

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL 13

REPEAT-UNTIL



Disusun oleh:

NAYAKA FARROS RIANDRA

109082500055

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

**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.Scan(&word, &repetitions)

    counter := 0

    for done := false; !done; {
        fmt.Println(word)
        counter++
        done = (counter >= repetitions)
    }
}
```

Screenshot program

The screenshot shows a Go development environment with the following details:

- File Explorer:** Shows project structure under "PRAKTIKUM ALGORITMA".
 - minggu 11: Nayaka farro...
 - minggu 12: Nayaka Farr...
 - LAPRAK MODUL 12_S1I...
 - LAPRAK MODUL 12_S1I...
 - soal1.go
 - soal2.go
 - soal3.go
 - minggu 13:
 - guide1.go (selected)
 - guide2.go
 - guide3.go
 - LAPRAK MODUL 12_S1I...
 - tugas1.go
 - tugas2.go
 - tugas3.go
 - modul 3
 - modul 5-6
 - modul 7
 - soal uts
 - GUIDED.go
 - GUIDED2.go
 - LATSOI MODUL 12 (1).pdf
 - Materi Alpro1.pdf
 - MODUL 12 (5).pdf
 - soal B.docx
 - Tipe Soal_Nama_NIM_EV...
 - tugas2.go
 - tugas3.go
 - tugas4.go
- Code Editor:** Displays guide1.go with a syntax error highlight at line 5: "func main() { main redeclared in this block (see details)".

```
1 package main
2
3 import "fmt"
4
5 func main() { main redeclared in this block (see details)
6     var word string
7     var repetitions int
8
9     fmt.Scan(&word, &repetitions)
10
11     counter := 0
12
13     for done := false; !done; {
14         fmt.Println(word)
15         counter++
16         done = (counter >= repetitions)
17     }
18 }
```
- Terminal:** Shows command-line output from running the program.

```
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\guide1.go"
pagi
pagi
pagi
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\guide1.go"
kursi
kursi
kursi
kursi
kursi
```
- Output Window:** Displays student information.

File	Edit	View	A&A	...
NAMA :	NAYAKA FARROS RIANDRA			
KELAS :	S1IF-13-07			
NIM :	109082500055			

Deskripsi program

Program ini merupakan aplikasi sederhana yang ditulis menggunakan bahasa pemrograman Go untuk mencetak sebuah kata secara berulang-ulang sesuai dengan jumlah yang diinginkan oleh pengguna. Prosesnya dimulai dengan mendeklarasikan variabel bertipe string untuk menyimpan kata dan variabel integer untuk jumlah repetisi, kemudian program meminta input dari pengguna melalui terminal.

Setelah mendapatkan input, program mengeksekusi struktur perulangan for yang dikendalikan oleh logika boolean dan sebuah variabel penghitung (counter) yang dimulai dari nol. Di dalam setiap putaran perulangan tersebut, program menampilkan kata ke layar lalu menambahkan nilai penghitung satu per satu. Proses ini akan terus berjalan dan baru berhenti secara otomatis ketika nilai penghitung sudah mencapai jumlah repetisi yang ditentukan sebelumnya.

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

```
File Edit Selection View ... ← → Q Praktikum Algoritma
EXPLORER ... main.go guide3.go 1 tugas1.go 1 tugas2.go 1 tugas3.go 1 guidet.go 6 guide2.go 1 main2.go main3.go ...
PRAKTIKUM ALGORITMA minggu 11 Nayaka farro...
└ minggu 12 Nayaka Far...
    └ guide1.go
        └ guide2.go
        └ guide3.go
    LAPRAK MODUL 12_S1I...
    LAPRAK MODUL 12_S1I...
    └ soal1.go
    └ soal2.go
    └ soal3.go
    minggu 13
        └ guide1.go
            └ guide2.go
            └ guide3.go
    LAPRAK MODUL 12_S1I...
    tempCodeRunnerFile_1
    tugas1.go
    tugas2.go
    tugas3.go
    modul 3
    modul 5-6
    modul 7
    soal uts
GUIDED.go 4
GUIDE2.go 1
LAPRAK MODUL 11_S1I...
LATSOI MODUL 12 (1).pdf
Materi Alpro1.pdf
MODUL 12 (5).pdf
soal 8.docx
Tipe Soal_Nama_NIM_EVA...
tugas2.go 1
tugas3.go 1
OUTLINE
TIMELINE
GO
PACKAGE OUTLINE
minggu 13 > guide1.go > main
package main
import "fmt"
func main() {    main redeclared in this block
    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)
}
PROBLEMS 20 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\tempCodeRunnerFile.go"
-2
-1
0
5
5 adalah bilangan bulat positif
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\tempCodeRunnerFile.go"
17
17 adalah bilangan bulat positif
PS D:\tugas kuliah\Praktikum Algoritma>
File Edit View AA + X
NAMA : NAYAKA FARROS RIANDRA
KELAS : S1IF-13-07
NIM : 109082500055
Ln 5, Col 19 68 character AA For 100% Wind UTF-8
```

Deskripsi program

Program ini dirancang untuk memvalidasi input pengguna agar hanya menerima bilangan bulat positif menggunakan mekanisme perulangan yang dikontrol oleh variabel boolean. Pada awalnya, program mendeklarasikan variabel integer untuk menampung angka dan variabel boolean sebagai kondisi penentu berjalannya loop. Selanjutnya, program masuk ke dalam struktur perulangan yang akan terus meminta input dari pengguna dan mengevaluasi apakah angka tersebut bernilai nol atau negatif. Selama pengguna memasukkan angka nol atau bilangan negatif, status variabel kondisi akan tetap bernilai benar sehingga program terus mengulang permintaan input. Perulangan baru akan berhenti secara otomatis ketika pengguna memasukkan angka positif, yang kemudian diikuti dengan pencetakan pesan konfirmasi bahwa angka tersebut adalah bilangan bulat positif yang valid.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var a, b int
    fmt.Scan(&a)
    fmt.Scan(&b)

    for kondisi := false; !kondisi; {
        a=a-b
        fmt.Println(a)

        if a<=0{
            kondisi= a == 0
            fmt.Println(kondisi)

            kondisi=true
        }
    }
}
```

Screenshot program

The screenshot shows a code editor with a sidebar containing a file tree and a main editor area displaying Go code. The code defines a `main` package with a `main` function. Inside the function, there is a loop that scans two integers (`a` and `b`) from standard input. It then prints the value of `a` if it is less than zero, and prints "true" if `a` is equal to zero. The code editor highlights line 5 with a red box, indicating a redeclaration error: `main` is already declared in the same block.

```
File Edit Selection View ... ← → Q: Praktikum Algoritma ...
```

```
EXPLORER ... main.go guide3.go 1 tugasi.go 1 tugas2.go 1 tugas3.go 1 guide1.go 6 guide2.go 1 main2.go main3.go
```

```
PRAKTIKUM ALGORITMA minggu 13 > guide3.go > main
```

```
minggu 11 (Nayaka farro...  
soal3.go  
minggu 12 (Nayaka Farr...  
guide1.go  
guide2.go  
guide3.go  
LAPRAK MODUL 12_S1L...  
LAPRAK MODUL 12_S1L...  
soal1.go  
soal2.go  
soal3.go  
minggu 13  
guide1.go 6  
guide2.go 1  
guide3.go 1  
LAPRAK MODUL 12_S1L...  
tempCodeRunnerF...  
tugas1.go 1  
tugas2.go 1  
tugas3.go 1  
> modul 3  
> modul 5-6  
> modul 7  
> soal uts  
GUIDED.go 4  
GUIDED2.go 1  
LAPRAK MODUL 11_S1F...  
LATSOI MODUL 12 (1).pdf  
Materi Algorit.pdf  
MODUL 12 (3).pdf  
soal8.docx  
Tipe Soal_Nama_NIM_EV...  
tugas1.go 1  
tugas3.go 1  
> OUTLINE  
> TIMELINE  
> GO  
> PACKAGE OUTLINE
```

```
1 package main  
2  
3 import "fmt"  
4  
func main(){ main redeclared in this block  
5 var a, b int  
6 fmt.Scan(&a)  
7 fmt.Scan(&b)  
8  
9 for kondisi := false; !kondisi{  
10 a=a-b  
11 fmt.Println(a)  
12  
13 if a<0{  
14 kondisi= a == 0  
15 fmt.Println(kondisi)  
16 }  
17 kondisi=true  
18 }  
19  
20 }  
21  
22 }
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

```
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\guide3.go"  
5  
2  
3  
1  
-1  
false  
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\guide3.go"  
15  
3
```

```
File Edit View Aa NA + - ×
```

```
NAMA : NAYAKA FARROS RIANDRA  
KELAS : SIIIF-13-07  
NIM : 109082500055
```

```
Ln 5, Col 19 68 character AA Fo 100% Wind UTF-8
```

Deskripsi program

Algoritma pada kode ini berfokus pada manipulasi aritmatika dua bilangan bulat melalui mekanisme pengurangan bertingkat. Variabel pertama yang diinputkan pengguna akan terus-menerus dikurangi nilainya sebesar variabel kedua di dalam sebuah blok perulangan, kemudian sisa hasil setiap operasi tersebut langsung ditampilkan secara berurutan. Siklus operasi ini diatur untuk terus berjalan selama hasilnya masih positif, dan sistem baru akan memutus mata rantai perulangan secara otomatis tepat saat mendeteksi bahwa nilai akhir telah menyentuh angka nol atau menjadi negatif melalui pemeriksaan kondisi boolean di dalam blok if.

TUGAS

1. Tugas 1

Source Code

```
package main

import "fmt"

func main() {
    var n, digit int
    digit = 0
    fmt.Scan(&n)

    for n > 0 {
        n = n/10
        digit++
    }
    fmt.Println(digit)
}
```

Screenshot program

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows a project structure under "PRAKTIKUM ALGORITMA".
 - minggu 11\Nayaka farro... (containing soal5.go)
 - minggu 12\Nayaka Farri... (containing guide1.go, guide2.go, guide3.go)
 - LAPRAK MODUL 12_S11... (containing LAPRAK MODUL 12_S11..., tempCodeRunnerFile.go, tugas1.go, tugas2.go, tugas3.go)
 - minggu 13 (containing guide1.go, guide2.go, guide3.go)
 - LAPRAK MODUL 12_S11... (containing LAPRAK MODUL 12_S11..., tempCodeRunnerFile.go, tugas1.go, tugas2.go, tugas3.go)
 - modul 3
 - modul 5-6
 - modul 7
 - soal uts
 - GUIDED.go (4)
 - GUIDED2.go (1)
 - LATSOI MODUL 12 (1).pdf
 - MODUL 12 (5).pdf
 - soal 8.docx
 - Tipe Soal_Nama_NIM_EV...
 - tugas3.go (1)
 - tugas3.go (1)
 - OUTLINE
 - TIMELINE
 - GO
 - PACKAGE OUTLINE
- Code Editor:** Displays a Go file named "main.go".

```
package main
import "fmt"
func main() {
    var n, digit int
    digit = 0
    fmt.Scan(&n)
    for n > 0 {
        n = n / 10
        digit++
    }
    fmt.Println(digit)
}
```
- Terminal:** Shows the output of running the program.

```
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\tempCodeRunnerFile.go"
1
1
1
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\tempCodeRunnerFile.go"
78787
5
3
1894256
7
```
- Output Panel:** Displays student information.

NAMA :	NAYAKA FARROS RIANDRA
KELAS :	S1IF-13-07
NIM :	109082500055

Deskripsi program

Program ini berfungsi untuk menghitung jumlah digit yang terdapat dalam sebuah bilangan bulat positif yang diinputkan oleh pengguna. Setelah mendeklarasikan variabel untuk menampung angka input dan variabel penghitung yang dimulai dari nol, program menjalankan struktur perulangan yang terus beroperasi selama nilai angka tersebut lebih besar dari nol. Di dalam setiap iterasi perulangan, angka input dibagi dengan sepuluh untuk menghilangkan digit terakhirnya secara bertahap, dan secara bersamaan variabel penghitung dinaikkan nilainya satu per satu. Proses ini akan berhenti ketika angka tersebut habis atau bernilai nol, lalu program mencetak total jumlah digit yang telah dihitung ke layar terminal.

2. Tugas 2

Source code

```
package main

import (
    "fmt"
    "math"
)

func main(){
    var n float64
    fmt.Scan(&n)

    batas := math.Ceil(n)

    saatini := int(n*10)
    nlanjutan := int(batas*10)

    for i := saatini + 1; i <= nlanjutan; i++ {
        if i%10 == 0 {
            fmt.Println(i/10)
        } else {
            fmt.Println("%.1f\n", float64(i)/10.0)
        }
    }
}
```

```

        }

    }

}

```

Screenshot program

The screenshot shows the Microsoft Visual Studio Code (VS Code) interface. The main area displays a Go file named `tugas2.go` with the following code:

```

package main

import (
    "fmt"
    "math"
)

func main() {
    var n float64
    fmt.Scan(&n)

    batas := math.Ceil(n)

    saatini := int(n*10)
    nlanjutan := int(batas*10)

    for i := saatini + 1; i <= nlanjutan; i++ {
        if i*10 == 0 {
            fmt.Println(i/10)
        } else {
            fmt.Println("", float64(i) / 10.0)
        }
    }
}

```

The terminal at the bottom shows the command `PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\tugas2.go"` and its output:

```

0.2
0.3
0.4
0.5
0.6
0.8
0.9
1

```

The status bar at the bottom right indicates the current position is Line 5, Column 19, with 68 characters, and the font size is set to 100%.

Deskripsi program

Program ini dirancang untuk menampilkan urutan bilangan desimal dengan kenaikan sebesar nol koma satu, dimulai dari angka setelah input pengguna hingga mencapai bilangan bulat pembulatan ke atas terdekat. Setelah menerima input bertipe data *floating-point*, program menghitung batas akhir perulangan menggunakan fungsi matematika *ceiling*. Uniknya, untuk menjaga presisi perhitungan dan menghindari *floating-point error*, logika program mengonversi semua nilai menjadi bilangan bulat dengan mengalikannya dengan sepuluh sebelum masuk ke blok perulangan. Loop kemudian berjalan menelusuri rentang angka tersebut, lalu membagi kembali nilainya dengan sepuluh saat menampilkan *output* ke layar, sembari melakukan pengecekan kondisi untuk menentukan apakah angka tersebut harus dicetak sebagai bilangan bulat murni atau tetap dalam format desimal.

3. Tugas 3

Source code

```
package main

import "fmt"

func main(){
    var target, total, donatur int
    fmt.Println("masukkan target donasi : ")
    fmt.Scan(&target)

    total = 0
    donatur = 0

    fmt.Println("masukkan jumlah uang untuk didonasikan : ")

    for total < target {
        var donasi int
        fmt.Scan(&donasi)
        donatur+=1
        total += donasi

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

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

Screenshot program

The screenshot shows the Microsoft Visual Studio Code (VS Code) interface. The main area displays a Go file named `tugas3.go` with the following code:

```
func main() {
    var target, total, donatur int
    fmt.Println("masukkan target donasi : ")
    fmt.Scan(&target)

    total = 0
    donatur = 0

    for total < target {
        var donasi int
        fmt.Scan(&donasi)
        donatur+=1
        total += donasi
    }

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

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

The terminal window at the bottom shows the output of running the program:

```
PS D:\tugas kuliah\Praktikum Algoritma> go run "d:\tugas kuliah\Praktikum Algoritma\minggu 13\tugas3.go"
masukkan target donasi :
300
masukkan jumlah uang untuk didonasikan :
100
Donatur 1: menyumbang 100. total terkumpul: 100
50
Donatur 2: menyumbang 50. total terkumpul: 150
200
Donatur 3: menyumbang 200. total terkumpul: 350
```

On the left, the Explorer sidebar shows a file tree with various Go files, PDFs, and other documents. The status bar at the bottom right indicates the current file is `tugas3.go`, line 5, column 19, with 68 characters.

Deskripsi program

Program ini dirancang sebagai sistem pencatat donasi sederhana yang akan terus menerima masukan dana hingga target nominal tertentu terpenuhi. Pada tahap awal, pengguna diminta untuk menetapkan angka target donasi yang ingin dicapai, sementara variabel untuk total dana dan jumlah donatur diinisialisasi ke angka nol. Selanjutnya, program memasuki struktur perulangan yang akan terus berjalan secara otomatis selama total dana yang terkumpul masih berada di bawah target yang ditetapkan. Di setiap putaran donasi, program meminta input nominal sumbangan, menambahkan jumlah donatur, mengakumulasikan dana ke dalam total saldo, serta menampilkan laporan progres terkini ke layar. Sistem baru akan berhenti meminta input ketika total donasi sudah menyamai atau melampaui target, yang kemudian ditutup dengan menampilkan ringkasan akhir berisi total dana yang berhasil dihimpun beserta jumlah donatur yang berpartisipasi.