

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

**EVALUASI**



**Disusun oleh:**

**Nabil Nailur Ridho**

**109082530008**

**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

    for {
        fmt.Print("Input ketinggian piramida (N < 8): ")
        fmt.Scan(&n)

        if n < 8 {
            break
        }

        fmt.Println("Ketinggian harus kurang dari 8")
    }

    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.Println(" ")  
        }  
  
        fmt.Println(" | ")  
    }  
}
```

## Screenshot program

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- File Explorer:** On the left, it shows a tree view of files under "TEST" and "UAS". The "UAS" folder contains several Go files: "lat1.go", "lat2.go", "lat3.go", "contoh2.go", "contoh3.go", "forloop.go", "guided1.go", "guided2.go", "guided3.go", "lat1.go", "lat2.go", "lat3.go", "test.go", "Testing.go", "Testing2.go", and "Testing3.go".
- Code Editor:** The main editor area displays the content of "lat1.go". The code prints a pyramid pattern based on user input. It uses nested loops to print asterisks and spaces. The code is as follows:

```
UAS > <lat1.go> main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7
8     for {
9         fmt.Print("Input ketinggian piramida (N < 8): ")
10        fmt.Scan(&n)
11
12        if n < 8 {
13            break
14        }
15        fmt.Println("Ketinggian harus kurang dari 8")
16    }
17
18    for i := 1; i <= n; i++ {
19        for j := 0; j < n-i; j++ {
20            fmt.Print(" ")
21        }
22        for k := 0; k < (2*i - 1); k++ {
23            fmt.Print("*")
24        }
25        fmt.Println()
26    }
27    for i := 0; i < 2; i++ {
28        for j := 0; j < n-1; j++ {
29            fmt.Print(" ")
30        }
31        fmt.Println("|")
32    }
33 }
```

- Terminal:** At the bottom right, there is a terminal window titled "powershell". It shows the output of running the program:

```
Nama : Nabil Nailur Ridho
NIM : 109082530008
Kelas: S1IF-13-07
```

## Deskripsi program

Program ini buat menampilkan pola piramida bintang di layar berdasarkan tinggi piramida yang dimasukkan oleh pengguna melalui input keyboard. Pada awal program pengguna diminta memasukkan nilai tinggi piramida dengan syarat nilai tersebut harus kurang dari 8. Program menggunakan perulangan for tanpa kondisi untuk terus meminta input hingga syarat tersebut terpenuhi, sehingga memastikan validasi data berjalan dengan benar. Setelah mendapatkan nilai yang valid, program memanfaatkan perulangan bersarang untuk mencetak spasi di sebelah kiri dan karakter bintang (\*) di tengah agar membentuk pola piramida yang simetris. Setiap baris piramida dicetak

secara berurutan dari atas ke bawah dengan jumlah bintang yang bertambah, kemudian diakhiri dengan pencetakan dua baris tambahan sebagai alas piramida atau batang pohon natal, sehingga pola yang dihasilkan terlihat lebih menyerupai bentuk piramida atau pohon natal.

## 2. SOAL 2

### Source Code

```
package main

import "fmt"

func main() {
    var pilihan, jumlah, harga, total 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)

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

    switch pilihan {
        case 1:
            namaProduk = "Little Trees"
        case 2:
            namaProduk = "Lap Microfiber"
        case 3:
            namaProduk = "Cover Steer"
        case 4:
            namaProduk = "Sponge Cuci Mobil"
    }

    harga := map[string]int{"Little Trees": 35000, "Lap Microfiber": 25000, "Cover Steer": 150000, "Sponge Cuci Mobil": 10000}
    total = harga[namaProduk] * jumlah
    fmt.Println("Total harga: ", total)
}
```

```
harga = 35000

case 2:

    namaProduk = "Lap Microfiber"

    harga = 25000

case 3:

    namaProduk = "Cover Steer"

    harga = 150000

case 4:

    namaProduk = "Sponge Cuci Mobil"

    harga = 10000

default:

    fmt.Println("Pilihan tidak valid")

    return

}

total = harga * jumlah

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

fmt.Printf("Produk : %s\n", namaProduk)

fmt.Printf("Harga : Rp %d\n", harga)

fmt.Printf("Jumlah : %d\n", jumlah)

fmt.Printf("Total : Rp %d\n", total)

}
```

## Screenshot program

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

- Code Editor:** The main window displays a Go file named `lat1.go` containing a program to calculate the total price of purchases from a store. The code includes a switch statement for product selection and a calculation loop.
- Terminal:** The terminal window shows the execution of the program and its output. It lists four products: Little Trees (Rp35.000), Lap Microfiber (Rp25.000), Cover Steer (Rp150.000), and Sponge Cuci Mobil (Rp10.000). A user inputs quantity 5 for the first product. The program calculates the total price as Rp175.000.
- PowerShell:** An adjacent PowerShell window shows the user's name, NIM, and class information.

## Deskripsi program

Program ini buat implementasi sistem kasir yang bertujuan untuk membantu proses perhitungan transaksi pembelian di sebuah toko. Program menampilkan daftar produk beserta harga dari masing-masing produk agar pengguna dapat memilih barang yang ingin dibeli. Setelah pengguna memasukkan nomor pilihan produk dan jumlah pembelian, program menggunakan struktur percabangan switch untuk menentukan nama produk dan harga satuan sesuai dengan pilihan tersebut. Selanjutnya, program menghitung total pembayaran dengan mengalikan harga produk dan jumlah yang dibeli. Sebagai hasil akhir, program menampilkan struk pembayaran yang berisi informasi lengkap, seperti nama produk, harga satuan, jumlah barang yang dibeli, serta total harga yang harus dibayarkan, sehingga pengguna dapat memahami detail transaksi secara jelas.

### 3. SOAL 3

#### Source Code

```
package main
```

```
import "fmt"
```

```
func main() {
```

```
var a, b, c int

fmt.Print("Masukkan tiga bilangan: ")

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

        fmt.Println("segitiga sama kaki")

    } 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 {

        fmt.Println("segitiga sembarang")

    }

}

}
```

## Screenshot program

```
UAS > lat3.go 1
1 package main
2
3 import "fmt"
4
5 func main() {
6     var a, b, c int
7     fmt.Println("Masukkan tiga bilangan: ")
8     fmt.Scan(&a, &b, &c)
9
10    if a+b <= c || a+c <= b || b+c <= a {
11        fmt.Println("bukan segitiga")
12    } else {
13        if a == b && b == c {
14            fmt.Println("segitiga sama sisi")
15        } else if a == b || a == c || b == c {
16            fmt.Println("segitiga sama kaki")
17        } else if (a*a + b*b == c*c) || (a*a + c*c == b*b) || (b*b + c*c == a*a) {
18            fmt.Println("segitiga siku-siku")
19        } else {
20            fmt.Println("segitiga sembarang")
21        }
22    }
23
24
25
26 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Test\UAS> go run lat3.go
Masukkan tiga bilangan: 3 3 3
segitiga sama sisi
PS C:\Test\UAS> go run lat3.go
Masukkan tiga bilangan: 5 5 8
segitiga sama kaki
PS C:\Test\UAS> go run lat3.go
Masukkan tiga bilangan: 3 4 5
segitiga siku-siku
PS C:\Test\UAS> go run lat3.go
Masukkan tiga bilangan: 4 5 6
segitiga sembarang
PS C:\Test\UAS> go run lat3.go
Masukkan tiga bilangan: 1 2 3
bukan segitiga
PS C:\Test\UAS>
```

## Deskripsi program

Program ini dibuat untuk mengklasifikasikan jenis segitiga berdasarkan tiga buah bilangan yang dimasukkan oleh pengguna sebagai panjang sisi segitiga. Program diawali dengan proses input tiga nilai dan dilanjutkan dengan pengecekan apakah ketiga sisi tersebut memenuhi syarat pembentukan segitiga, yaitu jumlah dua sisi harus lebih besar dari sisi lainnya. Jika syarat tersebut tidak terpenuhi, program akan menampilkan bahwa bangun tersebut bukan segitiga. Namun, jika valid, program akan melanjutkan ke proses klasifikasi menggunakan percabangan logika untuk menentukan apakah segitiga tersebut merupakan segitiga sama sisi, segitiga sama kaki, segitiga siku-siku berdasarkan teorema Pythagoras, atau segitiga sembarang. Hasil klasifikasi kemudian ditampilkan pada output terminal sehingga pengguna dapat langsung mengetahui jenis segitiga dari nilai sisi yang dimasukkan.