

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
        }
    }
}
```

Screenshot program

The screenshot shows a Go programming environment. On the left, there is a code editor window displaying the following Go code:

```
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 }
```

Below the code editor are tabs for PROBLEMS, OUTPUT, TERMINAL, and PORTS. The TERMINAL tab is selected, showing the following terminal session:

```
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>
```

To the right of the terminal is a status bar with the following information: Nama Rayl, File, Edit, View, Aa, Nama: Rayhan Ahza Widayamukti, Nim: 109082500210, Ln 1, Col 29, 46 character, Plain t, 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")
}
```

Screenshot program

The screenshot shows a Go programming environment with a dark theme. On the left, there is a code editor window displaying the following Go code:

```
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 }
```

Below the code editor is a terminal window titled "TERMINAL". It shows the command `go run guided2/guided2.go` being run, followed by the user input "10" and the program's response "10 merupakan bilangan bulat positif".

On the right, there is a small window titled "Nama Rayi" containing the text "Nama: Rayhan Ahza Widayamukti" and "Nim: 109082500210".

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)

}
```

Screenshot program

The screenshot shows a Go programming environment. The code in the editor is:

```
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 execution of the program:

```
> ^ TERMINAL
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>
```

A floating window titled "Nama Rayl" displays the user's input:

Nama: Rayhan Ahza Widyamukti
Nim: 109082500210

Information at the bottom of the terminal window:

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

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 if $x \leq 0$ 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)
}
```

Screenshot program

The screenshot shows a Go programming environment. The code editor displays the following Go code:

```
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 }
```

The terminal window shows the execution of the program:

```
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>
```

The status bar at the bottom right indicates the current file is "Nama Rayl", the character count is 46, the font is Plain t, the zoom level is 100%, and the encoding is UTF-8.

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
        }
    }
}
```

Screenshot program

The screenshot shows a Go programming environment. On the left, there is a code editor with the following Go code:

```
2
3     import "fmt"
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 }
```

Below the code editor is a terminal window titled 'TERMINAL' showing the execution of the program:

```
> 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>
```

To the right of the terminal is a status bar displaying the user's name and NIM:

Nama Rayl • + File Edit View Aa 🌐 🎨 ⚙️ Ln 1, Col 29 | 46 caracte Plain t 100% Wind UTF-8
Nama: Rayhan Ahza Widyamukti
Nim: 109082500210

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.Println("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.\n"
Total terkumpul: %d", donatur, donasi, total)

        if total >= target {
            break
        }
    }

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

Screenshot program

The screenshot shows a terminal window with a dark background featuring a character's face. The terminal displays a Go program and its execution output.

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

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>
```

A floating window titled "Nama Rayl" is open, showing the user's information:

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.