

**LAPORAN PRAKTIKUM**  
**Algoritma Pemrograman**

**EVALUASI**



**Disusun oleh:**

**FIRDAUS RAMADHANA**

**109082500151**

**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)

    if n >= 8 {
        fmt.Println("Tinggi pohon harus kurang dari 8")
        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()
    }

    for l := 0; l < 2; l++ {
        for m := 1; m <= n-1; m++ {
            fmt.Print(" ")
        }
        fmt.Println("|")
    }
}
```

## Screenshot program

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

```
func main() {
    var n int
    fmt.Print("Masukkan tinggi pohon: ")
    fmt.Scan(&n)

    if n >= 8 {
        fmt.Println("Tinggi pohon harus kurang dari 8")
        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()
    }
    for l := 0; l < 2; l++ {
        for m := 1; m <= n-1; m++ {
            fmt.Print(" ")
        }
        fmt.Println("|")
    }
}
```

D:\2. PERULIHAWI ASN/GERDHY\Praktikum\go\Assesmen-2\soal1>go run main.go  
Masukkan tinggi pohon: 3  
\*\*\*  
|  
  
D:\2. PERULIHAWI ASN/GERDHY\Praktikum\go\Assesmen-2\soal1>go run main.go  
Masukkan tinggi pohon: 7  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
|

## Deskripsi program

Program diatas adalah program untuk membuat pohon natal, dimana jika kita input sebuah bilangan bulat lebih atau sama dengan 8 maka program tidak akan berjalan. Jika kita input bilangan kurang dari 8 maka program akan berjalan. Dimana cara kerja logicnya yaitu pada loop iterasi i adalah looping untuk barisnya, kemudian untuk iterasi j looping untuk spasi kiri agar rata tengah, untuk loop iterasi k menambahkan asterisk atau (\*) pada program dimana urutanya pasti 1,3,5,7 agar membentuk daun pohon. Untuk iterasi l untuk melakukan looping dan print (|) agar menjadi batang pohon dan rata ditengah.

## 2. SOAL 2

### Source Code

```
package main

import "fmt"

func main() {
    var menu, jumlah, harga int
    var namaBarang string
```

```
fmt.Println("==== DAFTAR PRODUK TOKO BUDI ====")
fmt.Println("1. Little Trees - 35000")
fmt.Println("2. Lap microfiber - 25000")
fmt.Println("3. Cover Steer - 150000")
fmt.Println("4. Sponge cuci mobil - 10000")

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

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

switch menu {
    case 1:
        namaBarang = "Little Trees"
        harga = 35000
    case 2:
        namaBarang = "Lap microfiber"
        harga = 25000
    case 3:
        namaBarang = "Cover Steer"
        harga = 150000
    case 4:
        namaBarang = "Sponge cuci mobil"
        harga = 10000
    default:
        fmt.Println("Menu tidak tersedia")
        return
}

totalHarga := harga * jumlah
```

```

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

        fmt.Println("Produk :", namaBarang)

        fmt.Println("Jumlah :", jumlah)

        fmt.Println("Total : ", totalHarga)

    }

```

## Screenshot program

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

- Code Editor:** The main window displays the source code for a Go program named `main.go`. The code defines a menu with four items: Little Trees (harga 15000), Lap microfiber (harga 25000), Cover Steer (harga 150000), and Sponge cuci mobil (harga 10000). It prompts the user to enter a menu number (1-4) and a quantity, then calculates the total price.
- Terminal:** Below the editor, the terminal window shows the execution of the program. It asks for a menu choice (1-4), receives input '2' for Lap microfiber, asks for quantity (jumlah), receives input '2', and then calculates the total price (125000).
- Output Window:** A separate window titled "FIRDAUS RAMADHANA" displays the results: FIRDAUS RAMADHANA, 109082500151, and IF-13-07.

## Deskripsi program

Program diatas adalah program kasir sederhana dari Toko Budi Racing klub ihir. Cara kerjanya cukup simpel, dimana program menyiapkan variabel untuk menampung pilihan menu, jumlah barang yang dibeli, harga satuan, dan nama barang. Setelah itu, program menampilkan daftar produk beserta harganya agar pengguna tahu opsi yang tersedia. Pengguna kemudian diminta memilih satu menu dengan memasukkan angka 1 sampai 4 dan memasukkan jumlah barang yang ingin dibeli. Pilihan menu yang dimasukkan pengguna diproses menggunakan struktur switch, di mana setiap nomor menu mengarah ke nama barang dan harga yang sesuai. Jika pengguna memasukkan angka di luar pilihan yang tersedia, program langsung menghentikan proses dan menampilkan pesan bahwa menu tidak tersedia, sehingga program salah dan tidak dilanjutkan. Jika pilihan benar, program menghitung total harga dengan mengalikan harga satuan barang dengan jumlah yang dibeli. Terakhir, program menampilkan struk pembayaran yang berisi nama produk, 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.Print("Masukkan tiga sisi: ")
    fmt.Scan(&a, &b, &c)

    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
    }

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

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

## Screenshot program

The screenshot shows a code editor interface with two tabs open: 'main.go' and 'main.go' (with a red circle). The code in 'main.go' is as follows:

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var a, b, c int
7     fmt.Println("Masukkan tiga sisi: ")
8     fmt.Scan(&a, &b, &c)
9
10    if a+b <= c || a+c <= b || b+c <= a {
11        fmt.Println("Bukan segitiga")
12        return
13    }
14    if a == b && b == c {
15        fmt.Println("Segitiga sama sisi")
16        return
17    }
18    if a*a+b*b == c*c || a*a+c*c == b*b || b*b+c*c == a*a {
19        fmt.Println("Segitiga siku-siku")
20        return
21    }
22    if a == b || a == c || b == c {
23        fmt.Println("Segitiga sama kaki")
24        return
25    }
26    fmt.Println("Segitiga sembarang")
27 }

```

Below the code editor is a terminal window titled 'FIRDAUS RAMADHANA' showing the output of the program:

```

FIRDAUS RAMADHANA
File Edit View H1 ...
Ln 3, Col 9 39 characters Plain text 90% Windows (CF) UTF-8
D:\U-PERULIANKU\ASV\GERONY\Praktikum\go\Assesmen-2\soal1>cd ..
D:\U-PERULIANKU\ASV\GERONY\Praktikum\go\Assesmen-2\soal1>soal1
D:\U-PERULIANKU\ASV\GERONY\Praktikum\go\Assesmen-2\soal1>go run main.go
Masukkan tiga sisi: 3 4 5
Segitiga sembarang
D:\U-PERULIANKU\ASV\GERONY\Praktikum\go\Assesmen-2\soal1>go run main.go
Masukkan tiga sisi: 3 4 6
Segitiga siku-siku
D:\U-PERULIANKU\ASV\GERONY\Praktikum\go\Assesmen-2\soal1>go run main.go
Masukkan tiga sisi: 3 5 8
Segitiga sama kaki
D:\U-PERULIANKU\ASV\GERONY\Praktikum\go\Assesmen-2\soal1>

```

## Deskripsi program

Program diatas adalah program untuk mengvalidasi apakah suatu angka yg kita inputkan bisa menjadi segitiga atau bukan, jika iya maka program akan berlanjut di perkondisian. Dimana jika ketiga sisinya sama panjang maka program akan mencetak segitiga sama sisi. Lanjut jika tidak maka akan di cek di perkondisian ketiga yaitu jika memiliki sudut siku-siku dengan membandingkan kuadrat panjang sisi-sisinya sesuai aturan Pythagoras, maka program akan mencetak segitiga siku-siku. Jika tidak, maka akan masuk ke perkondisian ke 4 yaitu, jika dua sisi yang sama panjang. Jika ada, segitiga tersebut termasuk segitiga sama kaki. Jika semua pemeriksaan itu tidak terpenuhi, berarti ketiga sisinya berbeda dan tidak memiliki sifat khusus, sehingga program menyimpulkan bahwa segitiga tersebut adalah segitiga sembarang.