

**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

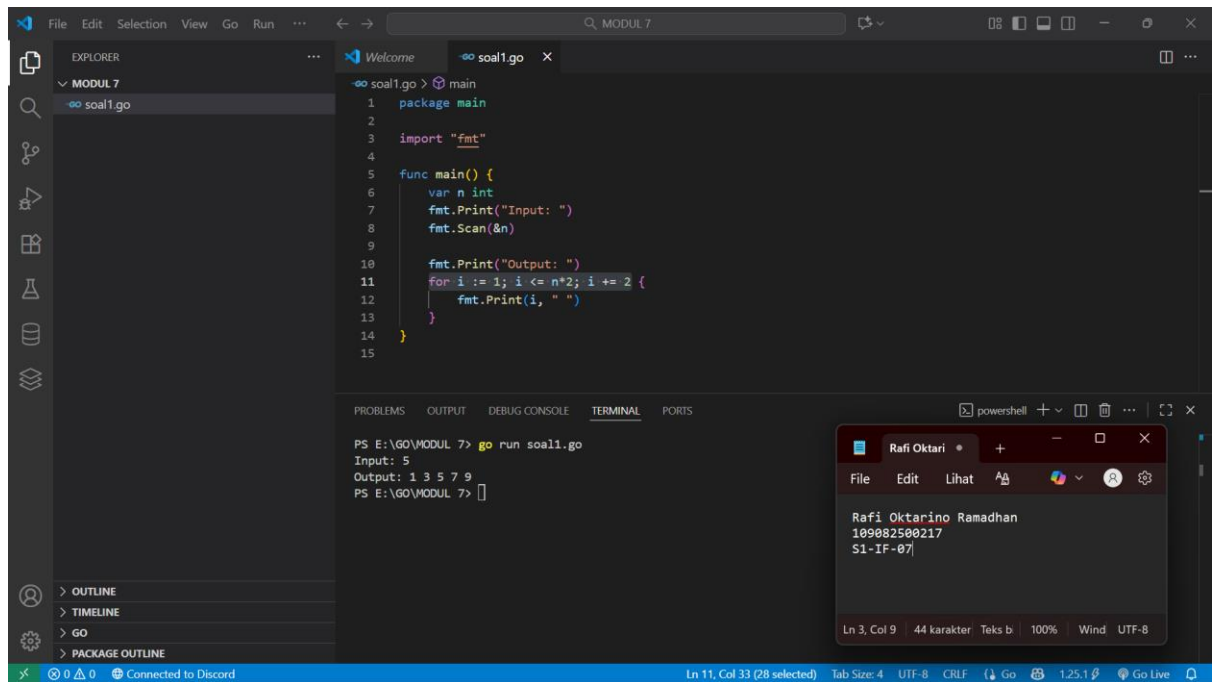
    fmt.Print("Input: ")

    fmt.Scan(&n)

    fmt.Print("Output: ")

    for i := 1; i <= n*2; i += 2 {
        fmt.Print(i, " ")
    }
}
```

#### Screenshoot program



### Deskripsi program

Disini saya di perintahkan untuk membuat program yang berfungsi untuk menampilkan contoh deret bilangan ganjil, tetapi ada inputan untuk menentukan berapa kali mencetak bilangan ganjil yang berurutan dari angka 1. Yang pertama pasti saya menaruh package main, import "fmt" dan func main() terlebih dahulu. Lalu simpan nilai n di var, jangan lupa menggunakan int karna hasil input nya adalah bilangan bulat. Lalu masukan rumus perulangan yang Dimana kita memulai variabel dari angka 1 sesuai dengan perintah, lalu selama i masih kurang dari 1 atau sama dengan  $n*2$ , maka perulangan akan tetap berjalan. Selanjutnya nilai i akan ditambah 2 terus, dikarenakan di perintahkan  $i += 2$ . Yang terakhir fmt.Print (i, " ")

## 2. SOAL 2

### Source Code

```
package main

import "fmt"

func main() {

    var x, y int

    fmt.Print("Masukkan x dan y: ")
```

```

        fmt.Scan(&x, &y)

        hasil := 1

        for i := x; i <= y; i++ {

            hasil *= i

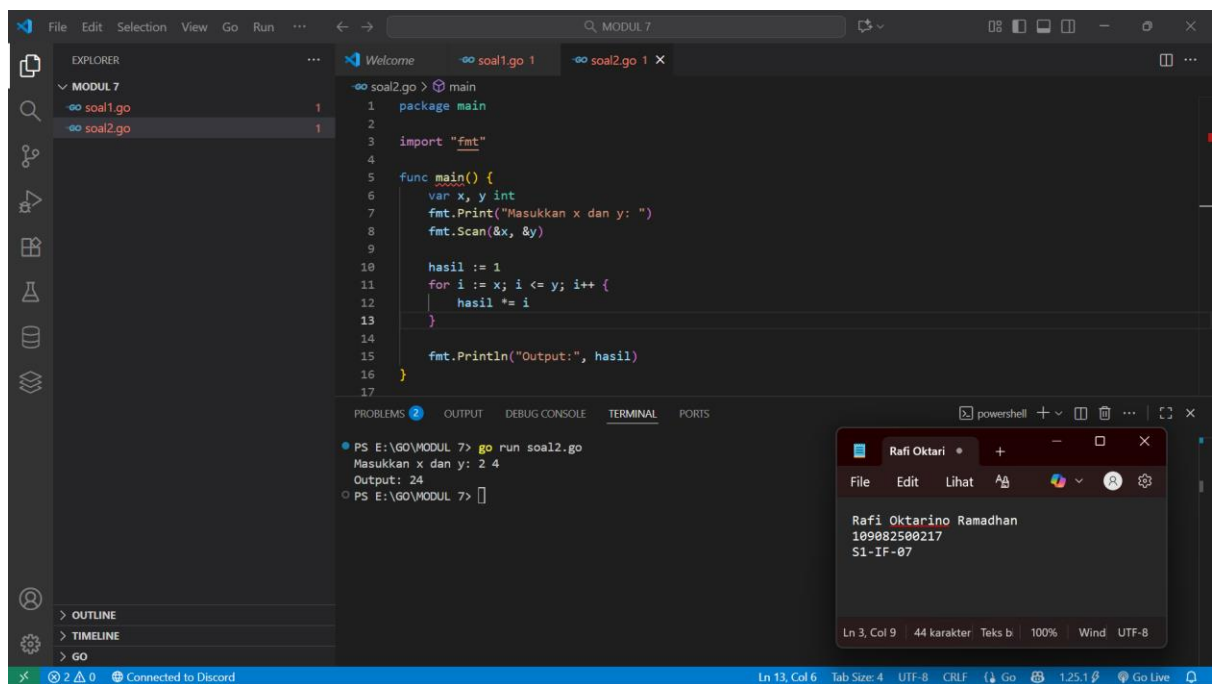
        }

        fmt.Println("Output:", hasil)

    }

```

### Screenshoot program



### Deskripsi program

Disini saya di perintahkan untuk membuat perkalian berurut sesuai dengan perintah inputnya. Yang pertama saya memasukan code package main, import "fmt" dan func main(). Selanjutnya saya mengisi var dengan x, y int supaya nilai inputan x dan y bisa tersimpan dan inputan dari x dan y tersebut bilangan bulat. Lalu scan pada nilai x dan y supaya membaca program awalan nya sampai akhiran nya. Lalu kita memulai perkalian tersebut dari angka 1 setelah itu kita masukan codingan perulangan untuk menghitung perkalian yang berurut. Jangan lupa masukan codingan hasil = hasil \* i agar nilai hasil akan terus dikalikan dengan angka berikutnya.

### 3. SOAL 3

#### Source Code

```
package main

import "fmt"

func main() {
    var keping int
    fmt.Print("Masukkan jumlah keping: ")
    fmt.Scan(&keping)

    kepingPerIkat := 10
    ikatPerKarung := 10
    karungPerPeti := 8

    kepingPerPeti := karungPerPeti * ikatPerKarung *
kepingPerIkat

    peti := keping / kepingPerPeti
    sisa := keping % kepingPerPeti

    kepingPerKarung := ikatPerKarung * kepingPerIkat
    karung := sisa / kepingPerKarung
    sisa = sisa % kepingPerKarung

    ikat := sisa / kepingPerIkat
```

```

        sisaKeping := sisa % kepingPerIkat

        fmt.Printf("%d peti, %d karung, %d ikat, dan %d
keping\n", peti, karung, ikat, sisaKeping)
    }
}

```

## Screenshoot program

```

// MODUL 7
package main
import "fmt"

func main() {
    var keping int
    fmt.Print("Masukkan jumlah keping: ")
    fmt.Scan(&keping)

    kepingPerIkat := 10
    ikatPerKarung := 10
    karungPerPeti := 8

    kepingPerPeti := karungPerPeti * ikatPerKarung * kepingPerIkat

    peti := keping / kepingPerPeti
    sisa := keping % kepingPerPeti

    kepingPerKarung := ikatPerKarung * kepingPerIkat
    karung := sisa / kepingPerKarung
    sisa = sisa % kepingPerKarung

    ikat := sisa / kepingPerIkat

    sisaKeping := sisa % kepingPerIkat

    fmt.Printf("%d peti, %d karung, %d ikat, dan %d keping\n", peti, karung, ikat, sisaKeping)
}

```

```

PS E:\GO\MODUL 7> go run soal3.go
Masukkan jumlah keping: 1000
1 peti, 0 karung, 0 ikat, dan 0 keping
PS E:\GO\MODUL 7>

```

## Deskripsi program

Program ini dibuat untuk membantu seorang pedagang menghitung berapa banyak peti, karung, ikat, dan keping yang ia miliki berdasarkan jumlah keping yang dimasukkan. Awalnya, pengguna diminta memasukkan total keping yang ia punya. Setelah itu, program melakukan perhitungan dengan menggunakan hubungan antar satuan, yaitu 1 peti berisi 8 karung, 1 karung berisi 10 ikat, dan 1 ikat berisi 10 keping. Dari hubungan tersebut, program mengetahui bahwa 1 peti sama dengan 8000 keping. Selanjutnya, program menghitung berapa banyak peti yang bisa dibuat dari jumlah keping yang ada, lalu menghitung sisa keping untuk diubah menjadi karung, kemudian ikat, dan terakhir keping yang tersisa. Hasil akhirnya ditampilkan dalam bentuk kalimat yang menjelaskan jumlah peti, karung, ikat, dan keping yang setara dengan total keping yang dimasukkan pengguna.