

LAPORAN PRAKTIKUM ALGORITMA

DAN PEMROGRAMAN 1

EVALUASI



**Telkom
University
PURWOKERTO**

Disusun oleh:

MOHAMAD ERLANGGA ZEIN

109082500020

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. SOAL 1

Source Code

```
package main

import "fmt"

func main() {
    var n int

    fmt.Print("Input ketinggian piramida (N): ")

    fmt.Scan(&n)

    for i := 1; i <= n; i++ {
        for j := 0; j < n-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 < n-1; j++ {
            fmt.Print(" ")
        }
    }
}
```

```

        fmt.Println("|")

    }

}

```

Screenshot program

The screenshot shows a VS Code interface with a dark theme. The Explorer sidebar on the left lists three files: `soal1.go`, `soal2.go`, and `soal3.go`. The `soal1.go` file is open in the editor, displaying the following Go code:

```

1 package main
2 import "fmt"
3
4 func main() {
5     var n int
6
7     fmt.Print("Input ketinggian piramida (N): ")
8     fmt.Scan(&n)
9
10    for i := 1; i <= n; i++ {
11        for j := 0; j < n-i; j++ {
12            fmt.Print(" ")
13        }
14        for k := 0; k < (2*i)-1; k++ {
15            fmt.Print("*")
16        }
17        fmt.Println()
18    }
19
20    for i := 0; i < 2; i++ {
21        for j := 0; j < n-1; j++ {
22            fmt.Print(" ")
23        }
24        fmt.Println("|")
25    }
}

```

The terminal window at the bottom shows the output of running the program with `go run soal1.go`. It first asks for input, then prints a pyramid for `n=3` and a batang for `n=10`.

Deskripsi program

Program ini adalah program untuk mencetak pola visual berbentuk pohon atau piramida bintang, dimana program akan berjalan berdasarkan input ketinggian yang dimasukkan user, disini terdapat satu variabel utama integer n untuk menampung nilai tinggi tersebut, serta variabel i, j, dan k sebagai variabel bantu iterasi, dimana proses dimulai dengan perulangan utama untuk mencetak bagian daun dari 1 sampai n, di dalam perulangan ini terdapat dua logika yaitu mencetak spasi sebanyak $n-i$ untuk perataan tengah dan mencetak bintang sebanyak $(2*i)-1$ untuk membentuk pola ganjil yang melebar, kemudian setelah bagian daun selesai, program melanjutkan ke perulangan terpisah untuk mencetak batang sebanyak dua baris dengan logika spasi $n-1$, contoh misal user menginputkan ketinggian piramida (N) sebanyak 3, maka pada perulangan pertama saat $i=1$, program akan mencetak spasi $3-1 = 2$ spasi, kemudian mencetak bintang $(2*1)-1 = 1$ bintang, lanjut ke iterasi kedua saat $i=2$, spasi menjadi $3-2 = 1$ dan bintang menjadi $(2*2)-1 = 3$ bintang, lalu saat $i=3$, spasi menjadi 0 dan bintang menjadi 5, setelah itu program masuk ke bagian batang, program akan mencetak spasi $3-1 = 2$ spasi lalu mencetak garis | dan proses ini diulang dua kali, sehingga terbentuklah pohon dengan susunan 1, 3, 5 bintang dan batang di bawahnya yang tersusun rapi di tengah.

2. SOAL 2

Source Code

```
package main

import "fmt"

func main() {
    var menu int
    var jumlahBeli int
    var total int
    var namaMenu string
    var hargaMenu int

    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(&jumlahBeli)

    switch menu {
        case 1:
            namaMenu = "Empal Gentong Biasa"
```

```
    hargaMenu = 22000

    case 2:

        namaMenu = "Empal Asem"

        hargaMenu = 22000

    case 3:

        namaMenu = "Sate Kambing Muda"

        hargaMenu = 40000

    case 4:

        namaMenu = "Nasi Lengko Pagongan"

        hargaMenu = 15000

    default:

        fmt.Println("tidak ada di dalam menu woii")

    }

    total = hargaMenu * jumlahBeli

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

    fmt.Println("Menu      : ", namaMenu)

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

    fmt.Println("Jumlah   : ", jumlahBeli)

    fmt.Println("Total     : Rp ", total)

    fmt.Println("\n==== Code Execution Successful ====")

}
```

Screenshot program

The screenshot shows a GoLang IDE interface. On the left, the code editor displays a Go program named `soal2.go`. The code defines a `main` function with a `main` block redeclaration. It lists five menu items: Empal Gentong Biasa (Rp22.000), Empal Asem (Rp22.000), Sate Kambing Muda (Rp40.000), and Nasi Lengko Pagongan (Rp15.000). It then prompts the user to choose a menu item (1-4) and enter the quantity. The `switch` statement handles four cases corresponding to the menu items. If the input is not 1-4, it falls back to a default case. The total price is calculated as the product of the menu price and the quantity. Finally, a receipt is printed with the customer's name, NIM, and class. The terminal window below shows the command `go run soal2.go` being run, followed by the program's output which matches the screenshot above.

```
1 package main
2 import "fmt"
3
4 func main() {
5     var menu int
6     var jumlahBeli int
7     var total int
8     var namaMenu string
9     var hargaMenu int
10
11    fmt.Println("== MENU EMPAL GENTONG MAS FUAD ===")
12    fmt.Println("1. Empal Gentong Biasa - Rp22.000")
13    fmt.Println("2. Empal Asem - Rp22.000")
14    fmt.Println("3. Sate Kambing Muda - Rp40.000")
15    fmt.Println("4. Nasi Lengko Pagongan - Rp15.000")
16
17    fmt.Print("Pilih menu(1-4): ")
18    fmt.Scan(&menu)
19    fmt.Print("Masukkan jumlah beli: ")
20    fmt.Scan(&jumlahBeli)
21
22    switch menu {
23        case 1:
24            namaMenu = "Empal Gentong Biasa"
25            hargaMenu = 22000
26        case 2:
27            namaMenu = "Empal Asem"
28            hargaMenu = 22000
29        case 3:
30            namaMenu = "Sate Kambing Muda"
31            hargaMenu = 40000
32        case 4:
33            namaMenu = "Nasi Lengko Pagongan"
34            hargaMenu = 15000
35        default:
36            fmt.Println("tidak ada di dalam menu woi!")
37    }
38
39    total = hargaMenu * jumlahBeli
40
41    fmt.Println("\n== STRUK PEMBAYARAN ==")
42    fmt.Println("Menu : ", namaMenu)
43    fmt.Println("Harga : Rp ", hargaMenu)
44    fmt.Println("Jumlah : ", jumlahBeli)
45    fmt.Println("Total : Rp ", total)
46    fmt.Println("== Code Execution Successful ==")
47 }
```

```
PS C:\GoLang> go run 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):
Masukkan jumlah beli: 2
== STRUK PEMBAYARAN ==
Menu : Empal Asem
Harga : Rp 22000
Jumlah : 2
Total : Rp 44000
== Code Execution Successful []
PS C:\GoLang>
```

Deskripsi program

Program ini adalah program yang mirip seperti kasir sederhana untuk menghitung total pembayaran di warung makan "Empal Gentong Mas Fuad", disini terdapat 5 variabel utama dengan tipe data integer dan string, dimana 5 variabel itu adalah menu, jumlahBeli, total, namaMenu, dan hargaMenu, proses dimulai dengan menampilkan daftar 4 pilihan makanan beserta harganya ke layar, kemudian user diminta untuk menginputkan angka pilihan menu (1-4) dan memasukkan jumlah porsi yang ingin dibeli, selanjutnya program akan masuk ke logika percabangan *switch-case* untuk menentukan isi dari variabel namaMenu dan hargaMenu sesuai inputan user, lalu setelah harga didapatkan, program akan melakukan proses perhitungan matematika dimana variabel total akan diisi hasil perkalian antara hargaMenu dengan jumlahBeli, contoh misal user menginputkan pilihan menu nomor 2, maka apakah input 2 ada di dalam *case*? Ya, maka sistem mendeteksi itu adalah "Empal Asem" dengan harga 22000, kemudian user memasukkan jumlah beli sebanyak 2 porsi, maka program akan menghitung $total = 22000 * 2 = 44000$, setelah perhitungan selesai program akan mencetak struk pembayaran yang berisi rincian nama menu, harga satuan, jumlah beli, dan total harga akhir, namun jika user memilih nomor di luar 1-4, program akan lari ke default dan mencetak pesan error "tidak ada di dalam menu woi".

3. SOAL 3

Source Code

```
package main

import "fmt"

func main() {
    var a, b, c int
    fmt.Print("Masukkan angka: ")
    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")
    }
}
```

Screenshot program

The screenshot shows the VS Code interface with the following details:

- Editor:** The main editor window displays a Go file named `soal3.go`. The code defines a `main` function that prompts the user for three integers (`a`, `b`, `c`) and prints their sum. It then uses a switch statement to classify the triangle based on the lengths of its sides:
 - If $a + b \leq c$, it prints "Bukan segitiga".
 - If $a = c$, it prints "Segitiga sama sisi".
 - If $(a*a)+(b*b) == (c*c)$, it prints "Segitiga siku siku".
 - If $a == b \text{ || } b == c$, it prints "Segitiga sama kaki".
 - Otherwise, it prints "Segitiga sembarang".
- Terminal:** Below the editor, the terminal window shows the execution of the program. It runs `go run soal3.go` multiple times with different input values (e.g., 3 4 5, 5 5 8, 4 5 6, 1 2 3) and prints the corresponding triangle types.
- Output:** A floating terminal window titled "Nan" shows the results of the program execution for input 1, 2, 3.

Deskripsi program

Program ini adalah program untuk mengidentifikasi jenis segitiga berdasarkan panjang ketiga sisinya, disini terdapat 3 variabel integer yaitu `a`, `b`, dan `c` yang menampung nilai panjang sisi yang diinputkan user, proses dimulai ketika user memasukkan tiga angka tersebut, lalu program melakukan pengecekan beruntun menggunakan logika switch case, kondisi pertama mengecek validitas segitiga dimana jika $a + b \leq c$ maka dianggap "Bukan segitiga", contoh misal user menginput 1, 2, 3, apakah $1+2 \leq 3$? Ya, maka program berhenti dan mencetak "Bukan segitiga", namun jika tidak, program lanjut ke kondisi kedua mengecek apakah $a == c$, jika ya seperti pada input 3, 3, 3 maka dicetak "Segitiga sama sisi", jika masih belum terpenuhi, program mengecek rumus Pythagoras $(a*a)+(b*b) == (c*c)$, contoh input 3, 4, 5, karena $9+16$ adalah 25, maka dicetak "Segitiga siku siku", selanjutnya program mengecek apakah ada dua sisi yang sama panjang dengan $a == b$ atau $b == c$ untuk jenis "Segitiga sama kaki" seperti input 5, 5, 8, dan terakhir jika semua kondisi diatas tidak terpenuhi, contohnya input 4, 5, 6, maka program otomatis menganggapnya sebagai "Segitiga sembarang" melalui kondisi default.