

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

Rafi Oktarino Ramadhan

109082500217

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.Println("Masukkan tinggi pohon (harus lebih dari 8): ")
        fmt.Scan(&n)

        if n > 8 {
            break
        }

        fmt.Println("Tinggi pohon harus lebih dari 8. Silakan coba lagi.")
    }

    for i := 1; i <= n; i++ {
        for j := i; j < n; j++ {
            fmt.Print(" ")
        }
        fmt.Println()
    }
}
```

```
}
```



```
for k := 1; k <= (2*i - 1); k++ {
```



```
    fmt.Println("*")
```



```
}
```



```
    fmt.Println()
```



```
}
```



```
}
```

Screenshot program

The screenshot shows a Go development environment in Visual Studio Code. The code editor displays a file named `soal1.go` with the following content:

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

The terminal window shows the execution of the program and its output:

```
PS E:\GO\assesment 2> go run soal1.go
Masukkan tinggi pohon (harus lebih dari 8): 9
*
 ***
 ****
 *****
```

A floating terminal window shows the user's profile information:

```
Rafi Oktarino Ramadhan
109082500217
S1-IF-07
```

Deskripsi program

Program Go ini diawali dengan package `main` sebagai penanda program utama dan import `"fmt"` untuk menggunakan fungsi input–output. Di dalam fungsi `main`, variabel `n` digunakan untuk menyimpan tinggi pohon. Program kemudian menjalankan perulangan `for` tanpa kondisi untuk meminta input tinggi pohon dari pengguna hingga nilainya lebih dari 8. Jika input belum memenuhi syarat, program akan menampilkan pesan peringatan dan meminta input ulang. Setelah tinggi pohon valid, perulangan `for` berikutnya digunakan untuk mencetak pohon natal. Perulangan pertama mengatur

jumlah baris, perulangan kedua mencetak spasi agar pohon berbentuk segitiga simetris, dan perulangan ketiga mencetak simbol bintang (*) yang jumlahnya bertambah di setiap baris. Hasil akhirnya adalah tampilan pohon natal berbentuk segitiga.

2. SOAL 2

Source Code

```
package main

import "fmt"

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

    fmt.Println("==== MENU EMPAL GENTONG MAS FUAD ====")
    fmt.Println("1. Empal Gentong Biasa - Rp22000")
    fmt.Println("2. Empal Asem - Rp22000")
    fmt.Println("3. Sate Kambing Muda - Rp40000")
    fmt.Println("4. Nasi Lengko Pagongan - Rp15000")

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

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

    if pilihan == 1 {
```

```
namaMenu = "Empal Gentong Biasa"

harga = 22000

} else if pilihan == 2 {

    namaMenu = "Empal Asem"

    harga = 22000

} else if pilihan == 3 {

    namaMenu = "Sate Kambing Muda"

    harga = 40000

} else if pilihan == 4 {

    namaMenu = "Nasi Lengko Pagongan"

    harga = 15000

} else {

    fmt.Println("Menu tidak tersedia")

    return

}

total := harga * jumlah

fmt.Println()

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

fmt.Println("Menu      :", namaMenu)

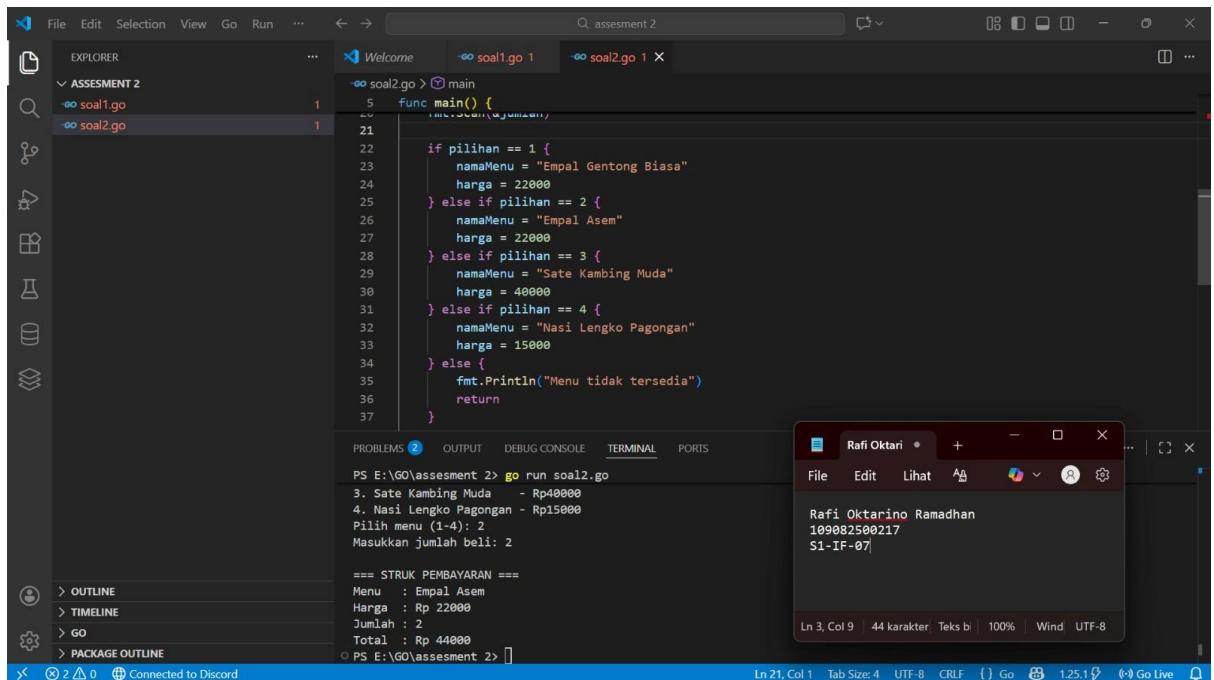
fmt.Println("Harga   : Rp", harga)

fmt.Println("Jumlah  :", jumlah)

fmt.Println("Total   : Rp", total)

}
```

Screenshot program



Deskripsi program

Program Go ini merupakan program POS (Point of Sale) sederhana untuk membantu kasir mencatat pesanan dan menghitung total pembayaran. Program dimulai dengan package main dan import "fmt" untuk keperluan input dan output. Di dalam fungsi main, beberapa variabel disiapkan untuk menyimpan pilihan menu, jumlah beli, nama menu, dan harga. Program kemudian menampilkan daftar menu beserta harganya, lalu meminta kasir memilih menu dan memasukkan jumlah pembelian. Menggunakan percabangan if–else, program menentukan nama menu dan harga berdasarkan pilihan yang dimasukkan. Jika pilihan tidak valid, program akan menampilkan pesan kesalahan dan berhenti. Setelah itu, total pembayaran dihitung dari hasil perkalian harga dan jumlah beli, lalu ditampilkan dalam bentuk struk sederhana yang berisi menu yang dipesan, harga satuan, jumlah, dan total 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")
    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")  
}
```

Screenshot program

The screenshot shows a code editor interface with several tabs open. The main tab displays a Go program named 'soal3.go' with the following code:

```
func main() {  
    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("Bukan segitiga")  
}
```

The terminal tab shows the execution of the program with different input values:

- PS E:\GO\assesment > go run soal3.go
3 3 3
Segitiga sama sisi
- PS E:\GO\assesment > go run soal3.go
5 5 8
Segitiga sama kaki
- PS E:\GO\assesment > go run soal3.go
3 4 5
Segitiga siku-siku
- PS E:\GO\assesment > go run soal3.go
4 5 6
Segitiga sembarang
- PS E:\GO\assesment > go run soal3.go
1 2 3
Bukan segitiga

A small window titled "Rafi Oktari" is also visible, displaying the user's name, ID, and student number.

Deskripsi program

Program Go ini digunakan untuk menentukan jenis segitiga berdasarkan tiga panjang sisi yang dimasukkan oleh pengguna. Program diawali dengan package main dan import "fmt" untuk menjalankan program utama serta membaca dan menampilkan data. Di dalam fungsi main, tiga variabel a, b, dan c digunakan untuk menyimpan nilai sisi segitiga yang dibaca menggunakan fmt.Scan. Program pertama-tama mengecek apakah ketiga sisi memenuhi syarat membentuk segitiga, yaitu jumlah dua sisi harus lebih besar dari sisi lainnya; jika tidak, program menampilkan "Bukan segitiga" dan berhenti. Jika memenuhi syarat, program kemudian mengecek secara berurutan apakah segitiga tersebut merupakan segitiga sama sisi, segitiga siku-siku dengan rumus Pythagoras, atau segitiga sama kaki. Jika tidak termasuk ketiganya, maka hasil akhirnya adalah segitiga sembarang, dan jenis segitiga tersebut ditampilkan ke layar.