

LAPORAN PRAKTIKUM
Algoritma Pemrograman

Assesment 2

EIKEL PRINST SUKATENDEL



Disusun oleh:

EIKEL PRINST SUKATENDEL

103112430232

S1IF-13-07

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

TUGAS PENDAHULUAN

1. Tugas Pendahuluan 1

Source code

```
package main

import (
    "bufio"
    "fmt"
    "os"
)

func main() {
    reader := bufio.NewReader(os.Stdin)
    var n int

    for {
        fmt.Print("Masukkan tinggi pohon (N > 8): ")
        fmt.Fscan(reader, &n)

        if n > 8 {
            break
        }

        fmt.Println("Tinggi pohon harus lebih dari 8, silakan input ulang.")
    }

    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()

    }

}

```

Screenshoot program

The screenshot shows a Go IDE with the following components:

- EXPLORER:** Shows the project structure with 'ASSESSMENT' and 'Soal1.go'.
- EDITOR:** Displays the code for 'Soal1.go'. The code includes a 'main' function that reads input 'n' and prints a Christmas tree pattern using asterisks.
- TERMINAL:** Shows the program's execution. It prompts 'Masukkan tinggi pohon (N > 8): 10' and displays the resulting tree pattern, which is 10 rows high and symmetric.

Deskripsi program

Program di atas berfungsi untuk mencetak pola pohon natal menggunakan karakter bintang (*) berdasarkan tinggi yang dimasukkan oleh pengguna. Pada awal program, digunakan `bufio.NewReader(os.Stdin)` agar proses pembacaan input lebih stabil. Program kemudian meminta pengguna memasukkan nilai `n` sebagai tinggi pohon, dengan syarat nilai tersebut harus lebih dari 8. Jika pengguna memasukkan nilai yang kurang atau sama dengan 8, program akan menampilkan pesan kesalahan dan meminta input ulang hingga syarat terpenuhi.

Setelah mendapatkan nilai tinggi yang valid, program masuk ke proses pencetakan pola pohon natal. Perulangan pertama (`for i := 1; i <= n; i++`) digunakan untuk mengatur jumlah baris pohon. Di dalamnya, terdapat perulangan untuk mencetak spasi di sebelah kiri agar bentuk pohon terlihat simetris, serta perulangan lain untuk mencetak bintang dengan rumus $(2*i - 1)$ sehingga setiap baris membentuk segitiga.

yang semakin lebar ke bawah. Hasil akhirnya adalah pola pohon natal berbentuk segitiga sama kaki dengan tinggi sesuai input pengguna.

2. Tugas Pendahuluan 2

Source code

```
package main

import "fmt"

func main() {

    menu := []string{
        "Empal Gentong Biasa",
        "Empal Asem",
        "Sate Kambing Muda",
        "Nasi Lengko Pagongan",
    }

    harga := []int{
        22000,
        22000,
        40000,
        15000,
    }

    var pilihMenu int
    var jumlah int

    fmt.Println("=== MENU EMPAL GENTONG MAS FUAD ===")
    for i := 0; i < len(menu); i++ {
        fmt.Printf("%d. %s - Rp%d\n", i+1, menu[i], harga[i])
    }

    fmt.Print("Pilih menu (1-4): ")
}
```

```

    fmt.Scan(&pilihMenu)

    fmt.Print("Masukkan jumlah beli: ")

    fmt.Scan(&jumlah)

    total := harga[pilihMenu-1] * jumlah

    fmt.Println("\n=== STRUK PEMBAYARAN ===")
    fmt.Println("Menu      :", menu[pilihMenu-1])
    fmt.Println("Harga   : Rp", harga[pilihMenu-1])
    fmt.Println("Jumlah  :", jumlah)
    fmt.Println("Total   : Rp", total)
}

```

Screenshoot program

The screenshot shows a Go IDE with the following components:

- EXPLORER:** Shows a project named 'ASSESSMENT' with two files: 'Soal1.go' and 'Soal2.go'.
- EDITOR:** Displays the source code for 'Soal2.go'. The code defines a menu with four items and their prices, and a main function that prompts the user for a menu choice and the quantity to buy.
- TERMINAL:** Shows the output of running the program. It displays the menu, the user's selection (menu 3, quantity 2), and the calculated total (Rp 80000).

```

// Soal2.go
package main

import "fmt"

func main() {
    // Data menu
    menu := []string{
        "Empal Gentong Biasa",
        "Empal Asem",
        "Sate Kambing Muda",
        "Nasi Lengko Pagongan",
    }

    // ... (rest of the code)
}

// Terminal Output
PS D:\ALPRO 3 Ulangan\Assesment> go run "d:\ALPRO 3 Ulangan\Assesment\Soal2.go"
=== MENU EMPAL GENTONG MAS FUAD ===
1. Empal Gentong Biasa - Rp22000
2. Empal Asem - Rp22000
3. Sate Kambing Muda - Rp40000
4. Nasi Lengko Pagongan - Rp15000
Pilih menu (1-4): 3
Masukkan jumlah beli: 2

=== STRUK PEMBAYARAN ===
Menu      : Sate Kambing Muda
Harga   : Rp 40000
Jumlah  : 2
Total   : Rp 80000
PS D:\ALPRO 3 Ulangan\Assesment>

```

Deskripsi program

Program di atas merupakan contoh sederhana aplikasi Point of Sale (POS) menggunakan bahasa Go. Program ini menampilkan daftar menu makanan beserta harga, kemudian kasir diminta untuk memilih menu yang diinginkan dan memasukkan jumlah porsi yang dibeli. Input dari pengguna disimpan menggunakan variabel bertipe integer, sedangkan nama menu disimpan dalam variabel string.

Proses pemilihan menu dilakukan menggunakan percabangan switch-case agar kode lebih rapi dan mudah dibaca. Setelah menu dan harga ditentukan, program akan menghitung total pembayaran dengan mengalikan harga satuan dengan jumlah beli. Terakhir, program menampilkan struk pembayaran yang berisi nama menu, harga, jumlah, dan total yang harus dibayar. Program ini cocok untuk pemula karena strukturnya sederhana dan mudah dipahami.

3. Tugas Pendahuluan 3

Source code

```
package main

import "fmt"

func main() {
    var a, b, c int

    fmt.Print("Masukkan sisi a: ")
    fmt.Scan(&a)
    fmt.Print("Masukkan sisi b: ")
    fmt.Scan(&b)
    fmt.Print("Masukkan sisi c: ")
    fmt.Scan(&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")
    } 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 {

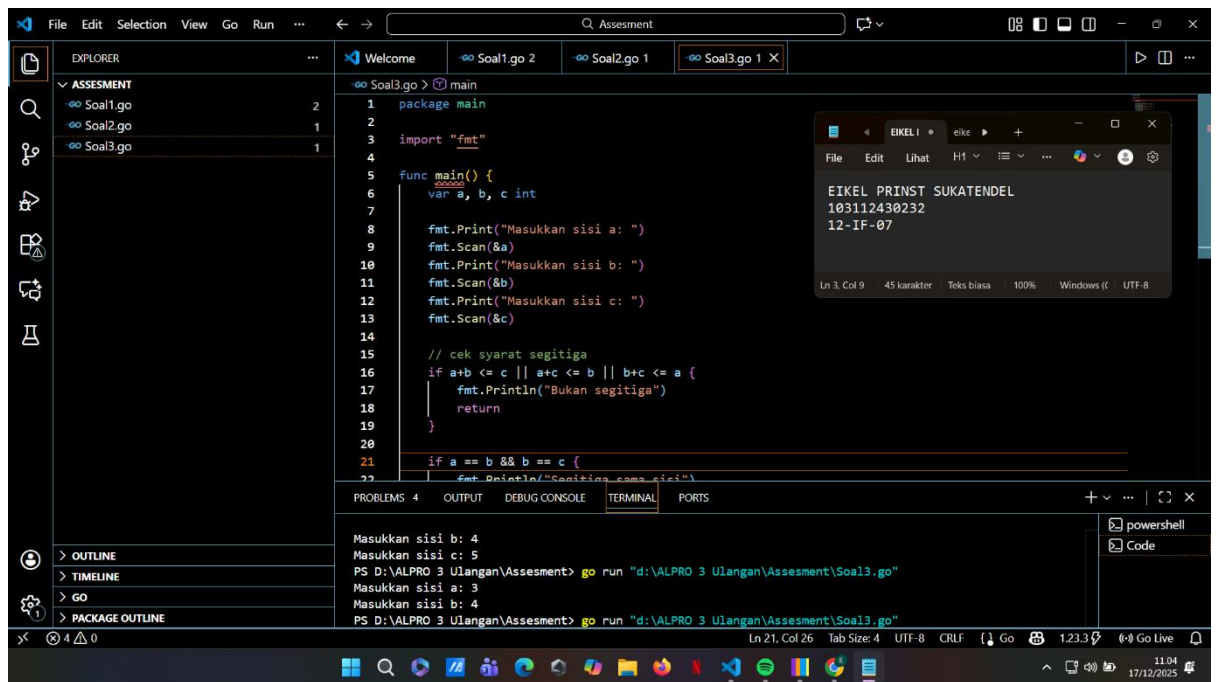
        fmt.Println("Segitiga sembarang")

    }

}

```

Screenshoot program



```

PS D:\ALPRO 3 Ulangan\Assesment> go run "d:\ALPRO 3 Ulangan\Assesment\Soal13.go"
Masukkan sisi a: 3
Masukkan sisi b: 3
Masukkan sisi c: 3
Segitiga sama sisi
PS D:\ALPRO 3 Ulangan\Assesment> go run "d:\ALPRO 3 Ulangan\Assesment\Soal13.go"
Masukkan sisi a: 5
Masukkan sisi b: 5
Masukkan sisi c: 8
Segitiga sama kaki
PS D:\ALPRO 3 Ulangan\Assesment> go run "d:\ALPRO 3 Ulangan\Assesment\Soal13.go"
Masukkan sisi a: 3
Masukkan sisi b: 4
Masukkan sisi c: 5
Segitiga siku-siku
PS D:\ALPRO 3 Ulangan\Assesment> go run "d:\ALPRO 3 Ulangan\Assesment\Soal13.go"
Masukkan sisi a: 4
Masukkan sisi b: 5
Masukkan sisi c: 6
Segitiga sembarang
PS D:\ALPRO 3 Ulangan\Assesment> go run "d:\ALPRO 3 Ulangan\Assesment\Soal13.go"
Masukkan sisi a: 1
Masukkan sisi b: 2
Masukkan sisi c: 3
Bukan segitiga
PS D:\ALPRO 3 Ulangan\Assesment> █

```

Deskripsi program

Program ini digunakan untuk menentukan jenis segitiga berdasarkan tiga sisi yang dimasukkan oleh pengguna. Pertama, program menerima tiga bilangan bulat positif sebagai panjang sisi segitiga. Setelah itu, program akan mengecek apakah ketiga sisi tersebut memenuhi syarat membentuk segitiga, yaitu jumlah dua sisi harus lebih besar dari sisi yang lain. Jika syarat ini tidak terpenuhi, maka program langsung menampilkan output "Bukan segitiga".

Apabila ketiga sisi memenuhi syarat segitiga, program akan melanjutkan ke proses pengecekan jenis segitiga menggunakan percabangan if-else. Jika semua sisi sama panjang, maka segitiga tersebut adalah segitiga sama sisi. Jika hanya dua sisi yang sama, maka termasuk segitiga sama kaki. Program juga mengecek teorema Pythagoras untuk menentukan segitiga siku-siku. Jika tidak memenuhi semua kondisi tersebut, maka segitiga dikategorikan sebagai segitiga sembarang.