

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 13
REPEAT - UNTIL**



Disusun oleh:
BENING PUTRI NARESWARI SUKARNO
109082500211
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 perulangan int

    var hitung int = 0

    fmt.Scan(&kata)

    fmt.Scan(&perulangan)

    for hitung < perulangan {

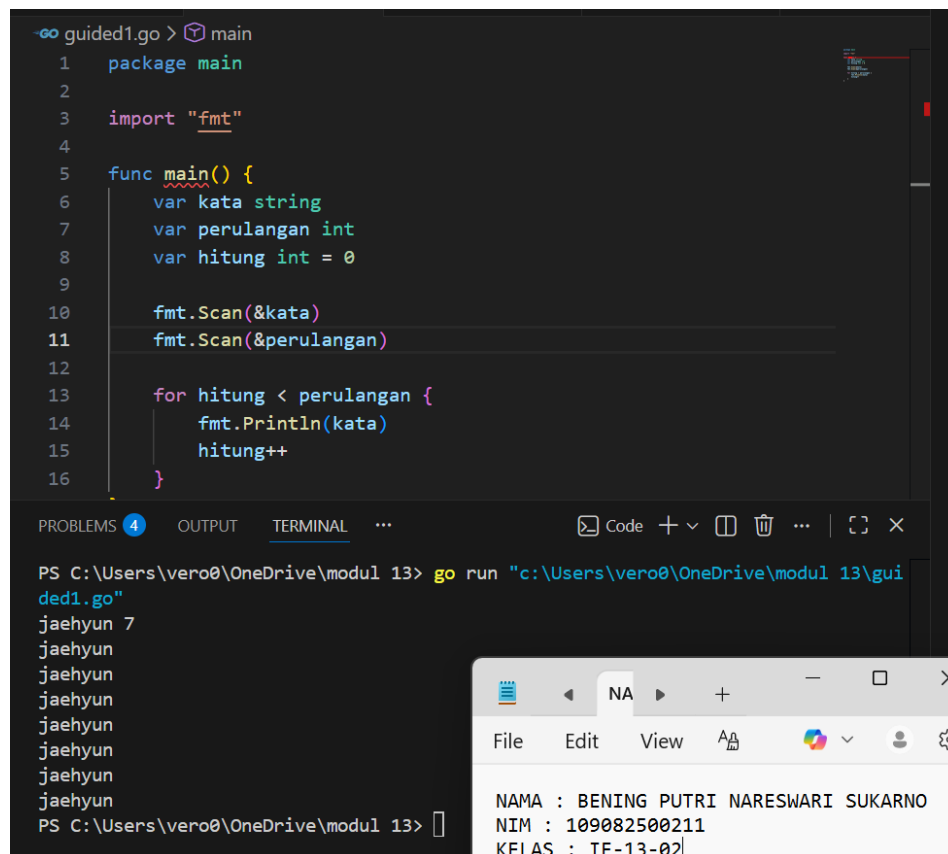
        fmt.Println(kata)

        hitung++

    }

}
```

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     var kata string
7     var perulangan int
8     var hitung int = 0
9
10    fmt.Scan(&kata)
11    fmt.Scan(&perulangan)
12
13    for hitung < perulangan {
14        fmt.Println(kata)
15        hitung++
16    }
```

The terminal output shows the program running and printing the word "jaehyun" seven times, corresponding to the input value of 7 for the loop variable "perulangan".

```
PS C:\Users\vero0\OneDrive\modul 13> go run "c:\Users\vero0\OneDrive\modul 13\gui
ded1.go"
jaehyun 7
jaehyun
jaehyun
jaehyun
jaehyun
jaehyun
jaehyun
jaehyun
PS C:\Users\vero0\OneDrive\modul 13>
```

Deskripsi program

Program membaca satu kata (string) dan satu angka (integer) dari input pengguna, dan kemudian mencetak kata tersebut sebanyak angka yang dimasukkan.

1. Program menunggu pengguna memasukkan inputan
 - (fmt.Scan(&kata)): Menerima satu kata (string)
 - (fmt.Scan(&perulangan)): Menerima satu bilangan bulat (integer).
2. Program menjalankan blok kode perulangan (for) dengan kondisi:
Kondisi: Perulangan akan terus berjalan selama nilai variabel hitung kurang dari nilai variabel perulangan** (hitung < perulangan').
Aksi: Di setiap iterasi (putaran) perulangan: Program mencetak nilai dari variabel kata ke terminal (fmt.Println(kata)). Nilai dari variabel hitung ditambahkan satu (hitung++).

Proses ini akan berlanjut hingga nilai hitung sama dengan nilai perulangan, dan perulangan pun berhenti.

2. Guided 2

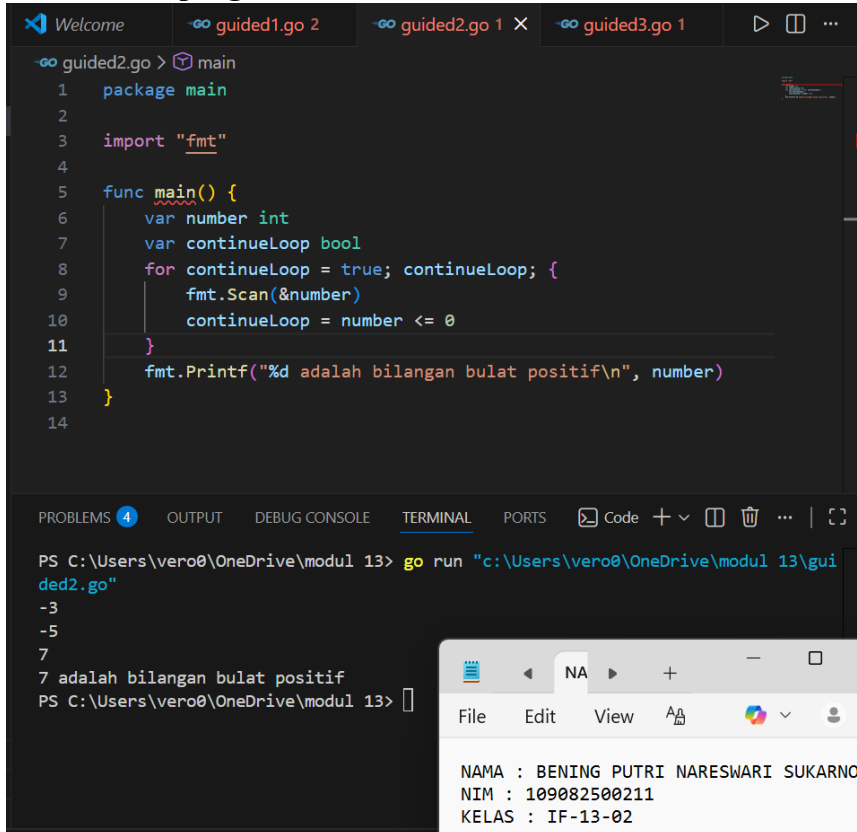
Source Code

```
package main

import "fmt"

func main() {
    var number int
    var continueLoop bool
    for continueLoop = true; continueLoop; {
        fmt.Scan(&number)
        continueLoop = number <= 0
    }
    fmt.Printf("%d adalah bilangan bulat positif\n", number)
}
```

Screenshoot program



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

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var number int
7     var continueLoop bool
8     for continueLoop = true; continueLoop; {
9         fmt.Scan(&number)
10        continueLoop = number <= 0
11    }
12    fmt.Printf("%d adalah bilangan bulat positif\n", number)
13 }
14
```

The terminal output shows the program being run and the user inputting numbers -3, -5, and 7. The program stops when 7 is entered and prints the message:

```
PS C:\Users\vero0\OneDrive\modul 13> go run "c:\Users\vero0\OneDrive\modul 13\guided2.go"
-3
-5
7
7 adalah bilangan bulat positif
PS C:\Users\vero0\OneDrive\modul 13>
```

A small window titled 'NA' is overlaid on the bottom right, containing the following text:

```
NAMA : BENING PUTRI NARESWARI SUKARNO
NIM : 109082500211
KELAS : IF-13-02
```

Deskripsi program

Program tersebut berfungsi untuk menerima masukan angka bulat dari pengguna secara berulang-ulang sampai pengguna menginputkan angka 0 (nol) atau bilangan negatif. Setelah perulangan berhenti, program akan mencetak angka bulat positif terakhir yang valid yang dimasukkan oleh pengguna.

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)

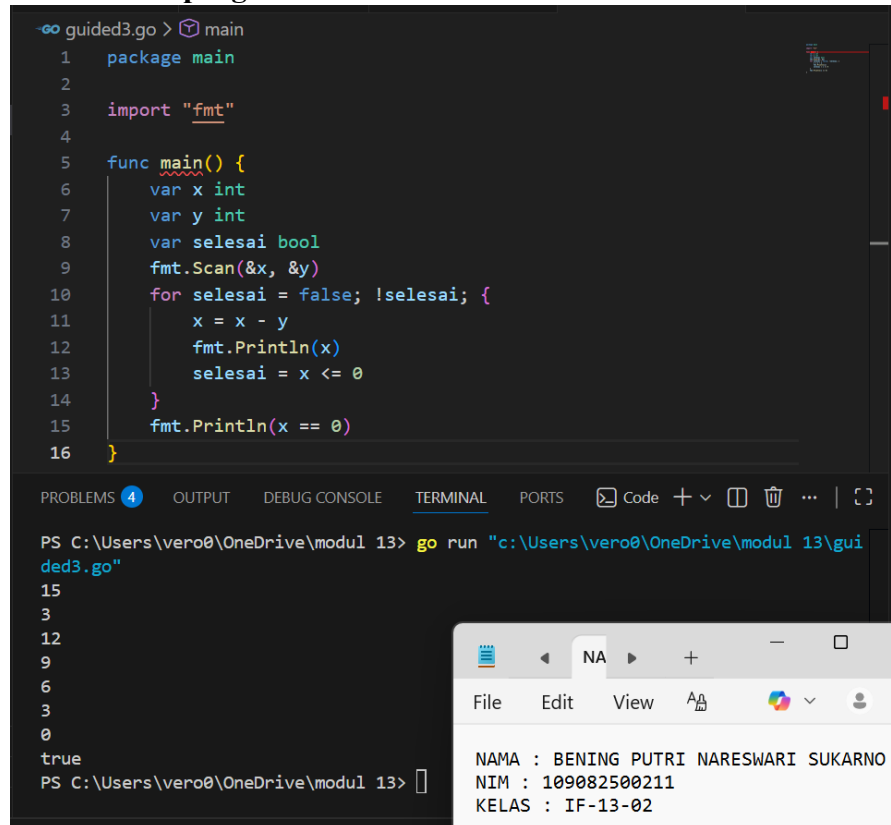
        selesai = x <= 0

    }

    fmt.Println(x == 0)

}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The code defines a `main` function that takes two integers `x` and `y` as input. It enters a `for` loop that continues as long as `selesai` is `false` and `x` is greater than or equal to 0. Inside the loop, `x` is decremented by `y`, and the current value of `x` is printed. The loop ends when `x` is less than 0, at which point `selesai` becomes `true`. Finally, the program prints the value of `x == 0`.

```
guided3.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x int
7     var y int
8     var selesai bool
9     fmt.Scan(&x, &y)
10    for selesai = false; !selesai; {
11        x = x - y
12        fmt.Println(x)
13        selesai = x <= 0
14    }
15    fmt.Println(x == 0)
16 }
```

Terminal output:

```
PS C:\Users\vero0\OneDrive\modul 13> go run "c:\Users\vero0\OneDrive\modul 13\guided3.go"
15
3
12
9
6
3
0
true
PS C:\Users\vero0\OneDrive\modul 13>
```

NA

File Edit View

NAMA : BENING PUTRI NARESWARE SUKARNO
NIM : 109082500211
KELAS : IF-13-02

Deskripsi program

Program tersebut berfungsi untuk melakukan operasi pengurangan berulang (pengurangan bertahap) terhadap dua bilangan bulat positif yang diinputkan pengguna. Proses ini pada dasarnya adalah simulasi dari modulo (sisa bagi). Hasil akhir dari program adalah menentukan apakah pengurangan berulang tersebut menghasilkan nilai 0 (nol).

1. Program menunggu dua masukan dari pengguna secara bersamaan (`x` dan `y`).
2. Program menjalankan blok perulangan (`for`) dengan kondisi: Perulangan akan terus berjalan selama `selesai` bernilai `false` (`!selesai`).
 - Pengurangan (`x = x - y`): Nilai `x` dikurangi dengan nilai `y`, dan hasilnya disimpan kembali ke `x`. Ini adalah operasi pengurangan bertahap.
 - Mencetak Nilai Baru `X` (`fmt.Println(x)`): Nilai `x` setelah dikurangi dicetak ke terminal, menunjukkan sisa pengurangan saat ini.
 - Nilai variabel `selesai` diperbarui berdasarkan apakah nilai `x` saat ini kurang dari atau sama dengan nol (`x <= 0`).
 - Jika `x` masih positif, `selesai` tetap `false`, dan perulangan berlanjut.
 - Jika `x` menjadi nol atau negatif, `selesai` menjadi `true`, dan perulangan berhenti.
3. Setelah perulangan berhenti, program mencetak nilai boolean dari ekspresi `x == 0`.
 - Jika pengurangan berulang tepat menyisakan 0, maka hasilnya adalah `true`.
 - Jika pengurangan berulang menyisakan bilangan negatif (karena `y` terlalu besar), maka hasilnya adalah `false`.

TUGAS

1. Tugas 1

Source code

```
package main

import (
    "fmt"
)

func main() {
    var number int
    var digit int = 0

    fmt.Print("Masukkan bilangan bulat positif: ")
    fmt.Scan(&number)

    if number == 0 {
        digit = 1
    }

    for number > 0 {
        number = number / 10

        digit++
    }
}
```



```
fmt.Printf("Banyaknya digit adalah: %d\n", digit)
}
```

Screenshoot program

```

1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var number int
9     var digit int = 0
10
11     fmt.Print("Masukkan bilangan bulat positif: ")
12     fmt.Scan(&number)
13
14     if number == 0 {
15         digit = 1
16     }
17
18     for number > 0 {
19         number = number / 10
20     }
21 }

```

PROBLEMS 6 OUTPUT TERMINAL ...

PS C:\Users\vero0\OneDrive\modul 13> go run "c:\Users\vero0\OneDrive\modul 13\soal11.go"

Masukkan bilangan bulat positif: 3456

Banyaknya digit adalah: 4

PS C:\Users\vero0\OneDrive\modul 13> |

File Edit View

NAMA : BENING PUTRI NARESWARI SUKARN
NIM : 109082500211

Deskripsi program

Program dirancang untuk menghitung jumlah digit (angka) yang terdapat pada sebuah bilangan bulat positif yang dimasukkan oleh pengguna.

Program ini menggunakan prinsip pembagian integer (integer division) berulang dengan basis 10 untuk menghitung digit.

1. Program memulai dengan mendeklarasikan dua variabel: number (untuk menyimpan masukan) dan digit (yang diatur ke 0 dan berfungsi sebagai penghitung).
2. Program meminta pengguna memasukkan bilangan dan menyimpannya di variabel number.
3. Ada pemeriksaan awal (if number == 0) untuk memastikan bahwa jika pengguna memasukkan 0, penghitung digit segera diatur ke 1, karena angka 0 dianggap memiliki satu digit.
4. Program masuk ke loop for number > 0. Perulangan ini akan terus berjalan selama nilai number masih lebih besar dari nol.
 - Number dibagi dengan 10 (number = number / 10). Karena ini adalah pembagian integer, satu digit paling kanan akan terpotong atau dihilangkan.

- Variabel digit dinaikkan (+1) (digit++).
 - Proses ini diulang sampai number menjadi 0, menunjukkan semua digit telah dihilangkan dan dihitung.
5. Keluaran: Setelah loop selesai, program mencetak nilai akhir dari variabel digit, yang merupakan jumlah total digit dari masukan awal.

2. Tugas 2

Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var input float64
    var nilai float64

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

    nilai = input
    batas := math.Ceil(input)

    for {
        nilai = nilai + 0.1
        fmt.Printf("%.1f\n", nilai)
        if nilai >= batas {
            break
        }
    }
}
```

}

Screenshot program

```

6 func main() {
7     var input float64
8     var nilai float64
9
10    fmt.Print("Masukkan bilangan desimal: ")
11    fmt.Scan(&input)
12
13    nilai = input
14    batas := math.Ceil(input)
15
16    for {
17        nilai = nilai + 0.1
18
19        fmt.Printf("%.1f\n", nilai)
20
21        if nilai >= batas {
22            break
23        }
24    }
25 }

```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS Code + View [] [] [] [] []

Masukkan bilangan desimal: 0.4

0.5

0.6

0.7

0.8

0.9

1.0

1.1

PS C:\Users\vero0\OneDrive\modul 13>

NA

File Edit View [] [] [] [] []

NAMA : BENING PUTRI NARESWARI SUKARNO

NTM : 100082500211

Deskripsi program

Program Go ini berfungsi untuk menghasilkan deret bilangan desimal secara berulang yang dimulai tepat setelah masukan pengguna, dengan menambahkan 0.1 pada setiap langkahnya. Perulangan ini akan terus berlangsung sampai bilangan yang dihasilkan mencapai atau melewati batas bilangan bulat terdekat di atasnya (ceiling).

1. Program menerima bilangan desimal dari pengguna dan menyimpannya di input.
2. Variabel batas dihitung menggunakan fungsi `math.Ceil(input)`, yang mengambil nilai pembulatan ke atas (misalnya, 0.4 menjadi 1.0, 2.7 menjadi 3.0).
3. Program masuk ke loop tak terbatas (`for {}`).
 - Aksi (Penambahan & Cetak): Di setiap iterasi, program:
Menambahkan 0.1 ke nilai (`nilai nilai + 0.1`).
Mencetak nilai baru tersebut ke terminal.
 - Kondisi Berhenti (`if...break`): Setelah pencetakan, program memeriksa apakah nilai saat ini sudah lebih besar atau sama dengan batas (`if nilai >= batas`). Jika ya, pernyataan `break` akan menghentikan seluruh perulangan.

3. Tugas 3

Source code

```
package main

import (
    "fmt"
)

func main() {
    var target int
    var donasi int
    var totalDonasi int = 0
    var jumlahDonatur int = 0

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

    for {
        fmt.Printf("Masukkan donasi Donatur %d: ",
            jumlahDonatur+1)
        fmt.Scan(&donasi)

        totalDonasi += donasi
        jumlahDonatur++

        fmt.Printf("Donatur %d: Menyumbang %d. Total
            terkumpul: %d\n", jumlahDonatur, donasi, totalDonasi)
        if totalDonasi >= target {
            break
        }
    }

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

```
soal3.go > ...
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var target int
9     var donasi int
10    var totalDonasi int = 0
11    var jumlahDonatur int = 0
12
13    fmt.Print("Masukkan target donasi: ")
14    fmt.Scan(&target)
15
16    for {
17        fmt.Printf("Masukkan donasi Donatur %d: ", jumlahDonatur+
18            1)
```

PROBLEMS 10 OUTPUT TERMINAL ...

Masukkan target donasi: 700
Masukkan donasi Donatur 1: 370
Donatur 1: Menyumbang 370. Total terkumpul: 370
Masukkan donasi Donatur 2: 200
Donatur 2: Menyumbang 200. Total terkumpul: 570
Masukkan donasi Donatur 3: 210
Donatur 3: Menyumbang 210. Total terkumpul: 780
Target tercapai! Total donasi: 780 dari 3 donatur.
PS C:\Users\vero0\OneDrive\modul 13>

NA +

File Edit View

NAMA : BENING PUTRI NARESWARI
NIM : 109082500211

Program ini untuk mensimulasikan pengumpulan donasi hingga target tertentu tercapai. Program ini meminta target donasi di awal, kemudian secara berulang meminta masukan donasi dari setiap donatur hingga total donasi terkumpul mencapai atau melebihi target yang telah ditetapkan.

1. Program memulai dengan menerima dan menyimpan nilai target donasi dari pengguna.
2. Perulangan Donasi (Repeat-Until): Program menjalankan loop yang berulang untuk:
 - Menerima donasi dari setiap donatur secara bergantian.
 - Mengakumulasi donasi ke dalam totalDonasi dan mencatat jumlahDonatur.
 - Mencetak laporan real-time dari setiap sumbangan yang masuk.
3. Loop ini akan berhenti secara otomatis (break) segera setelah totalDonasi mencapai atau melebihi nilai target.
4. Program mengakhiri eksekusi dengan mencetak ringkasan final yang memuat total donasi yang berhasil terkumpul dan jumlah donatur yang berkontribusi.