

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 13
Repeat-Until**



Disusun oleh:

Rayhan Ahza Widyamukti

109082500210

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 jumlah int
    var hitung int

    fmt.Print("Masukkan kata: ")
    fmt.Scan(&kata)

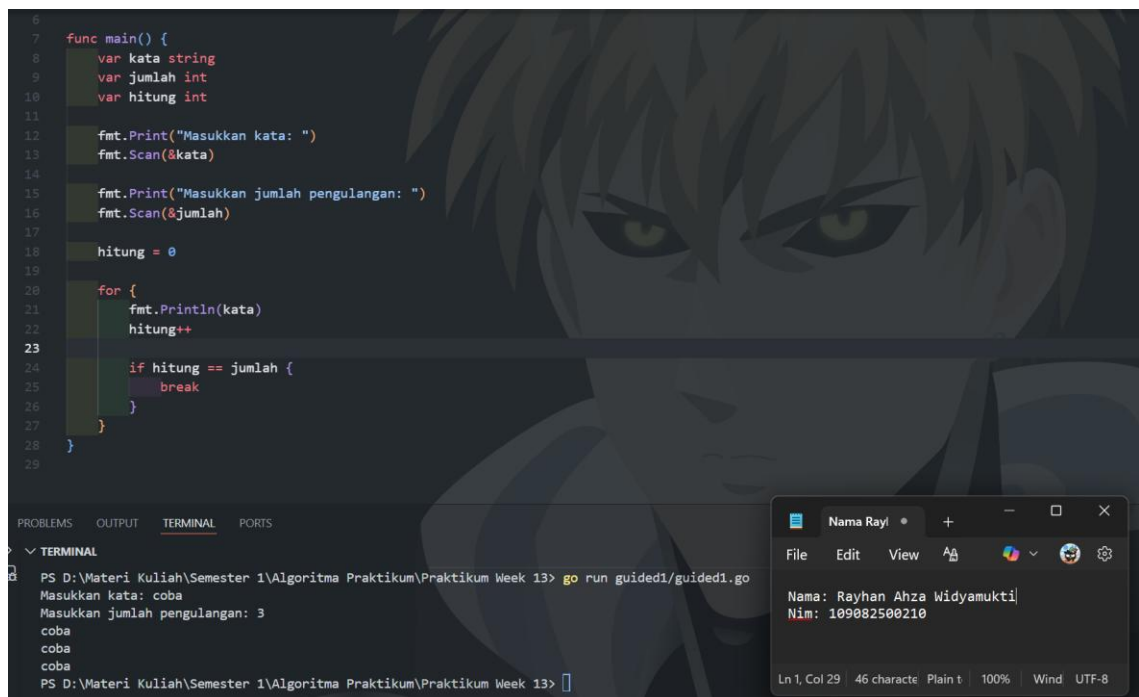
    fmt.Print("Masukkan jumlah pengulangan: ")
    fmt.Scan(&jumlah)

    hitung = 0

    for {
        fmt.Println(kata)
        hitung++

        if hitung == jumlah {
            break
        }
    }
}
```

Screenshoot program

The screenshot shows a code editor with a Go program and a terminal window. The Go code defines a main function that takes a string 'kata' and an integer 'jumlah'. It uses a loop to print 'kata' 'jumlah' times. The terminal shows the execution of the program, where 'coba' is entered for 'kata' and '3' for 'jumlah', resulting in 'coba' being printed three times. A small window in the bottom right corner displays the user's name 'Nama Rayhan Ahza Widyamukti' and their student ID 'Nim: 109082500210'.

```
6
7 func main() {
8     var kata string
9     var jumlah int
10    var hitung int
11
12    fmt.Print("Masukkan kata: ")
13    fmt.Scan(&kata)
14
15    fmt.Print("Masukkan jumlah pengulangan: ")
16    fmt.Scan(&jumlah)
17
18    hitung = 0
19
20    for {
21        fmt.Println(kata)
22        hitung++
23
24        if hitung == jumlah {
25            break
26        }
27    }
28 }
29
```

PROBLEMS OUTPUT **TERMINAL** PORTS

PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13> go run guided1/guided1.go
Masukkan kata: coba
Masukkan jumlah pengulangan: 3
coba
coba
coba
PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13>

Nama Rayhan Ahza Widyamukti
Nim: 109082500210

Ln 1, Col 29 46 character Plain text 100% Wind UTF-8

Deskripsi program

Program tersebut meminta pengguna memasukkan sebuah kata dan jumlah pengulangannya, lalu mencetak kata tersebut sesuai jumlah yang diminta menggunakan mekanisme perulangan tanpa batas (infinite loop). Variabel hitung digunakan sebagai penghitung iterasi, dimulai dari 0. Setiap kali loop dieksekusi, program menampilkan kata, kemudian menaikkan nilai hitung sebesar 1. Ketika nilai hitung sudah mencapai nilai jumlah yang dimasukkan pengguna, kondisi `if hitung == jumlah` terpenuhi sehingga pernyataan `break` dijalankan untuk menghentikan loop. Dengan demikian, program memastikan kata dicetak tepat sebanyak jumlah pengulangan yang diminta.

2. Guided 2

Source Code

```
package main

import (
    "fmt"
)

func main() {
    var n int

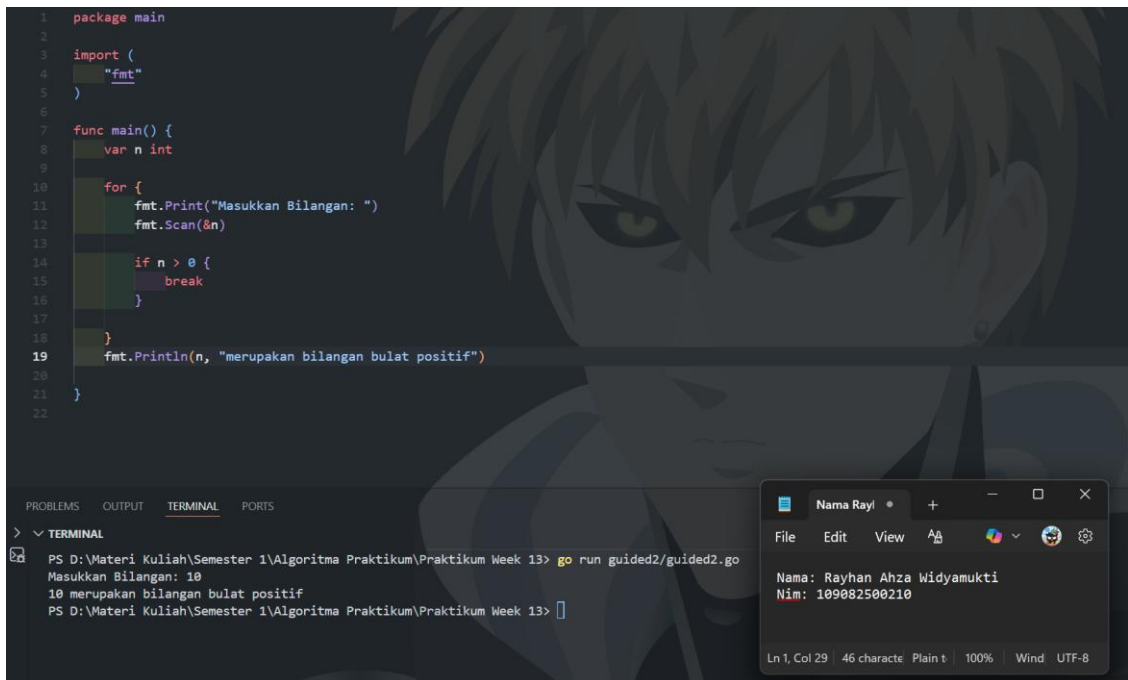
    for {
        fmt.Print("Masukkan Bilangan: ")
        fmt.Scan(&n)

        if n > 0 {
            break
        }

    }

    fmt.Println(n, "merupakan bilangan bulat positif")
}
```

Screenshoot program

The screenshot shows a code editor with a Go program. The code is as follows:

```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var n int
9
10    for {
11        fmt.Print("Masukkan Bilangan: ")
12        fmt.Scan(&n)
13
14        if n > 0 {
15            break
16        }
17    }
18
19    fmt.Println(n, "merupakan bilangan bulat positif")
20 }
21
22
```

The terminal output shows the program being run, with the user inputting '10' and the program outputting '10 merupakan bilangan bulat positif'. A small window titled 'Nama Rayl' is also visible, containing the text 'Nama: Rayhan Ahza Widyamukti' and 'Nim: 109982500210'.

Deskripsi program

Program tersebut menggunakan perulangan tanpa batas (infinite loop) untuk memastikan pengguna hanya memasukkan bilangan bulat positif. Di dalam blok `for {}`, program meminta input melalui `fmt.Scan(&n)`. Setelah nilai dimasukkan, kondisi `if n > 0` dicek; apabila benar, pernyataan `break` dijalankan untuk keluar dari loop. Jika nilai yang dimasukkan bukan bilangan positif (misalnya nol atau negatif), loop akan terus berulang dan meminta input kembali. Setelah loop berhenti, program mencetak pesan bahwa nilai tersebut merupakan bilangan bulat positif.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var x, y int
    fmt.Println("Masukan nilai x dan y:")
    fmt.Scan(&x, &y)

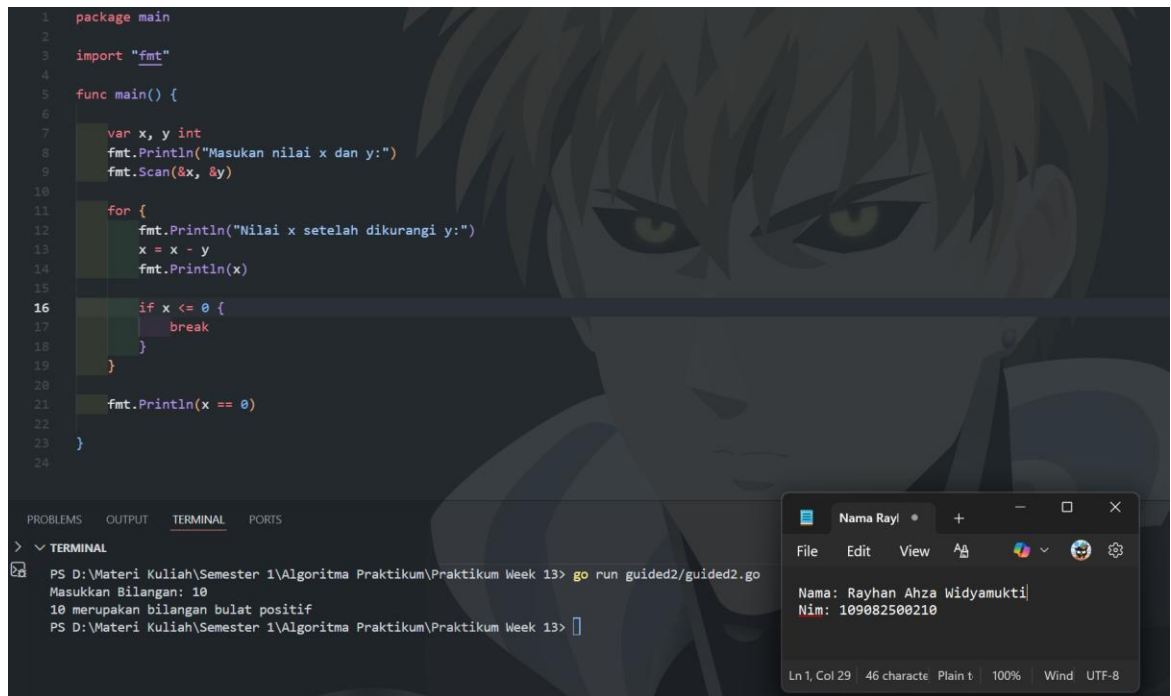
    for {
        fmt.Println("Nilai x setelah dikurangi y:")
        x = x - y
        fmt.Println(x)

        if x <= 0 {
            break
        }
    }

    fmt.Println(x == 0)

}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var x, y int
8     fmt.Println("Masukan nilai x dan y:")
9     fmt.Scan(&x, &y)
10
11     for {
12         fmt.Println("Nilai x setelah dikurangi y:")
13         x = x - y
14         fmt.Println(x)
15
16         if x <= 0 {
17             break
18         }
19     }
20
21     fmt.Println(x == 0)
22
23 }
```

The terminal output shows the program running successfully. It prompts the user to input two integers, x and y. The user enters 10 for x and 10 for y. The program then enters a loop that subtracts y from x and prints the result. The output shows the result of the subtraction (0) and the boolean value of the expression x == 0 (true).

```
PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13> go run guided2/guided2.go
Masukkan Bilangan: 10
10 merupakan bilangan bulat positif
PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13>
```

Deskripsi Program

Program tersebut meminta pengguna memasukkan dua nilai bilangan bulat, yaitu x dan y, kemudian menjalankan perulangan tanpa batas untuk mengurangi nilai x secara berulang dengan y. Setiap iterasi, program menampilkan hasil pengurangan tersebut dan memperbarui nilai x. Perulangan terus berlanjut sampai nilai x menjadi kurang dari atau sama dengan nol, yang ditandai dengan kondisi $x \leq 0$ sehingga perintah break menghentikan loop. Setelah loop selesai, program mencetak nilai boolean dari ekspresi $x == 0$, yang akan bernilai true jika nilai akhir x tepat nol, atau false jika nilai akhir x kurang dari nol.

TUGAS

Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var n int
    var hitung int = 0

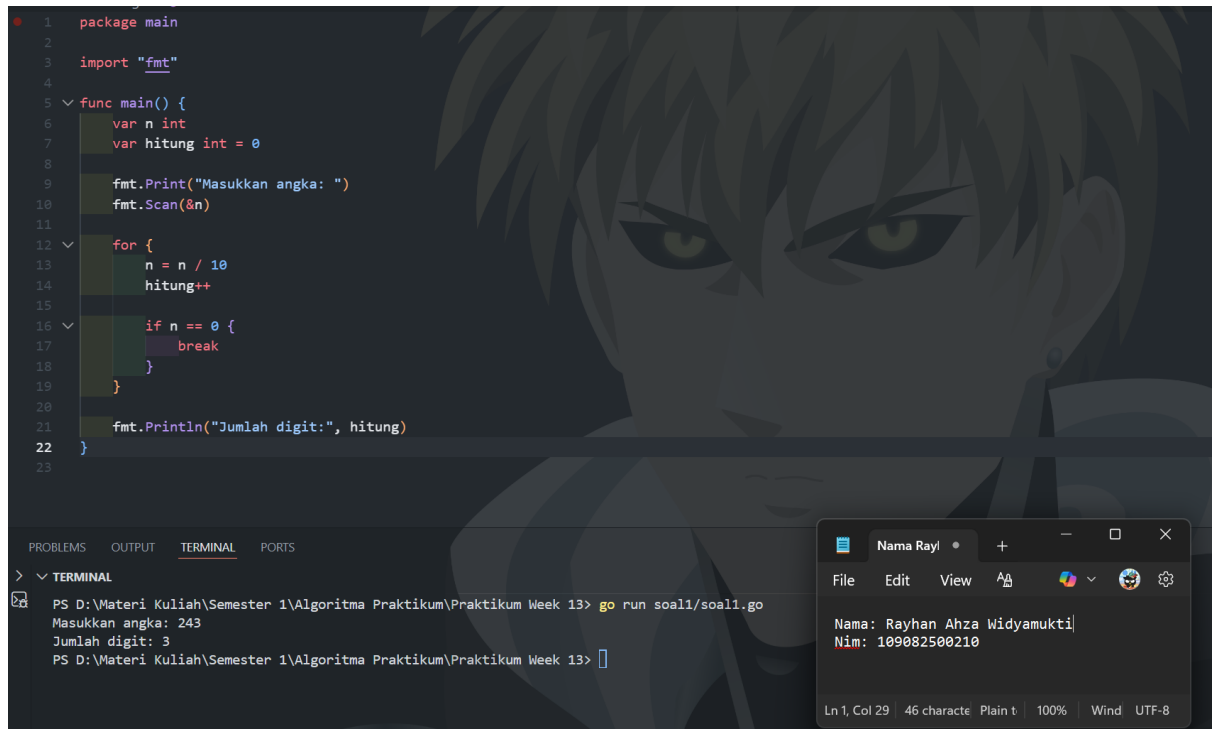
    fmt.Print("Masukkan angka: ")
    fmt.Scan(&n)

    for {
        n = n / 10
        hitung++

        if n == 0 {
            break
        }
    }

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


Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The program is designed to count the number of digits in an integer by repeatedly dividing it by 10. The terminal shows the user inputting '243' and the program outputting 'Jumlah digit: 3'.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     var hitung int = 0
8
9     fmt.Print("Masukkan angka: ")
10    fmt.Scan(&n)
11
12    for {
13        n = n / 10
14        hitung++
15
16        if n == 0 {
17            break
18        }
19    }
20
21    fmt.Println("Jumlah digit:", hitung)
22 }
23
```

TERMINAL

```
PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13> go run soal1/soal1.go
Masukkan angka: 243
Jumlah digit: 3
PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13>
```

Deskripsi program

Program tersebut digunakan untuk menghitung jumlah digit dari sebuah angka dengan memanfaatkan perulangan tanpa batas. Setelah pengguna memasukkan nilai n , program menginisialisasi variabel penghitung `hitung` dengan nilai awal 0. Di dalam loop `for {}`, nilai n dibagi 10 pada setiap iterasi untuk menghilangkan satu digit dari belakang, kemudian variabel `hitung` dinaikkan satu sebagai penanda bahwa satu digit telah dihitung. Proses ini berlangsung terus sampai nilai n menjadi 0, yang menandakan seluruh digit telah diproses, sehingga kondisi `if n == 0` terpenuhi dan loop dihentikan menggunakan `break`. Setelah loop selesai, program menampilkan total digit yang berhasil dihitung melalui nilai `hitung`.

Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var x float64
    var batas float64

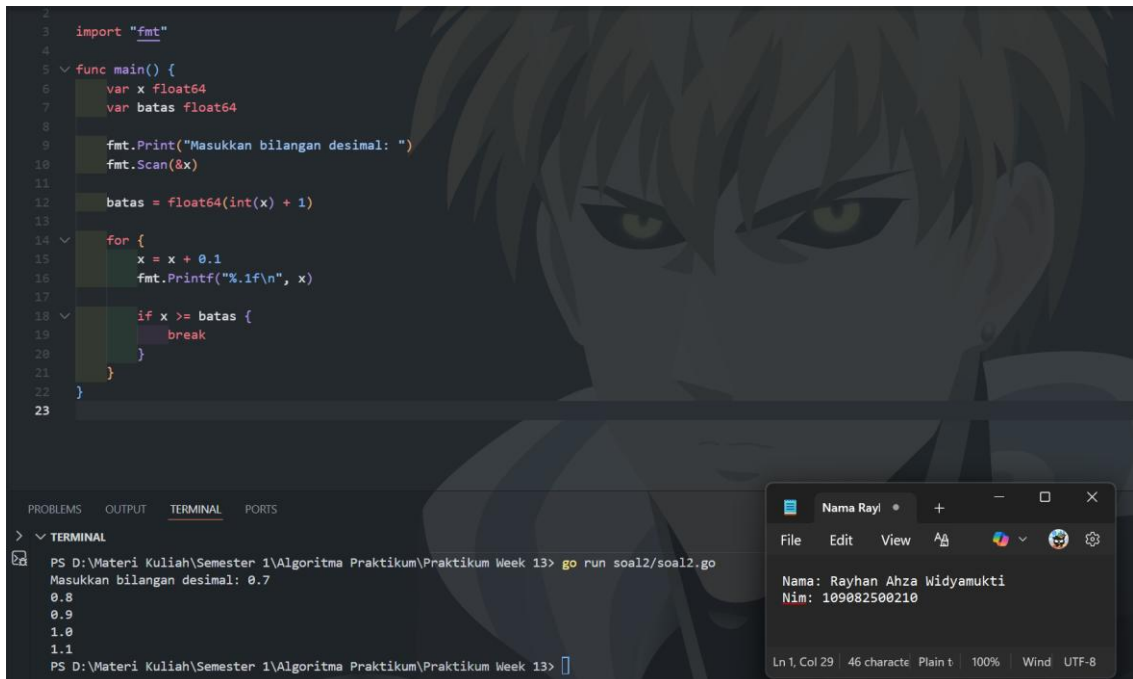
    fmt.Print("Masukkan bilangan desimal: ")
    fmt.Scan(&x)

    batas = float64(int(x) + 1)

    for {
        x = x + 0.1
        fmt.Printf("%.1f\n", x)

        if x >= batas {
            break
        }
    }
}
```

Screenshoot program

The screenshot shows a Go program in a code editor with a dark theme. The code defines a main function that prompts the user for a decimal number, calculates the next integer ceiling, and prints the values from the input up to that ceiling in increments of 0.1. The terminal window at the bottom shows the execution of the program, displaying the input '0.7' and the resulting sequence of numbers: 0.8, 0.9, 1.0, and 1.1. A small window titled 'Nama Rayl' is also visible in the bottom right corner.

```
1
2 import "fmt"
3
4
5 func main() {
6     var x float64
7     var batas float64
8
9     fmt.Print("Masukkan bilangan desimal: ")
10    fmt.Scan(&x)
11
12    batas = float64(int(x) + 1)
13
14    for {
15        x = x + 0.1
16        fmt.Printf("%.1f\n", x)
17
18        if x >= batas {
19            break
20        }
21    }
22 }
23
```

PROBLEMS OUTPUT TERMINAL PORTS

> TERMINAL

PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13> go run soal2/soal2.go

Masukkan bilangan desimal: 0.7

0.8

0.9

1.0

1.1

PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13>

Nama Rayl

File Edit View

Nama: Rayhan Ahza Widyamukti

Nim: 109082500210

Ln 1, Col 29 46 character Plain text 100% Windows UTF-8

Deskripsi program

Program tersebut berfungsi untuk menaikkan nilai desimal yang dimasukkan pengguna hingga mencapai batas bilangan bulat berikutnya. Setelah pengguna memasukkan nilai x , program menghitung nilai batas dengan mengambil bagian integer dari x lalu menambahkannya satu. Selanjutnya, program menjalankan perulangan tanpa batas, di mana pada setiap iterasi nilai x ditambah 0.1 dan kemudian dicetak dengan satu angka di belakang koma menggunakan `fmt.Printf("%.1f\n", x)`. Proses ini terus berlangsung sampai nilai x mencapai atau melampaui nilai batas, sehingga kondisi `if x >= batas` terpenuhi dan pernyataan `break` menghentikan loop. Program memastikan bahwa nilai desimal meningkat secara bertahap hingga menyentuh bilangan bulat berikutnya.

Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var target int
    var total int
    var donasi int
    var donatur int

    fmt.Print("Masukkan target donasi: ")
    fmt.Scan(&target)

    total = 0
    donatur = 0

    for {
        donatur = donatur + 1

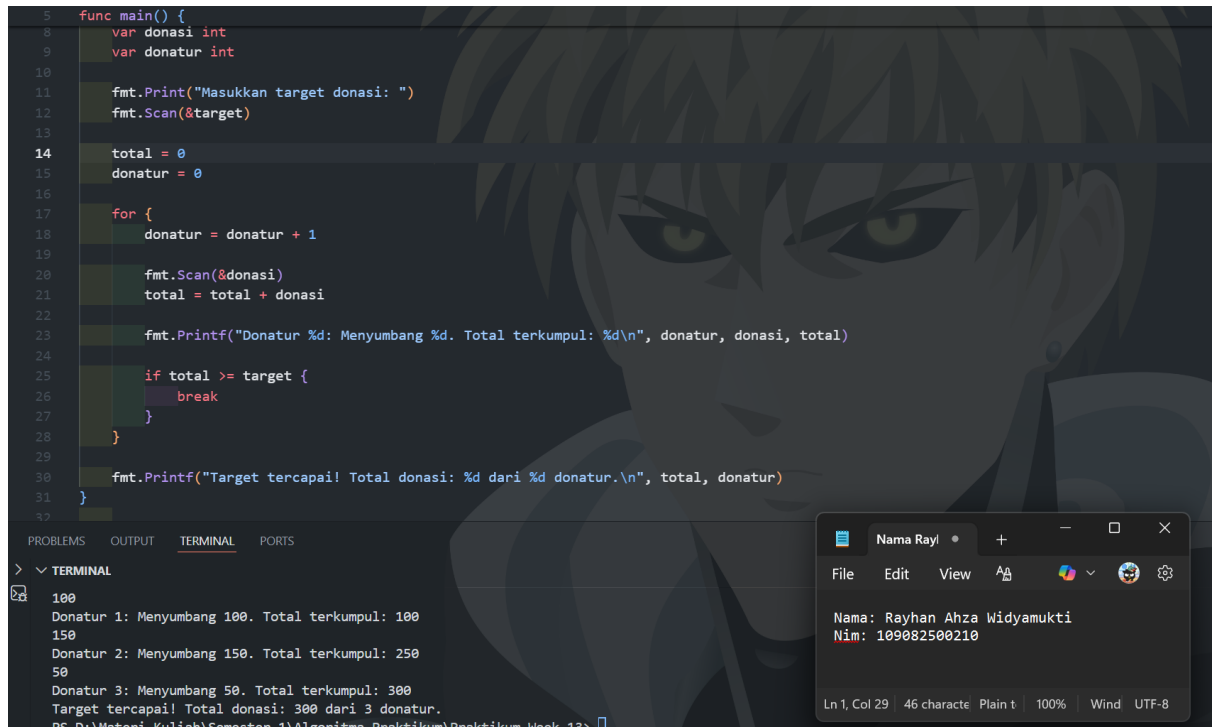
        fmt.Scan(&donasi)
        total = total + donasi

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

        if total >= target {
            break
        }
    }

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

Screenshoot program



```
5 func main() {
8     var donasi int
9     var donatur int
10
11     fmt.Println("Masukkan target donasi: ")
12     fmt.Scan(&target)
13
14     total = 0
15     donatur = 0
16
17     for {
18         donatur = donatur + 1
19
20         fmt.Scan(&donasi)
21         total = total + donasi
22
23         fmt.Printf("Donatur %d: Menyumbang %d. Total terkumpul: %d\n", donatur, donasi, total)
24
25         if total >= target {
26             break
27         }
28     }
29
30     fmt.Printf("Target tercapai! Total donasi: %d dari %d donatur.\n", total, donatur)
31 }
32
```

PROBLEMS OUTPUT **TERMINAL** PORTS

> **TERMINAL**

```
100
Donatur 1: Menyumbang 100. Total terkumpul: 100
150
Donatur 2: Menyumbang 150. Total terkumpul: 250
50
Donatur 3: Menyumbang 50. Total terkumpul: 300
Target tercapai! Total donasi: 300 dari 3 donatur.
PS D:\Materi Kuliah\Semester 1\Algoritma Praktikum\Praktikum Week 13>
```

Nama Rayl

File Edit View

Nama: Rayhan Ahza Widyamukti
Nim: 109082500210

Ln 1, Col 29 46 character Plain t 100% Wind UTF-8

Deskripsi program

Program tersebut digunakan untuk menghitung akumulasi donasi hingga mencapai target yang ditentukan. Setelah pengguna memasukkan nilai target, program menginisialisasi total dan donatur dengan nilai awal 0. Di dalam perulangan tanpa batas, setiap iterasi dianggap sebagai satu donatur baru sehingga variabel donatur ditambah 1. Program kemudian membaca nilai donasi melalui `fmt.Scan(&donasi)` dan menambahkannya ke variabel total. Setelah itu, program menampilkan informasi donatur, jumlah yang disumbangkan, serta total donasi yang telah terkumpul. Perulangan berlanjut sampai total donasi mencapai atau melampaui nilai target; pada kondisi tersebut, perintah `break` menghentikan loop. Setelah loop selesai, program menutup dengan menampilkan total donasi yang terkumpul beserta jumlah donatur yang berpartisipasi hingga target tercapai.