

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

Husni Rizal Sahbana

109082500042

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 masukan_bilangan_n int

    for {
        fmt.Print("Masukkan tinggi pohon natal : ")
        fmt.Scan(&masukan_bilangan_n)
        if masukan_bilangan_n < 8 {
            break
        }
        fmt.Println("Tolong masukan tinggi harus kurang dari 8")
    }

    for i := 1; i <= masukan_bilangan_n; i++ {
        for s := masukan_bilangan_n - i; s > 0; s-- {
            fmt.Print(" ")
        }
        for b := 1; b <= 2*i-1; b++ {
            fmt.Print("*")
        }
        fmt.Println()
    }
}
```

The screenshot shows a Go IDE with a file named `soal-1.go`. The code defines a `main` function that reads an integer `masukan_bilangan_n` and prints a Christmas tree of height `n`. The tree is composed of asterisks (`*`) forming a triangle, with a row of stars (`*****`) at the base. The IDE interface includes tabs for `PROBLEMS`, `OUTPUT`, `DEBUG CONSOLE`, `TERMINAL`, and `PORTS`. The `TERMINAL` tab is active, showing the command `go run "c:\LAPRAK ALPRO\UAS\soal-1.go"` and the program's output, which includes the prompt "Masukkan tinggi pohon natal : 9" and the resulting tree structure.

Program tersebut digunakan untuk membuat tampilan pola **pohon Natal** menggunakan karakter bintang (*) dengan tinggi yang di inputkan oleh user. Pertama, program meminta pengguna memasukkan tinggi pohon dan akan terus mengulang input selama nilai yang dimasukkan **tidak kurang dari 8**, sehingga hanya nilai di bawah 8 yang akan di eksekusi. Setelah tinggi yang valid diperoleh, program menjalankan perulangan untuk mencetak baris demi baris pohon. Pada setiap baris, program mencetak spasi terlebih dahulu agar bintang berada di tengah, lalu mencetak bintang dengan jumlah ganjil yang bertambah di setiap baris ($2*i - 1$). Proses ini menghasilkan pola segitiga yang rapi dan simetris ke atas, menyerupai bentuk pohon Natal sesuai dengan tinggi yang dimasukkan user.

Source Code

```
package main

import "fmt"

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

```
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 : ")
fmt.Scan(&pilihan)


fmt.Print("Masukkan jumlah produk yang akan di beli
: ")
fmt.Scan(&jumlah)


if pilihan == 1 {
    namaProduk = "Little Trees"
    harga = 35000
} else if pilihan == 2 {
    namaProduk = "Lap Microfiber"
    harga = 25000
} else if pilihan == 3 {
    namaProduk = "Cover Steer"
    harga = 150000
} else if pilihan == 4 {
    namaProduk = "Sponge Cuci Mobil"
    harga = 10000
} else {
    fmt.Println("Maaf pilihan tidak valid")
    return
}


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

The screenshot shows a Go IDE with the following code in `soal-2.go`:

```

1  func main() {
2      fmt.Scan(&jumlah)
3
4      if pilihan == 1 {
5          namaProduk = "Little Trees"
6          harga = 35000
7      } else if pilihan == 2 {
8          namaProduk = "Lap Microfiber"
9          harga = 25000
10     }
11
12     // ... (rest of the code)
13 }

```

The terminal output shows the program execution:

```

ACER@LAPTOP-IFL4Q4E6 MINGW64 /c/LAPRAK ALPRO/UAS
$ go run "c:\LAPRAK ALPRO\UAS\soal-2.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
Masukkan jumlah produk yang akan di beli : 5

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

```

An inset window shows the generated receipt details:

```

NIM : 109082500042
Kelas : SIIF-13-07
Nama : Husni Rizal Sahbana

```

Deskripsi program

Program tersebut digunakan untuk mencontohkan pembelian produk di sebuah toko secara sederhana. Pada awal program, sistem menampilkan daftar produk yang tersedia di Toko Budi beserta harga masing-masing produk agar pengguna mengetahui pilihan yang bisa dibeli. Setelah itu, pengguna diminta memasukkan nomor produk yang ingin dibeli serta jumlah barang yang akan dibeli. Program kemudian memproses pilihan tersebut menggunakan percabangan if-else untuk menentukan nama produk dan harga yang sesuai dengan nomor yang dipilih. Jika pengguna memasukkan nomor produk yang tidak ada di daftar, program akan menampilkan pesan bahwa pilihan tidak valid dan langsung menghentikan proses agar tidak terjadi kesalahan perhitungan. Apabila pilihan produk valid, program menghitung total pembayaran dengan cara mengalikan harga satuan produk dengan jumlah yang dibeli. Terakhir, program menampilkan struk pembayaran yang berisi informasi lengkap seperti nama produk, harga per unit, jumlah pembelian, serta total biaya yang harus dibayar, sehingga pengguna dapat melihat ringkasan transaksinya.

3. SOAL 3

Source Code

```
package main

import "fmt"

func main() {
    var a, b, c int
    fmt.Print("Masukan 3 bilangan : ")
    fmt.Scan(&a, &b, &c)

    if a > b {
        a, b = b, a
    }
    if b > c {
        b, c = c, b
    }
    if a > b {
        a, b = b, a
    }

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

```

```
fmt.Println("Bukan segitiga")
```

```
}
```

```
}
```

Screenshoot program

The screenshot shows a Go program in an IDE and its execution output in a terminal. The program is named `soal-3.go` and is located in the directory `c:\LAPRAK ALPRO\UAS`. The code defines a `main` function that reads three integers from the user and checks if they form a triangle. The output shows three test cases: 1) Input 3 3 3 results in "Segitiga sama sisi". 2) Input 1 2 3 results in "Bukan segitiga". 3) Input 3 4 3 results in "Segitiga sama kaki".

```
soal-3.go 1 X
soal-3.go > main
5 func main() {
8     fmt.Scan(&a, &b, &c)
9
10    if a > b {
11        a, b = b, a
12    }
13    if b > c {
14        b, c = c, b
15    }
16 }

ACER@LAPTOP-IFL4Q4E6 MINGW64 /c/LAPRAK ALPRO/UAS
$ go run "c:\LAPRAK ALPRO\UAS\soal-3.go"
Masukan 3 bilangan : 3 3 3
Segitiga sama sisi

ACER@LAPTOP-IFL4Q4E6 MINGW64 /c/LAPRAK ALPRO/UAS
$ go run "c:\LAPRAK ALPRO\UAS\soal-3.go"
Masukan 3 bilangan : 1 2 3
Bukan segitiga

ACER@LAPTOP-IFL4Q4E6 MINGW64 /c/LAPRAK ALPRO/UAS
$ go run "c:\LAPRAK ALPRO\UAS\soal-3.go"
Masukan 3 bilangan : 3 4 3
Segitiga sama kaki

ACER@LAPTOP-IFL4Q4E6 MINGW64 /c/LAPRAK ALPRO/UAS
$
```

Deskripsi program

Program tersebut digunakan untuk menentukan **jenis segitiga** berdasarkan tiga bilangan yang dimasukkan pengguna. Pertama, program meminta pengguna memasukkan tiga bilangan yang merepresentasikan panjang sisi segitiga, lalu ketiga bilangan tersebut diurutkan dari yang paling kecil ke paling besar menggunakan pertukaran nilai ($a, b = b, a$) agar proses pengecekan lebih mudah dan terstruktur. Setelah urutan sisi benar, program memeriksa apakah ketiga sisi tersebut bisa membentuk segitiga dengan syarat jumlah dua sisi terkecil harus lebih besar dari sisi terbesar ($a + b > c$). Jika syarat ini tidak terpenuhi, maka program langsung menyatakan "Bukan segitiga". Jika memenuhi, program melanjutkan untuk menentukan jenis segitiga: jika ketiga sisi sama, maka segitiga tersebut adalah segitiga sama sisi; jika memenuhi rumus Pythagoras ($a^2 + b^2 = c^2$), maka segitiga tersebut adalah segitiga siku-siku; jika hanya dua sisi yang sama, maka termasuk segitiga sama kaki; dan jika semua sisi berbeda serta tidak memenuhi kondisi sebelumnya, maka segitiga tersebut adalah segitiga sembarang. Dengan alur ini, program mampu mengklasifikasikan segitiga secara logis dan sistematis berdasarkan panjang sisi yang diberikan pengguna.