menyumbang 50 . Total terkumpul: 150
200
Donatur 3 : Menyumbang 200 . Total terkumpul: 350
Target tercapai! Total donasi: 350 dari 3 donatur.
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M13> 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\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M13> go run Tugas3/Tugas3.go
200
300
Donatur 1 : Menyumbang 300 . Total terkumpul: 300
Target tercapai! Total donasi: 300 dari 1 donatur.
PS C:\Users\Janica\OneDrive\Desktop\Algoritma dan Pemrograman\TR\M13> 
```

## Deskripsi program

Program Golang ini berfungsi untuk menerima sumbangan / bilangan bulat yang akan menghitung donator / berapa kali inputan dengan total terkumpul setiap kali inputan baru hingga donasi tercapai.

Cara kerja program :

Pertama variable `tDonasi` dan `donasi` dideklarasikan dengan tipe data integer

Kedua pengguna menginputkan bilangan bulat / target donasi ke variable `tDonasi`

Ketiga variable donator diberi nilai 0

Keempat variable `tTerkumpul` diberi nilai 0

Kelima variable `tTercapai` diberi nilai false

Keenam program perulangan / for tidak `tTercapai` atau negasi `tTercapai` maka,

- Meminta pengguna menginputkan bilangan bulat untuk var donasi

- Nilai var donator ditambah 1
- Nilai var tTerkumpul sama dengan tTerkumpul ditambah donasi
- Mengeluarkan output Donatur beberapa: Menyumbang berapa. Total terkumpul: dari semua donasi
- Nilai tTercapai sama dengan tTerkumpul lebih besar sama dengan dari tDonasi

Terakhir mengeluarkan output Target tercapai! Total donasi: semua donasi dari banyaknya donator / input donatur.