

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 1**

**MODUL 13**



**Disusun Oleh :**

**Gustaf Adiyatma Alfito /103112400266**

**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 Menentukan Bilangan Prima Ke-N

### Soal Studi Case

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"

// Fungsi untuk memeriksa apakah angka adalah bilangan
prima func isPrime(n int) bool {    if n <= 1 {
return false
    }    for i := 2; i*i <=
n; i++ {        if n%i == 0 {
return false
        }
    }
    return true
}

// Fungsi untuk menghasilkan bilangan prima hingga batas
tertentu func generatePrimes(limit int) []int {    var primes
[]int    for i := 2; i <= limit; i++ {        if isPrime(i) {
primes = append(primes, i)
        }    }
return primes
}

func main() {    var n int
fmt.Print("Masukkan sebuah bilangan bulat: ")
fmt.Scan(&n)
    primes :=
generatePrimes(n)
    fmt.Println("Bilangan prima sampai", n, ":", primes)
}
```

## Screenshoot Output

```
PS C:\Users\ASUS> go run "C:\Users\ASUS\AppData\Local\Temp\tempCodeRunnerFile.go"
Masukkan sebuah bilangan bulat: 8
Bilangan prima sampai 8 : [2 3 5 7]
PS C:\Users\ASUS> |
```

## Deskripsi Program

Program ini bertujuan untuk menghasilkan dan menampilkan bilangan prima hingga batas yang dimasukkan oleh pengguna. Program dimulai dengan mendefinisikan fungsi `isPrime(n int) bool`, yang digunakan untuk memeriksa apakah suatu angka `n` adalah bilangan prima dengan cara memeriksa apakah angka tersebut hanya bisa dibagi oleh 1 dan dirinya sendiri. Selanjutnya, ada fungsi `generatePrimes(limit int) []int` yang menghasilkan daftar bilangan prima mulai dari 2 hingga batas `limit` yang diberikan, dengan memanfaatkan fungsi `isPrime` untuk memeriksa setiap angka dalam rentang tersebut. Pada bagian utama program, fungsi `main` meminta input bilangan bulat dari pengguna, kemudian memanggil fungsi `generatePrimes(n)` untuk menghasilkan bilangan prima hingga angka yang dimasukkan, dan akhirnya menampilkan daftar bilangan prima tersebut di layar. Program ini memberikan solusi sederhana dan efisien untuk menghitung bilangan prima dalam rentang tertentu.

## B. PRAKTIKUM Menentukan Bilangan Sempurna

### Soal Studi Case

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 faktr dari 6 yaitu 1,2,3 dan  $1+2+3 = 6$ )

### Sourcecode

```
package main
import
"fmt"

// Fungsi untuk menghitung jumlah faktor dari sebuah bilangan (kecuali bilangan
itu sendiri) func sumOfFactors(n int) int {    sum := 0
    for i := 1; i <= n/2; i++
{        if n%i == 0 {
sum += i
        }
    }
    return sum
}

// Fungsi utama untuk memeriksa apakah bilangan merupakan bilangan sempurna
```

```

func isPerfectNumber(n int) bool {
return sumOfFactors(n) == n
} func main() {    var n int
fmt.Print("Masukkan sebuah bilangan: ")
fmt.Scan(&n)
    if isPerfectNumber(n) {        fmt.Println("Ya, bilangan
tersebut adalah bilangan sempurna.")
    } else {        fmt.Println("Tidak, bilangan tersebut bukan
bilangan sempurna.")
    }
}

```

### Screenshoot Output

```

PS C:\Users\ASUS> go run "C:\Users\ASUS\AppData\Local\Temp\tempCodeRunnerFile.go"
Masukkan sebuah bilangan: 5
Tidak, bilangan tersebut bukan bilangan sempurna.
PS C:\Users\ASUS>

```

### Deskripsi Program

Program ini berfungsi untuk memeriksa apakah sebuah bilangan merupakan bilangan sempurna. Program pertama-tama mendefinisikan fungsi `sumOfFactors(n int) int`, yang menghitung jumlah faktor dari suatu bilangan `n`, kecuali bilangan itu sendiri. Fungsi `isPerfectNumber(n int) bool` kemudian memeriksa apakah jumlah faktor-faktor tersebut sama dengan bilangan `n` untuk menentukan apakah bilangan tersebut sempurna. Dalam fungsi `main`, program meminta input bilangan dari pengguna, kemudian memanggil fungsi `isPerfectNumber` untuk memeriksa apakah bilangan tersebut sempurna atau tidak, dan menampilkan hasilnya. Bilangan sempurna adalah bilangan yang jumlah faktor-faktor positifnya (kecuali dirinya sendiri) sama dengan bilangan itu sendiri, seperti 6 (faktor-faktor: 1, 2, 3;  $1+2+3 = 6$ ).