

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

JEREMY CHRISTIAN SIHOMBING

109082500178

S1IF-13-07

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

SOAL

1. SOAL 1

Source code

```
package main

import "fmt"

func main() {

    var asterik int

    fmt.Print("Masukkan ketinggian pohon natal lebih dari 8
asterik: ")

    fmt.Scan(&asterik)

    for i := 1; i <= asterik; i++ {

        for j := 0; j < asterik-i; j++ {

            fmt.Print(" ")

        }

        for k := 0; k < (2*i)-1; k++ {

            fmt.Print("*")

        }

        fmt.Println()

    }

    for i := 0; i < 2; i++ {

        for j := 0; j < asterik-1; j++ {

            fmt.Print(" ")

        }

        fmt.Println("|")

    }

}
```

The screenshot shows the VS Code editor with three files open: Soal1.go, Soal2.go, and Soal3.go. The active file is Soal1.go, which contains the following Go code:

```
1 package main
2 import "fmt"
3 func main() {
4
5     var asterik int
6     fmt.Println("Masukkan ketinggian pohon natal lebih dari 8 asterik: ")
7     fmt.Scan(&asterik)
8     for i := 1; i <= asterik; i++ {
9         for j := 0; j < asterik-i; j++ {
10             fmt.Print(" ")
11         }
12         for k := 0; k < (2*i)-1; k++ {
13             fmt.Print("*")
14         }
15         fmt.Println()
16     }
17
18     for i := 0; i < 2; i++ {
19         for j := 0; j < asterik-1; j++ {
20             fmt.Print(" ")
21         }
22         fmt.Println("|")
23     }
24 }
```

To the right of the editor, there is a window titled "NIM 10908" containing the following information:

```
NIM 109082500178
KELAS SIIF-13-07
NAMA JEREMY CHRISTIAN SIHOMBING
```

Below the editor, the TERMINAL panel shows the command prompt output:

```
PS C:\Users\LENOVO\Downloads\UAS ALPRO> go run '.\SOAL 1\Soal1.go'
Masukkan ketinggian pohon natal lebih dari 8 asterik: 10
***
*****
*****
*****
*****
*****
*****
*****
*****
*****
|
PS C:\Users\LENOVO\Downloads\UAS ALPRO>
```

Program di atas ini digunakan untuk menampilkan pola pohon Natal menggunakan karakter asterik (*) dan batang pohon menggunakan karakter garis vertikal (|). Pada awal program, pengguna diminta memasukkan tinggi pohon dalam jumlah asterik, dengan ketentuan nilainya lebih dari 8. Nilai ini disimpan dalam variabel asterik dan akan menentukan tinggi serta lebar pola pohon yang ditampilkan.

Setelah bagian daun selesai dicetak, program mencetak batang pohon. Batang ini dibuat dengan perulangan yang berjalan dua

kali untuk menampilkan tinggi batang. Pada setiap baris batang, program mencetak spasi terlebih dahulu agar batang berada tepat di tengah pohon, lalu mencetak satu karakter |. Jadi, keseluruhan output membentuk tampilan pohon Natal lengkap dengan batangnya.

2. SOAL 2

Source code

```
package main

import "fmt"

func main() {

    var menu, porsi, harga int
    var menumakanan string

    fmt.Println("=== MENU EMPAL GENTONG MAS FUAD ===")
    fmt.Println("1. Empal Gentong Biasa - Rp22.000")
    fmt.Println("2. Empal Asem - Rp22.000")
    fmt.Println("3. Sate Kambing Muda - Rp40.000")
    fmt.Println("4. Nasi Lengko Pagongan - Rp15.000")
    fmt.Print("Pilih menu (1-4): ")
    fmt.Scan(&menu)

    fmt.Print("Masukkan jumlah beli: ")
    fmt.Scan(&porsi)

    switch menu {
    case 1:
        menumakanan = "Empal Gentong Biasa"
        harga = 22000
    case 2:
        menumakanan = "Empal Asem"
        harga = 22000
    case 3:
```

```
        menumakanan = "Sate Kambing Muda"

        harga = 40000

    case 4:

        menumakanan = "Nasi Lengko Pagongan"

        harga = 15000

    default:

        harga = 0

    }

    fmt.Println("")

    fmt.Println("=== STRUK PEMBAYARAN ===")

    fmt.Println("Menu          :", menumakanan)

    fmt.Println("Harga          : Rp", harga)

    fmt.Println("Jumlah         :", porsi)

    fmt.Println("Total          : Rp", harga*porsi)

}
```

Screenshoot program

The screenshot shows a Go IDE with three files: Soal1.go, Soal2.go, and Soal3.go. The active file is Soal2.go, which contains a Go program for a menu simulation. The program defines a menu with four items, each with a name, price, and a default quantity of 2. It prompts the user to select a menu item and enter the quantity. A switch statement handles the selection, setting the menu name and price based on the user's choice. If the choice is invalid, the price is set to 0. Finally, it prints a receipt showing the selected menu, unit price, quantity, and total price.

```
1 package main
2 import "fmt"
3 func main() {
4
5     var menu, porsi, harga int
6     var menumakanan string
7     fmt.Println("=== MENU EMPAL GENTONG MAS FUAD ===")
8     fmt.Println("1. Empal Gentong Biasa - Rp22.000")
9     fmt.Println("2. Empal Asem - Rp22.000")
10    fmt.Println("3. Sate Kambing Muda - Rp40.000")
11    fmt.Println("4. Nasi Lengko Pagongan - Rp15.000")
12    fmt.Print("Pilih menu (1-4): ")
13    fmt.Scan(&menu)
14    fmt.Print("Masukkan jumlah beli: ")
15    fmt.Scan(&porsi)
16
17    switch menu {
18    case 1:
19        menumakanan = "Empal Gentong Biasa"
20        harga = 22000
21    case 2:
22        menumakanan = "Empal Asem"
23        harga = 22000
24    case 3:
25        menumakanan = "Sate Kambing Muda"
26        harga = 40000
27    case 4:
28        menumakanan = "Nasi Lengko Pagongan"
29        harga = 15000
30    default:
31        harga = 0
32    }
33    fmt.Println("")
34    fmt.Println("=== STRUK PEMBAYARAN ===")
35    fmt.Println("Menu      :", menumakanan)
36    fmt.Println("Harga       : Rp", harga)
37    fmt.Println("Jumlah      :", porsi)
38    fmt.Println("Total      : Rp", harga*porsi)
39 }
```

The terminal output shows the program's execution. It displays the menu, prompts for selection and quantity, and prints the receipt. The user input was menu 2 and quantity 2, resulting in a total of Rp 44,000.

```
PS C:\Users\LENOVO\Downloads\UAS ALPRO> go run '.\SOAL 2\Soal2.go'
=== MENU EMPAL GENTONG MAS FUAD ===
1. Empal Gentong Biasa - Rp22.000
2. Empal Asem - Rp22.000
3. Sate Kambing Muda - Rp40.000
4. Nasi Lengko Pagongan - Rp15.000
Pilih menu (1-4): 2
Masukkan jumlah beli: 2

=== STRUK PEMBAYARAN ===
Menu      : Empal Asem
Harga       : Rp 22000
Jumlah      : 2
Total      : Rp 44000
PS C:\Users\LENOVO\Downloads\UAS ALPRO>
```

Deskripsi program

Program di atas ini digunakan untuk mensimulasikan sistem pemesanan sederhana pada rumah makan Empal Gentong Mas Fuad. Jadi di awal program nanti akan ditampilkan daftar menu makanan lengkap dengan nomor pilihan dan harga masing-masing. Nanti pengguna diminta memilih menu dengan memasukkan nomor menu serta menentukan jumlah porsi yang ingin dibeli.

Setelah input diterima, program menggunakan struktur switch untuk menentukan nama menu dan harga berdasarkan nomor menu yang dipilih. Jika pilihan menu sesuai, variabel menumakanan dan harga akan diisi dengan data yang benar. Jika pengguna memasukkan nomor menu di luar pilihan yang tersedia, maka harga akan bernilai nol sebagai penanda bahwa pilihan tidak valid.

Selanjutnya, program menampilkan struk pembayaran yang berisi nama menu yang dipilih, harga satuan, jumlah porsi, serta total

harga yang harus dibayar. Total harga diperoleh dari hasil perkalian antara harga satuan dan jumlah porsi. Jadi, program ini membantu pengguna melihat ringkasan pesanan dan total biaya secara sederhana dan mudah dipahami.

3. SOAL 3

Source code

```
package main
import "fmt"
func main() {

    var a, b, c int
    fmt.Print("Masukkan tiga bilangan bulat positif a b
dan c: ")
    fmt.Scan(&a, &b, &c)
    switch {
    case a + b <= c:
        fmt.Println("Bukan segitiga")
    case a == c:
        fmt.Println("Segitiga sama sisi")
    case (a*a)+(b*b) == (c*c):
        fmt.Println("Segitiga siku-siku")
    case a == b || b == c:
        fmt.Println("Segitiga sama kaki")
    default:
        fmt.Println("Segitiga sembarang")
    }
}
```

Screenshoot program

```
SOAL 3 > -o Soal3.go > main
1 package main
2 import "fmt"
3 func main() {
4
5     var a, b, c int
6     fmt.Print("Masukkan tiga bilangan bulat positif a b dan c: ")
7     fmt.Scan(&a, &b, &c)
8     switch {
9     case a + b <= c:
10         fmt.Println("Bukan segitiga")
11     case a == c:
12         fmt.Println("Segitiga sama sisi")
13     case (a*a)+(b*b) == (c*c):
14         fmt.Println("Segitiga siku-siku")
15     case a == b || b == c:
16         fmt.Println("Segitiga sama kaki")
17     default:
18         fmt.Println("Segitiga sembarang")
19     }
20 }
```

```
PS C:\Users\LENOVO\Downloads\UAS ALPRO> go run '.\SOAL 3\Soal3.go'
Masukkan tiga bilangan bulat positif a b dan c: 3 3 3
Segitiga sama sisi
PS C:\Users\LENOVO\Downloads\UAS ALPRO> go run '.\SOAL 3\Soal3.go'
Masukkan tiga bilangan bulat positif a b dan c: 5 5 8
Segitiga sama kaki
PS C:\Users\LENOVO\Downloads\UAS ALPRO> go run '.\SOAL 3\Soal3.go'
Masukkan tiga bilangan bulat positif a b dan c: 3 4 5
Segitiga siku-siku
PS C:\Users\LENOVO\Downloads\UAS ALPRO> go run '.\SOAL 3\Soal3.go'
Masukkan tiga bilangan bulat positif a b dan c: 4 5 6
Segitiga sembarang
PS C:\Users\LENOVO\Downloads\UAS ALPRO> go run '.\SOAL 3\Soal3.go'
Masukkan tiga bilangan bulat positif a b dan c: 1 2 3
Bukan segitiga
PS C:\Users\LENOVO\Downloads\UAS ALPRO> 
```

Deskripsi program

Program di atas ini digunakan untuk menentukan jenis segitiga berdasarkan tiga bilangan bulat positif yang dimasukkan oleh pengguna, yaitu a , b , dan c . Ketiga bilangan tersebut dianggap sebagai panjang sisi-sisi segitiga. Setelah pengguna memasukkan nilainya, program langsung melakukan pengecekan menggunakan struktur switch dengan kondisi.

Pengecekan pertama dilakukan untuk memastikan apakah ketiga sisi tersebut dapat membentuk segitiga. Jika jumlah dua sisi lebih kecil atau sama dengan sisi yang lain, maka bangun tersebut tidak memenuhi syarat segitiga dan program menampilkan bahwa itu bukan segitiga. Jika lolos dari kondisi tersebut, program kemudian membandingkan nilai sisi-sisinya untuk menentukan jenis segitiga.

Apabila semua sisi sama panjang, program menyatakan bahwa segitiga tersebut adalah segitiga sama sisi. Jika memenuhi hubungan kuadrat sisi-sisi yang menunjukkan sudut siku-siku, maka segitiga dikategorikan sebagai segitiga siku-siku. Jika hanya dua sisi yang sama panjang, segitiga dinyatakan sebagai

segitiga sama kaki. Jika tidak memenuhi semua kondisi sebelumnya, maka segitiga tersebut termasuk segitiga sembarang.