

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 13
TIPE DATA DAN VARIABEL



Disusun oleh:

DIVA ZAHRAH NABILA

109082500112

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

1. Guided 1

Source code

```
package main

import "fmt"

func main() {
    var text string
    var n int
    fmt.Scan(&text, &n)
    i := 0
    for kondisi := false; !kondisi; {
        fmt.Println(text)
        i++
        kondisi = (i >= n)
    }
}
```

Screenshot program

The screenshot shows a code editor interface with a Go file named `guided1.go` open. The code defines a `main` function that reads a string and an integer from standard input, then prints the string repeatedly until the integer is reached. The terminal below shows the program running and printing 'pagi' 3 times and 'kursi' 5 times.

```
File Edit Selection View Go ... ← → Q MODUL 13
EXPLORER ... Welcome Untitled-1 -> guided1.go 2 × -> guided2.go 1 -> guided3.go 1
MODUL 13 -> guided1.go 2
-> guided2.go 1
-> guided3.go 1
3 import "fmt"
4
5 func main() {
6     var text string
7     var n int
8     fmt.Scan(&text, &n)
9     i := 0
10    for kondisi := false; !kondisi; {
11        fmt.Println(text)
12        i++
13        kondisi = (i >= n)
14    }
15
16
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\guided1.go"
pagi 3
pagi
pagi
pagi
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\guided1.go"
kursi 5
kursi
kursi
kursi
kursi
kursi
```

Deskripsi program

Program ini dirancang untuk meminta input dari pengguna dalam dua bentuk yaitu, sebuah teks akan dicetak, dan angka yang menunjukkan berapa kali teks tersebut harus ditampilkan. Setelah menerima input, program menggunakan sebuah loop yang dikendalikan oleh boolean untuk melakukan perulangan. Di dalam loop, program mencetak teks yang diberikan, kemudian menambah penghitung setiap kali teks dicetak. Saat jumlah pengulangan sama dengan angka yang dimasukkan pengguna, kondisi boolean diubah sehingga loop berhenti, dan program selesai dijalankan.

2. Guided 2

Source code

```
package main

import "fmt"

func main() {
    var x int
    for kondisi := false; !kondisi; {
        fmt.Scan(&x)
        kondisi = (x > 0)
    }
    fmt.Println(x, "adalah bilangan bulat positif")
}
```

Screenshot program

```
File Edit Selection View Go ... ← → MODUL 13
EXPLORER ... Welcome Untitled-1 guided1.go 2 guided2.go 1 guided3.go 1
-oo guided2.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x int
7     for kondisi := false; !kondisi; {
8         fmt.Scan(&x)
9         kondisi = (x > 0)
10    }
11    fmt.Println(x, "adalah bilangan bulat positif")
12 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\guided2.go"
-5
-2
-1
0
5
5 adalah bilangan bulat positif
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\guided2.go"
17
17 adalah bilangan bulat positif
```

Deskripsi program

Program ini meminta pengguna untuk memasukkan sebuah bilangan bulat dan akan terus mengulang permintaan input sampai pengguna memasukkan bilangan positif. Setiap kali pengguna memasukkan angka nol atau negatif, program tidak langsung berhenti, tapi meminta input lagi menggunakan loop yang dikontrol oleh kondisi boolean. Jika pengguna memasukkan bilangan positif, loop berhenti dan program mencetak angka tersebut beserta teks bahwa angka itu adalah bilangan bulat positif.

3. Guided 3

Source code

```
package main

import "fmt"

func main() {
    var x, y int
```

```

        fmt.Scan(&x, &y)
        for selesai := false; !selesai; {
            x = x - y
            fmt.Println(x)
            selesai = (x <= 0)
        }
        fmt.Println(x == 0)
    }
}

```

Screenshot program

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

- File Bar:** File, Edit, Selection, View, Go, ...
- Toolbar:** Welcome, Untitled-1, guided1.go 2, guided2.go 1, guided3.go 1 (highlighted)
- Editor Area:**

```

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

```

PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\guided3.go"
5 2
5 2
3
1
-1
false
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\guided3.go"
15 3
12
9
6
3
0
true
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\guided3.go"
25 5
20
15
10
5
0
true
```
- Right Panel:**
 - Build with Agent:** A button to generate AI responses.
 - Agent Instructions:** A text input field for generating agent instructions.
 - Build Next:** A button to build the next file.

Deskripsi program

Program ini meminta pengguna memasukkan dua bilangan bulat, x dan y , lalu mengurangi x dengan y secara berulang sambil menampilkan hasil setiap pengurangan. Proses pengulangan terus berlangsung sampai x menjadi nol atau negatif, menggunakan loop yang dikontrol oleh variabel boolean `selesai`. Setelah pengulangan selesai, jika program x sama dengan nol akan mencetak `true` dan jika tidak sama dengan nol akan menampilkan `false`.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

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

    digit := 0
    selesai := false
    for !selesai {
        n = n / 10
        digit++
        selesai = (n == 0)
    }
    fmt.Println(digit)
}
```

Screenshot program

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

- File Explorer:** Shows files in the 'MODUL 13' folder: 'guided1.go' (5), 'guided2.go' (1), 'guided3.go' (1), 'soal1.go' (1), 'soal2.go' (1), and 'soal3.go' (1). 'soal1.go' is currently selected.
- Code Editor:** Displays the source code for 'soal1.go'. The code is identical to the one shown above.
- Terminal:** Shows the command-line output of running the program:

```
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal1.go"
234
3
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal1.go"
78787
5
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal1.go"
1894256
7
```

Deskripsi program

Program ini membaca sebuah bilangan bulat n dari pengguna, lalu menghitung jumlah digit dari bilangan tersebut dengan cara membaginya dengan 10 secara berulang. Program menggunakan sebuah loop yang dikontrol oleh variabel Boolean. setiap kali n dibagi 10, penghitung digit bertambah satu. Loop berlanjut hingga n menjadi nol, yang berarti semua digit telah dihitung. Setelah loop selesai, program mencetak nilai digit, yaitu jumlah digit dari bilangan asli yang dimasukkan pengguna.

2. Tugas 2

Source code

```
package main

import "fmt"

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

    target := int(x)
    if float64(target) < x {
        target++
    }
    selesai := false
    for !selesai {
        x += 0.1
        fmt.Printf("%.1f\n", x)
        selesai = (x >= float64(target))
    }
}
```

Screenshot program

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

- File Explorer:** Shows files in the current workspace, including `guided1.go`, `guided2.go`, `guided3.go`, `soal1.go`, `soal2.go`, and `soal3.go`.
- Code Editor:** The main pane displays the content of `soal2.go`. The code reads a floating-point number `x` from standard input, converts it to an integer `target`, and then iterates, adding 0.1 to `x` in each step until `x` is greater than or equal to `target`. It then prints `x` to the console.
- Terminal:** The bottom pane shows the terminal output of running the program. It starts with the command `PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal2.go"`. The output lists values from 0.2 to 1.1, followed by the command `PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal2.go"` again, and finally the value 2.7.

Deskripsi program

Program ini membaca sebuah bilangan desimal `x` dari pengguna, kemudian menambahkan nilai 0.1 secara bertahap hingga mencapai bilangan bulat berikutnya. Pertama, program mengubah `x` menjadi bilangan bulat (`target`) dengan cara membuang bagian desimal. Jika `x` awal bukan bilangan bulat, program menaikkan `target` agar menjadi bilangan bulat yang lebih besar dari `x`. Lalu, program menggunakan loop yang dikontrol oleh variabel boolean `selesai` untuk menambahkan 0.1 ke `x` di setiap iterasi, sambil menampilkan nilai `x` dengan satu angka di belakang koma. Loop berhenti ketika `x` mencapai atau melebihi `target`.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var target, donasi int
    fmt.Scan(&target)

    total := 0
    jumlah := 0

    selesai := false
    for !selesai {
        fmt.Scan(&donasi)
        jumlah++
        total += donasi

        fmt.Printf("Donatur %d: Menyumbang %d. Total
terkumpul: %d\n", jumlah, donasi, total)
        selesai = (total >= target)
    }
    fmt.Printf("Target tercapai! Total donasi: %d dari %d
donatur.\n", total, jumlah)
}
```

Screenshot program

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

- File Explorer:** Shows files in the 'MODUL 13' directory: guided1.go, guided2.go, guided3.go, soal1.go, soal2.go, and soal3.go.
- Code Editor:** The 'soal3.go' file is open, displaying the provided Go code. The cursor is at the end of the final closing brace of the main function.
- Output Terminal:** The terminal window shows the execution of the code. It starts with the command 'go run "d:\LAPRAK\MODUL 13\soal3.go"'. The output displays four runs of the program, each with a different target value (100, 150, 200, 300) and a different number of donors (3, 4, 3, 4). In each run, it prints the individual donor contributions and the total amount collected until the target is reached.

```
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal3.go"
300
100
Donatur 1: Menyumbang 100. Total terkumpul: 100
50
Donatur 2: Menyumbang 50. Total terkumpul: 150
200
Donatur 3: Menyumbang 200. Total terkumpul: 350
Target tercapai! Total donasi: 350 dari 3 donatur.
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal3.go"
500
150
Donatur 1: Menyumbang 150. Total terkumpul: 150
100
Donatur 2: Menyumbang 100. Total terkumpul: 250
50
Donatur 3: Menyumbang 50. Total terkumpul: 300
300
Donatur 4: Menyumbang 300. Total terkumpul: 600
Target tercapai! Total donasi: 600 dari 4 donatur.
PS D:\LAPRAK\MODUL 13> go run "d:\LAPRAK\MODUL 13\soal3.go"
200
300
Donatur 1: Menyumbang 300. Total terkumpul: 300
Target tercapai! Total donasi: 300 dari 1 donatur.
PS D:\LAPRAK\MODUL 13>
```

Deskripsi program

Program Go ini dirancang untuk mencatat donasi hingga mencapai target tertentu. Program meminta pengguna memasukkan target, yaitu jumlah total donasi yang ingin dicapai. Lalu, program menggunakan loop yang dikontrol oleh variabel boolean selesai untuk membaca jumlah donasi dari setiap donatur satu per satu. Setiap kali ada input donasi baru (donasi), program menambahkan nilai tersebut ke total, menambah penghitung jumlah donatur, dan menampilkan informasi sementara yang mencakup nomor donatur, jumlah donasi yang diberikan, dan total donasi terkumpul hingga saat itu. Loop akan terus berjalan sampai total donasi mencapai atau melebihi target. Setelah target tercapai, program menampilkan pesan akhir yang memberi tahu bahwa target sudah terpenuhi, beserta total donasi yang terkumpul dan jumlah donatur yang berpartisipasi.