

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1**

**MODUL 13
KOMPOSISI**



Disusun Oleh :

Estetika Ananda Poetri Hariyanto / 103112400272

IF-12-05

Asisten Praktikum :

Ayu Susilowati

Noviana Rizki Anisa Putri

Dosen Pengampu :

Yudha Islami Sulistya

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

TUGAS PENDAHULUAN

A. PRAKTIKUM (Soal Contoh pada Modul)

Soal Studi Case

1. Buatlah program untuk menampilkan bilangan sejumlah bilangan prima

Petunjuk :

Inputan berupa bilangan bulat dari user

Output berupa barisan bilangan prima dari 1 hingga bilangan yang diinputkan user

Sourcecode

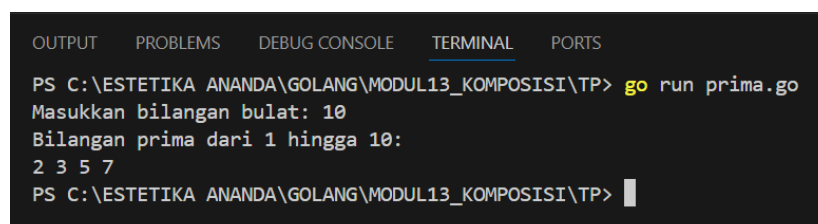
```
package main

import (
    "fmt"
)

func isPrime(num int) bool {
    if num < 2 {
        return false
    }
    for i := 2; i*i <= num; i++ {
        if num%i == 0 {
            return false
        }
    }
    return true
}

func main() {
    var n int
    fmt.Print("Masukkan bilangan bulat: ")
    fmt.Scan(&n)
    if n < 1 {
        fmt.Println("Masukkan bilangan bulat yang lebih besar dari 0.")
        return
    }
    fmt.Printf("Bilangan prima dari 1 hingga %d:\n", n)
    for i := 1; i <= n; i++ {
        if isPrime(i) {
            fmt.Print(i, " ")
        }
    }
    fmt.Println()
}
```

Screenshoot Output



```
OUTPUT  PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\ESTETIKA ANANDA\GOLANG\MODUL13_KOMPOSISI\TP> go run prima.go
Masukkan bilangan bulat: 10
Bilangan prima dari 1 hingga 10:
2 3 5 7
PS C:\ESTETIKA ANANDA\GOLANG\MODUL13_KOMPOSISI\TP> █
```

Deskripsi Program

Program ini dibuat untuk menampilkan bilangan sejumlah bilangan prima, berupa barisan bilangan prima dari 1 hingga bilangan yang diinputkan user.

- Algoritma:
 - Program minta bilangan bulat N.
 - Kalau $N < 1$, maka program berhenti dengan pesan bahwa input harus lebih besar dari 0.
 - Pengecekan bilangan prima menggunakan isPrime.
 - Menampilkan bil prima.

- Cara Kerja Program:
 - Pengguna diminta memasukan angka, contoh $N = 10$
 - Validasi input kurang 1 program berhenti. Contoh $N = -5$, program akan mencetak (masukan bilangan bulat yang lebih besar dari 0.
 - Perulangan program dimulai dari 1 hingga 10. Contoh : $I = 1$: tidak prima karena kurang dari 2.
 - Nah jadi jika bilangan bulat 10, bilangan prima dari 1 hingga 10 adalah 2 3 5 7.

- Output yang dihasilkan :
Masukan bilangan bulat : 10
Bilangan prima dari 1 hingga 10 :
2 3 5 7

B. TUGAS (Soal Latihan pada Modul)

Soal Studi Case

2. Bilangan sempurna adalah bilangan yang sama dengan jumlah dari factor-faktornya. Buatlah program untuk menentukan apakah sebuah bilangan merupakan bilangan sempurna

Contoh

Inputan : 6

Output : Ya (karena factor dari 6 yaitu 1,2,3 dan $1+2+3 = 6$)

Sourcecode

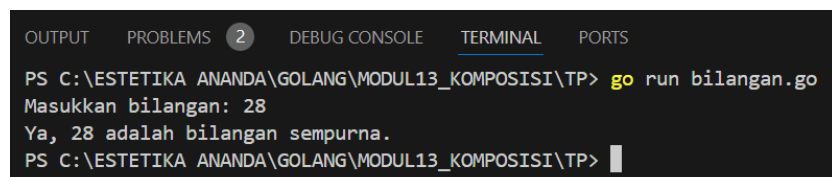
```
package main

import (
    "fmt"
)

func sumOfFactors(num int) int {
    sum := 0
    for i := 1; i < num; i++ {
        if num%i == 0 {
            sum += i
        }
    }
    return sum
}

func main() {
    var n int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)
    if n <= 0 {
        fmt.Println("Masukkan bilangan positif.")
        return
    }
    if sumOfFactors(n) == n {
        fmt.Printf("Ya, %d adalah bilangan sempurna.\n", n)
    } else {
        fmt.Printf("Tidak, %d bukan bilangan sempurna.\n", n)
    }
}
```

Screenshoot Output



```
PS C:\ESTETIKA ANANDA\GOLANG\MODUL13_KOMPOSISI\TP> go run bilangan.go
Masukkan bilangan: 28
Ya, 28 adalah bilangan sempurna.
PS C:\ESTETIKA ANANDA\GOLANG\MODUL13_KOMPOSISI\TP> 
```

Deskripsi Program

Jadi program ini untuk menentukan bilangan sempurna.

➤ Algoritma:

- Program minta pengguna untuk memasukan sebuah bilangan bulat positif N.
- Jika N kurang dari 0, maka menampilkan pesan bilangan harus positif.

- Pencarian factor factor bilangan.
- Pemeriksaan bilangan sempurna.
- Output.

➤ Cara Kerja Program:

- Fungsi sumOfFactors = menghitung jumlah factor bilangan N kecuali bilangan itu sendiri.
- Program meminta pengguna memasukan N
- Kalau N kurang dari sama dengan 0, maka langsung berhenti.
- Program memanggil sumOfFactors untuk menghitung jumlah factor.
- Kalau hasilnya sama dengan N program mencetak N adalah bilangan sempurna.
- Jika tidak, program mencetak bahwa N jumlahnya tidak sempurna.

➤ Output yang dihasilkan :

Masukan bilangan : 28

Ya, 28 adalah bilangan sempurna.