

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 13
REPEAT-UNTIL



Disusun oleh:

AKHSAN SABILI

109082500062

S1IF-13-04

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 p int

    var q string

    fmt.Scan(&q, &p)

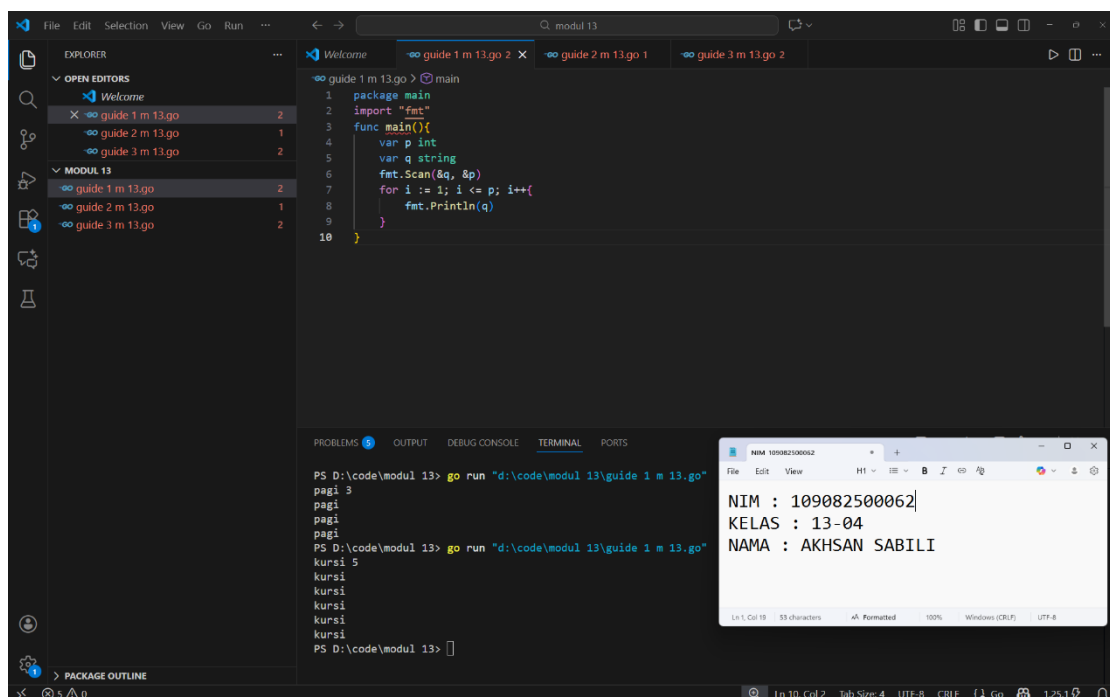
    for i := 1; i <= p; i++){

        fmt.Println(q)

    }

}
```

Screenshoot program



Deskripsi program

Pada program diatas terdapat dua variable utama, yaitu p dengan tipe interger dan q dengan tipe string. Terdapat Scan untuk membaca inputan dari pengguna dan memasukkannya ke variable q dan p. Lalu terdapat perulangan dengan inisiasi nilai awal adalah satu dan akan berhenti jika nilainya sama dengan p. Lalu perulangan akan akan menampilkan inputan q yang berupa string sebanyak p kali

2. Guided 2

Source Code

```
package main

import "fmt"

func main(){

    var n int

    var x bool

    for x == false{

        fmt.Scan(&n)

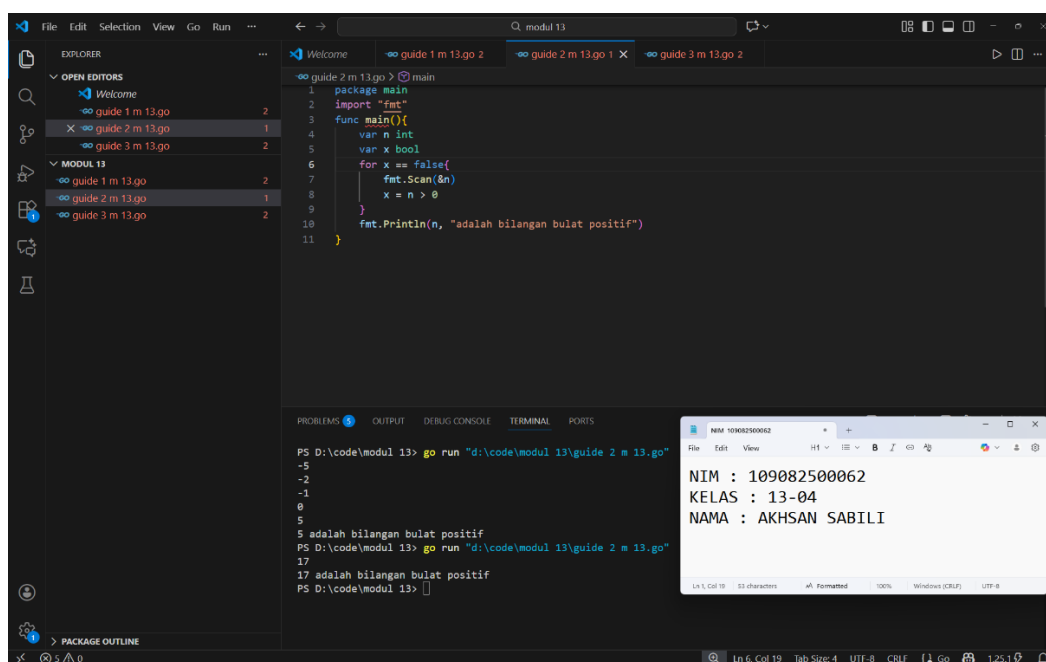
        x = n > 0

    }

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

}
```

Screenshoot program



Deskripsi program

Pada program diatas terdapat dua variable utama, yaitu n dengan tipe interger dan x dengan tipe Boolean. Lalu terdapat perulangan dengan nilai x adalah false yang berarti selama x bernilai false maka perulangan akan terus berjalan. Lalu terdapat Scan untuk membaca inputan pengguna. Scan dalam perulangan diatas berarti jika x masih false, maka program akan terus meminta inputan. Lalu terdapat x = n > 0 berarti program akan mengecek inputan, jika inputan lebih besar dari 0 maka x akan bernilai true dan perulangan akan berhenti dan program akan mencetak n dan pesan melalui Println

3. Guided 3

Source Code

```
package main

import "fmt"

func main(){
    var x, y int
    var kond bool
    fmt.Scan(&x, &y)
    for kond == false{
        x-= y
        fmt.Println(x)
        kond = x <= 0
    }
    fmt.Println(x == 0)
}
```

Screenshoot program

The screenshot displays a Go IDE interface. The Explorer panel on the left shows the project structure with files like 'Welcome', 'guide 1 m 13.go', 'guide 2 m 13.go', and 'guide 3 m 13.go'. The main editor shows the source code for 'guide 3 m 13.go', which is identical to the code provided in the 'Source Code' section. The bottom panel is split into 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL', and 'PORTS'. The 'TERMINAL' tab is active, showing the execution of the program. The output shows the program running successfully, with the final output being 'true'.

```
PS D:\code\modul 13> go run "d:\code\modul 13\guide 3 m 13.go"
5 2
3
1
-1
false
PS D:\code\modul 13> go run "d:\code\modul 13\guide 3 m 13.go"
15 3
12
9
6
3
0
true
PS D:\code\modul 13> go run "d:\code\modul 13\guide 3 m 13.go"
25 5
20
15
10
5
0
true
PS D:\code\modul 13>
```

Deskripsi program

Pada program diatas terdapat tiga variable utama, yaitu x dan y dengan tipe interger dan kond dengan tipe Boolean. Lalu terdapat Scan yang berfungsi untuk membaca inputan dari pengguna dan menyimpannya pada variable x dan y. Terdapat perulangan dengan nilai awal kond = false yang berarti jika kond masih false, maka perulangan akan terus berjalan.

Terdapat `x -= y` yang berarti x akan terus dikurangi dengan y dan hasilnya akan disimpan di variable x. Terdapat `Println(x)` yang berfungsi untuk menampilkan nilai x setelah dikurangi y. Lalu terdapat `kond = x <= 0` yang berfungsi untuk mengecek pengurangan x dengan y, apabila nilai x kurang atau sama dengan 0 maka perulangan akan berhenti. Lalu `Println(x==0)` digunakan sebagai output untuk menampilkan apakah x sama dengan 0 atau tidak. Jika x sama dengan 0 maka output akan menampilkan true, begitupun sebaliknya

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main(){

    var n int

    fmt.Scan(&n)

    hasil := 0

    for kons := true; !kons;{

        hasil++

        n /= 10

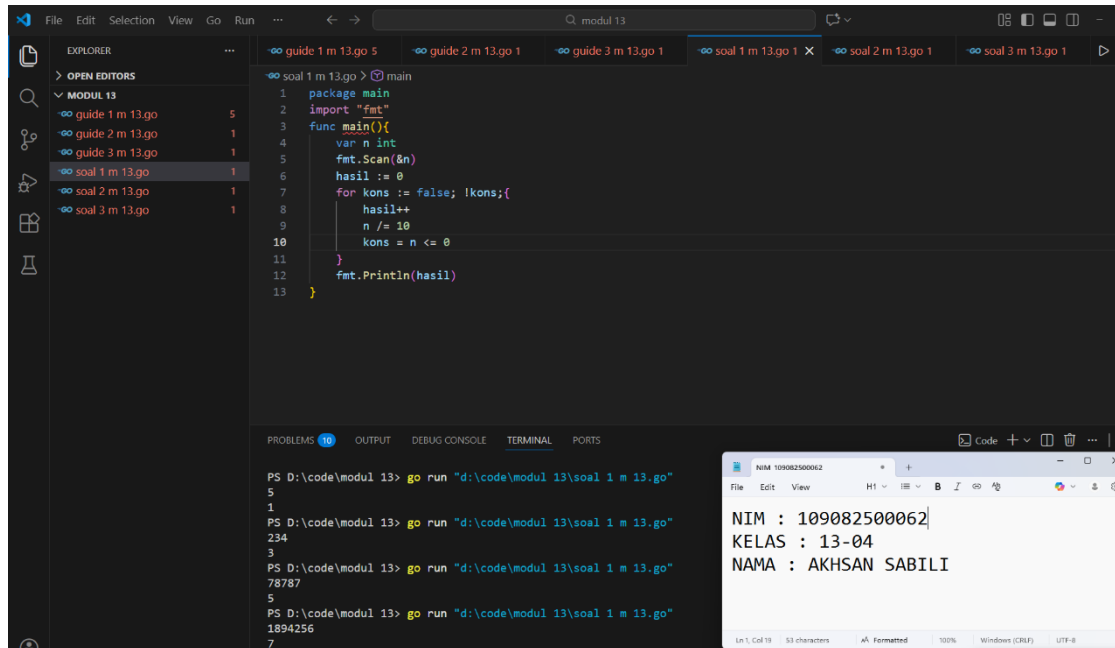
        kons = n <= 0

    }

    fmt.Println(hasil)

}
```

Screenshoot program



Deskripsi program

Pada program diatas terdapat satu variable utama, yaitu `n` dengan tipe interger. Lalu terdapat `Scan` untuk membaca inputan dari pengguna dan menyimpannya ke variable `n`. Lalu terdapat variable `hasil` dengan inisialisasi nilai awal 0. Terdapat perulangan `kons := false` yang berarti perulangan akan terus berjalan selama kondisi masih `false`. Lalu `hasil++` berarti menambah satu disetiap perulangan yang berjalan. Lalu `n /= 10` berfungsi menghapus satu angka paling belakang disetiap perulangannya. Terdapat `kons` yang berarti apabila `n` lebih kecil atau sama dengan 0 maka perulangan akan berhenti. Setelah perulangan selesai, program akan menampilkan output berupa jumlah angka yang dihitung.

2. Tugas 2

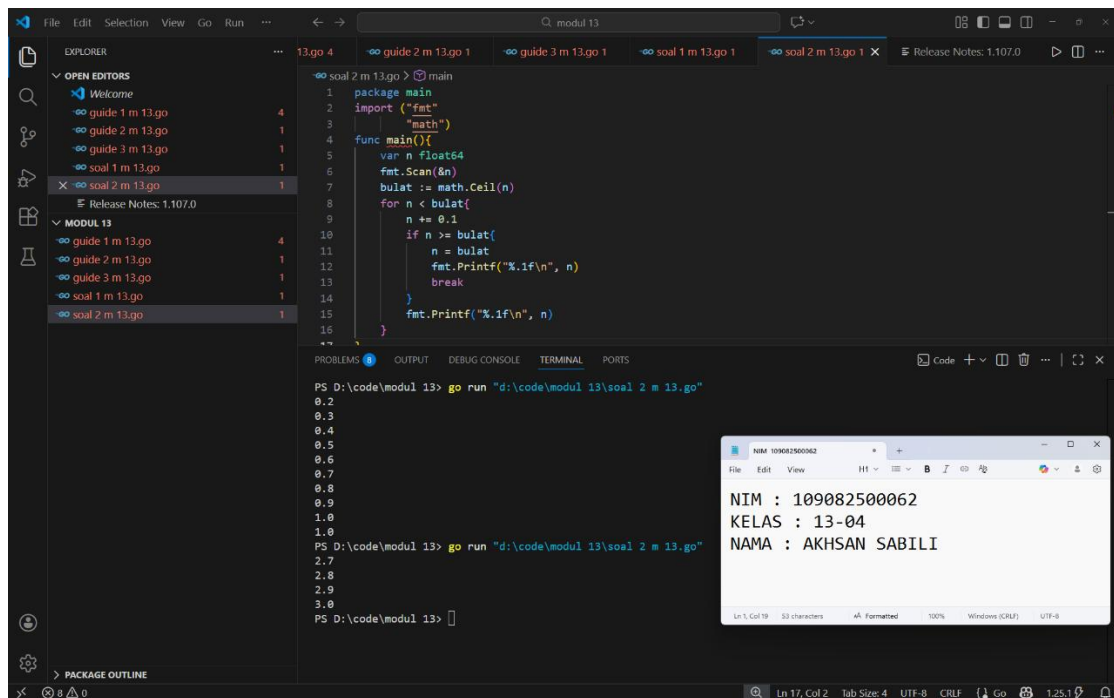
Source code

```
package main

import ("fmt"
        "math")

func main(){
    var n float64
    fmt.Scan(&n)
    bulat := math.Ceil(n)
    for n < bulat{
        n += 0.1
        if n >= bulat{
            n = bulat
            fmt.Printf("%.1f\n", n)
            break
        }
        fmt.Printf("%.1f\n", n)
    }
}
```

Screenshoot program



Deskripsi program

Pada program diatas terdapat satu variable utama, yaitu n dengan tipe float atau bilangan rill. Lalu Scan digunakan untuk membaca inputan dari pengguna dan menyimpannya ke dalam variable n. Berikutnya terdapat math.Cil yang berfungsi untuk membulatkan angka keatas, missal 2.3 menjadi 3.

Terdapat perulangan yang akan berjalan jika n lebih kecil dari bilangan yang sudah dibulatkan dengan math.Cil. Lalu `n += 0.1` yang berfungsi untuk terus menambahkan n dengan 0.1. Terdapat if yang apabila n lebih besar atau sama dengan bilangan bulat, maka n sama dengan bilangan bulat dan akan menampilkan n serta break yang membuat perulanganberhenti. Lalu terdapat Print di dalam for tetapi diluar if untuk menampilkan perulangan hingga selesai.

3. Tugas 3

Source code

```
package main

import "fmt"

func main(){

    var n, m int

    fmt.Scan(&n)

    hasil := 0

    hasil2 := 0
```



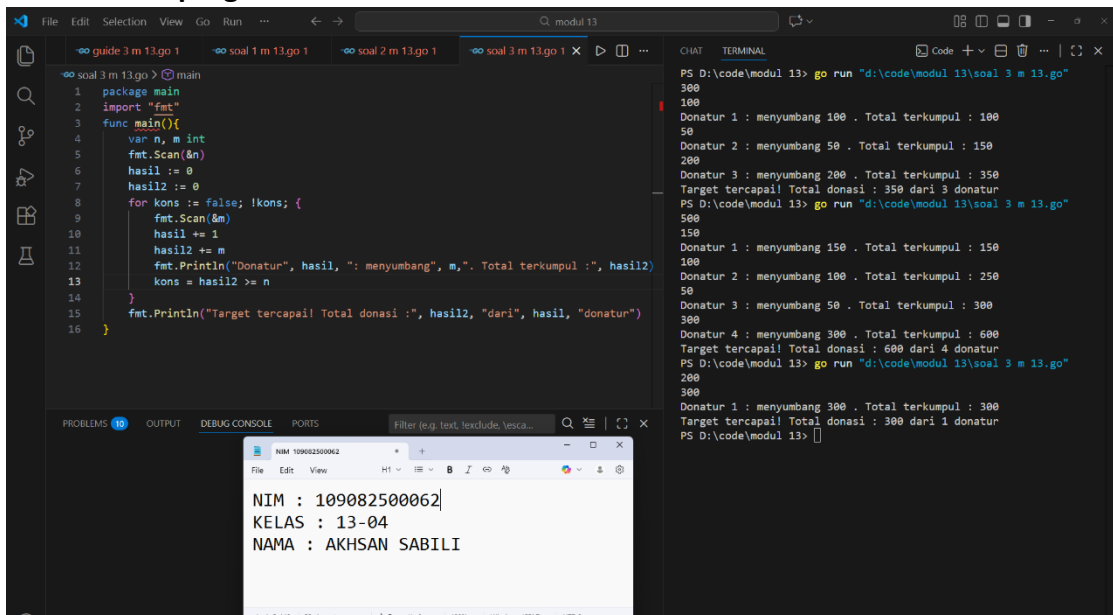
```

        for kons := false; !kons; {
            fmt.Scan(&m)
            hasil += 1
            hasil2 += m
            fmt.Println("Donatur", hasil, ": menyumbang",
m, ". Total terkumpul :", hasil2)
            kons = hasil >= n
        }

        fmt.Println("Target tercapai! Total donasi :",
hasil2, "dari", hasil, "donatur")
    }
}

```

Screenshoot program



Deskripsi program

Pada program diatas terdapat dua variable utama, yaitu n dan m dengan tipe interger. Lalu scan n untuk membaca inputan dari pengguna dan menyimpannya ke variable n. Terdapat dua variable tambahan, yaitu hasil dan hasil2 dengan inisialisasi nilai awal 0.

Terdapat perulangan yang akan terus berjalan hingga kons bernilai true. Lalu Scan m yang akan membaca inputan berupa donasi dari donatur secara terus menerus hingga perulangan berhenti. Terdapat hasil += 1 yang berarti menambah donator di setiap perulangannya dan hasil2 += m yang berarti menambahkan jumlah yang di donasikan secara terus menerus.

Terdapat `Println` untuk menampilkan informasi tentang donatur dan total donasi sementara. Lalu `kons = hasil2 >= n` berfungsi untuk mengecek apakah hasil donasi melebihi target, dan apabila sudah melebihi maka perulangan akan berhenti. Lalu `Println` terakhir berfungsi untuk menampilkan target yang tercapai ke layer.