

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

ALIN KARISA HIZANNAH

109082500010

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.Print("Masukkan tinggi pohon : ")
    fmt.Scan(&n)

    for n >= 8 {
        fmt.Println("Kurangi tinggi pohon!")
        fmt.Print("Masukkan tinggi pohon : ")
        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 l := 0; l < 2; l++ {

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

            fmt.Print(" ")

        }

        fmt.Println("|")

    }

}

```

Screenshot program

//tambahkan tangkapan layar dari program (boleh lebih dari 1 jika diperlukan)
CONTOH TANGKAPAN LAYAR: (GUNAKAN NOTEPAD)

The screenshot displays three windows side-by-side:

- Code Editor:** Shows the source code for 'soal1.go'. The code uses nested loops to print a Christmas tree pattern of asterisks ('*') based on user input 'n'. It includes error handling for invalid input and a requirement that 'n' must be less than or equal to 8.
- Terminal Window:** Shows two runs of the program. In the first run, it asks for 'tinggi pohon' (tree height) and prints a tree of height 7. In the second run, it asks for 'tinggi pohon' again and prints a tree of height 6.
- Note Pad:** Displays student information: '109082500010', 'IF-13-07', and 'ALIN KARISA HIZANNAH'.

Deskripsi program

Kode di atas digunakan untuk membuat bentuk pohon natal menggunakan lambang asterik ("*"). Ketinggian pohon dinyatakan dalam variabel input `&n`, yang dapat diinputkan oleh user. Dimana ketentuan tingginya telah ditentukan yaitu tinggi pohon harus kurang dari 8 asterik, jika lebih dari itu program akan meminta user mengurangi

ketinggian pohon. Proses menentukan tinggi pada kode di atas menggunakan logika while loop dalam bentuk for $n \geq 8$.

Kemudian untuk membuat bentuk pohon natal menggunakan lambang asterik, digunakan logika nested for loop. Berikut penjelasan dari setiap loopnya

- Perulangan for i digunakan untuk mengatur baris pohon yang sedang dicetak, sekaligus merepresentasikan tinggi pohon dari bagian atas hingga bagian bawah. Setiap kenaikan nilai i menandakan berpindah ke baris berikutnya.
- Di dalam perulangan tersebut, perulangan for j pertama berfungsi untuk mencetak spasi di sebelah kiri pada setiap baris. Jumlah spasi yang dicetak adalah sebanyak $n - i$, sehingga posisi bintang berada di tengah dan membentuk pola piramida.
- Selanjutnya, perulangan for k digunakan untuk mencetak simbol asterisk (*) dengan jumlah $2*i - 1$ pada setiap baris, sehingga lebar pohon bertambah secara bertahap dan membentuk segitiga sama kaki.
- Setelah bagian daun pohon selesai dicetak, program mencetak batang pohon. Perulangan for l digunakan untuk menentukan jumlah baris batang yang ditampilkan, yaitu sebanyak dua baris. Di dalamnya, perulangan for j kembali digunakan untuk mencetak spasi agar simbol batang (|) berada tepat di tengah, sejajar dengan bagian atas pohon.

2. SOAL 2

Source Code

```
package main

import "fmt"

func main() {
    var pilihan, jumlah, harga int
    var namaProduk string

    fmt.Println("==== DAFTAR PRODUK TOKO BUDI ====")
    fmt.Println("1. Little Trees - Rp35.000")
    fmt.Println("2. Lap Microfiber - Rp25.000")
    fmt.Println("3. Cover Steer - Rp150.000")
    fmt.Println("4. Sponge Cuci Mobil - Rp10.000")

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

    for pilihan < 1 || pilihan > 4 {
        fmt.Print("Produk tidak tersedia! Pilih produk (1-4): ")
        fmt.Scan(&pilihan)
    }

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

    switch pilihan {
```

```
case 1:  
    namaProduk = "Little Trees"  
    harga = 35000  
  
case 2:  
    namaProduk = "Lap Microfiber"  
    harga = 25000  
  
case 3:  
    namaProduk = "Cover Steer"  
    harga = 150000  
  
case 4:  
    namaProduk = "Sponge Cuci Mobil"  
    harga = 10000  
}
```

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

```
}
```

Screenshoot program

```

PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Asesemen 2> go run soal2.go
*** DAFTAR PRODUK TOKO BUDI ===
1. Little Trees - Rp35.000
2. Lap Microfiber - Rp25.000
3. Cover Steer - Rp150.000
4. Sponge Cuci Mobil - Rp10.000
Pilih produk (1-4): 1
Masukkan jumlah beli: 5

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

```

Deskripsi program

Kode di atas digunakan untuk membuat program Point of Sale (POS) sederhana. Program ini menampilkan daftar produk beserta harga, kemudian memungkinkan kasir memilih produk dan memasukkan jumlah pembelian untuk menghitung total harga yang harus dibayar.

Pada bagian awal, program menampilkan daftar produk menggunakan beberapa perintah `fmt.Println`. Pilihan produk dimasukkan oleh pengguna dan disimpan dalam variabel `pilihan`. Untuk memastikan bahwa pilihan yang dimasukkan valid, digunakan perulangan `for` dengan kondisi `pilihan < 1 || pilihan > 4`. Selama kondisi tersebut terpenuhi, program akan terus meminta pengguna memasukkan ulang pilihan produk hingga sesuai dengan menu yang tersedia. Perulangan ini berperan sebagai while-loop untuk validasi input.

Setelah pilihan produk valid, pengguna diminta memasukkan jumlah barang yang dibeli, yang disimpan dalam variabel `jumlah`. Selanjutnya, digunakan struktur percabangan `switch-case` untuk menentukan nama produk dan harga berdasarkan nilai `pilihan`. Setiap `case` mewakili satu produk yang tersedia, sehingga pemilihan data menjadi lebih terstruktur dan mudah dibaca.

Setelah harga produk ditentukan, program menghitung total pembayaran dengan mengalikan harga satuan dengan jumlah pembelian. Hasil perhitungan tersebut kemudian ditampilkan dalam bentuk struk pembayaran yang berisi nama produk, harga satuan, jumlah pembelian, dan total harga yang harus dibayar.

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")
    } else if a == b && b == c {
        fmt.Println("Segitiga sama sisi")
    } else if a*a+b*b == c*c || a*a+c*c == b*b || b*b+c*c == a*a {
        fmt.Println("Segitiga siku-siku")
    } else if a == b || a == c || b == c {
        fmt.Println("Segitiga sama kaki")
    } else {
        fmt.Println("Segitiga sembarang")
    }
}
```

Screenshot program

```
Asesemen 2
```

```
Asesemen 2
```

```
soal1.go 2 soal2.go 1 soal3.go 1
```

```
soal3.go > main
```

```
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")
```

```
    } else if a == b && b == c {
```

```
        fmt.Println("Segitiga sama sisi")
```

```
    } else if a*a+b*b == c*c || a*a+c*c == b*b || b*b+c*c ==
```

```
        fmt.Println("Segitiga siku-siku")
```

```
    } else if a == b || a == c || b == c {
```

```
        fmt.Println("Segitiga sama kaki")
```

```
    } else {
```

```
        fmt.Println("Segitiga sembarang")
```

```
}
```

```
powershell
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Asesemen 2> go run soal3.go
```

```
3 3 3
```

```
Segitiga sama sisi
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Asesemen 2> go run soal3.go
```

```
5 5 8
```

```
Segitiga sama kaki
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Asesemen 2> go run soal3.go
```

```
3 4 5
```

```
Segitiga siku-siku
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Asesemen 2> go run soal3.go
```

```
4 5 6
```

```
Segitiga sembarang
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Asesemen 2> go run soal3.go
```

```
1 2 3
```

```
Bukan segitiga
```

```
PS D:\TEL-U\Semester 1\Algoritma Pemrogramman\Asesemen 2> []
```

```
File Edit View Aa 🔍 ⚙️
```

```
109082500010
```

```
IF-13-07
```

```
ALIN KARISA HIZANNAH
```

```
Ln 5, Col 1 44 character Aa For 100% Wind UTF-8
```

```
Launchpad Live Share Reconnect to Discord Quokka Finish Setup
```

Deskripsi program

Program di atas digunakan untuk menentukan jenis segitiga berdasarkan tiga buah sisi yang dimasukkan oleh pengguna. Nilai sisi segitiga dimasukkan melalui variabel a, b, dan c menggunakan satu kali proses input.

Setelah nilai ketiga sisi diterima, program melakukan pengecekan menggunakan struktur percabangan if - else if - else. Pengecekan pertama dilakukan untuk memastikan apakah ketiga sisi tersebut memenuhi syarat sebagai segitiga. Apabila jumlah dua sisi lebih kecil atau sama dengan sisi lainnya, maka ketiga sisi tersebut tidak dapat membentuk segitiga dan program menampilkan keluaran "Bukan segitiga".

Jika syarat segitiga terpenuhi, program kemudian mengecek jenis segitiga secara berurutan. Pengecekan dimulai dari segitiga sama sisi, yaitu ketika ketiga sisi memiliki panjang yang sama. Selanjutnya, program memeriksa apakah segitiga tersebut merupakan segitiga siku-siku dengan menggunakan teorema Pythagoras. Setelah itu, dilakukan pengecekan segitiga sama kaki, yaitu ketika dua sisi memiliki panjang yang sama.

Apabila semua kondisi tersebut tidak terpenuhi, maka segitiga diklasifikasikan sebagai segitiga sembarang karena ketiga sisinya berbeda panjang dan tidak memiliki sudut siku-siku. Hasil klasifikasi segitiga kemudian ditampilkan ke layar sesuai dengan kondisi yang terpenuhi.