

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

Didi Hermawanto

109082500088

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 N int
    fmt.Scan(&N)

    if N >= 8 {
        fmt.Println("Tinggi pohon terlalu besar,
silakan kurangi (maksimal 7)")
        return
    }

    for i := 1; i <= N; i++ {
        for j := 1; j <= N-i; j++ {
            fmt.Print(" ")
        }
        for k := 1; k <= (2*i-1); k++ {
            fmt.Print("*")
        }
        fmt.Println()
    }
}
```

Screenshoot program


```
var pilihan, jumlah int

fmt.Println("=== DAFTAR PRODUK TOKO BUDI ===")
fmt.Println("1. Little Trees          - Rp35000")
fmt.Println("2. Lap microfiber        - Rp25000")
fmt.Println("3. Cover Steer                   - Rp150000")
fmt.Println("4. Sponge Cuci Mobil          - Rp10000")

fmt.Print("Pilih produk (1-4): ")
fmt.Scan(&pilihan)

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

var nama string
var harga int

if pilihan == 1 {
    nama = "Little Trees"
    harga = 35000
} else if pilihan == 2 {
    nama = "Lap microfiber"
    harga = 25000
} else if pilihan == 3 {
    nama = "Cover Steer"
    harga = 150000
} else if pilihan == 4 {
    nama = "Sponge Cuci Mobil"
    harga = 10000
}
```

```

        total := harga * jumlah

        fmt.Println("\n=== STRUK PEMBAYARAN ===")
        fmt.Println("Produk :", nama)
        fmt.Println("Harga   : Rp", harga)
        fmt.Println("Jumlah  :", jumlah)
        fmt.Println("Total   : Rp", total)
    }

```

Screenshoot program

```

6
7     fmt.Println("=== DAFTAR PRODUK TOKO BUDI ===")
8     fmt.Println("1. Little Trees      - Rp35000")
9     fmt.Println("2. Lap microfiber    - Rp25000")
10    fmt.Println("3. Cover Steer      - Rp150000")
11    fmt.Println("4. Sponge Cuci Mobil - Rp10000")
12
13    fmt.Print("Pilih produk (1-4): ")
14    fmt.Scan(&pilihan)

```

PROBLEMS 4 OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```

=== DAFTAR PRODUK TOKO BUDI ===
1. Little Trees      - Rp35000
2. Lap microfiber    - Rp25000
3. Cover Steer      - Rp150000
4. Sponge Cuci Mobil - Rp10000
Pilih produk (1-4): 1
Masukkan jumlah beli: 5

=== STRUK PEMBAYARAN ===
Produk : Little Trees
Harga   : Rp 35000
Jumlah  : 5
Total   : Rp 175000

```

PS C:\coding\didid\golang\Kode Asesmen>

Ln 11, Col 51 Tab Size: 4 UTF-8 CRLF Go 1.25.1 Go Live Windsurf: Login Prettier

Deskripsi program

Program diawali dengan deklarasi package main dan import fmt yang digunakan untuk menampilkan keluaran dan membaca input. Di dalam fungsi main, dideklarasikan dua variabel bertipe integer, yaitu pilihan untuk menyimpan nomor produk yang dipilih dan jumlah untuk menyimpan jumlah barang yang dibeli. Program kemudian menampilkan daftar produk beserta harga menggunakan fmt.Println. Setelah itu,

fmt.Scan digunakan untuk menerima input pilihan produk dan jumlah pembelian dari pengguna.

Selanjutnya, dideklarasikan variabel nama bertipe string dan harga bertipe integer yang akan diisi berdasarkan pilihan pengguna. Percabangan if–else if digunakan untuk menentukan nama produk dan harga sesuai dengan nomor produk yang dipilih. Setelah nama dan harga diketahui, program menghitung total pembayaran dengan mengalikan harga dan jumlah beli, lalu hasilnya disimpan dalam variabel total.

Pada bagian akhir, program menampilkan struk pembayaran yang berisi nama produk, harga satuan, jumlah pembelian, dan total harga. Dengan alur ini, program berjalan secara berurutan mulai dari menampilkan menu, menerima input, menentukan produk, menghitung total, hingga menampilkan hasil pembayaran.

3. SOAL 3

Source Code

```
package main

import "fmt"

func main() {
    var a, b, c int

    fmt.Scan(&a, &b, &c)

    if a+b <= c || a+c <= b || b+c <= a {
        fmt.Println("Bukan segitiga")
        return
    }

    if a == b && b == c {
        fmt.Println("Segitiga sama sisi")
        return
    }

    if a*a+b*b == c*c || a*a+c*c == b*b || b*b+c*c ==
    a*a {
```

```

        fmt.Println("Segitiga siku-siku")

        return

    }

    if a == b || a == c || b == c {

        fmt.Println("Segitiga sama kaki")

        return

    }

    fmt.Println("Segitiga sembarang")

}

```

Screenshoot program

```

Soal3.go 1 X
Soal3.go > main
14     if a == b && b == c {
15         fmt.Println("Segitiga sama sisi")
16         return
17     }
18
19     if a*a+b*b == c*c || a*a+c*c == b*b || b*b+c*c == a*a {
20         fmt.Println("Segitiga siku-siku")
21         return
22     }

```

```

PS C:\coding didi\golang\Kode Asesmen> go run "c:\coding didi\golang\Kode Asesmen\Soal3.go"
5 5 5
Segitiga sama sisi
PS C:\coding didi\golang\Kode Asesmen> go run "c:\coding didi\golang\Kode Asesmen\Soal3.go"
5 5 8
Segitiga sama kaki
PS C:\coding didi\golang\Kode Asesmen> go run "c:\coding didi\golang\Kode Asesmen\Soal3.go"
3 4 5
Segitiga siku-siku
PS C:\coding didi\golang\Kode Asesmen> go run "c:\coding didi\golang\Kode Asesmen\Soal3.go"
4 5 6
Segitiga sembarang
PS C:\coding didi\golang\Kode Asesmen> go run "c:\coding didi\golang\Kode Asesmen\Soal3.go"
1 2 3

```

Deskripsi program

Program ini dibuat untuk menentukan jenis segitiga berdasarkan tiga bilangan bulat positif yang dimasukkan oleh pengguna, yaitu a, b, dan c, yang merepresentasikan panjang sisi-sisi segitiga. Program diawali dengan membaca input, kemudian

melakukan pengecekan apakah ketiga sisi tersebut memenuhi syarat sebagai segitiga, yaitu jumlah dua sisi harus lebih besar dari sisi yang lain. Jika syarat ini tidak terpenuhi, program langsung menampilkan keluaran "Bukan segitiga" dan berhenti. Apabila memenuhi syarat segitiga, program melanjutkan pengecekan jenis segitiga secara berurutan, dimulai dari segitiga sama sisi dengan membandingkan apakah ketiga sisi memiliki panjang yang sama. Jika tidak, program mengecek apakah segitiga tersebut merupakan segitiga siku-siku menggunakan teorema Pythagoras. Selanjutnya, program memeriksa apakah terdapat dua sisi yang sama panjang untuk menentukan segitiga sama kaki. Jika semua kondisi sebelumnya tidak terpenuhi, maka segitiga tersebut dikategorikan sebagai segitiga sembarang. Alur algoritma disusun secara berurutan dengan percabangan if untuk memastikan setiap kondisi diperiksa secara tepat dan hasil yang ditampilkan sesuai dengan karakteristik sisi-sisi yang diberikan.