

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 5 & 6  
FOR-LOOP**



**Disusun oleh:**

**REZKY FARREL**

**109082500203**

**S1IF-13--02**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## LATIHAN KELAS – GUIDED

### 1. Guided 1

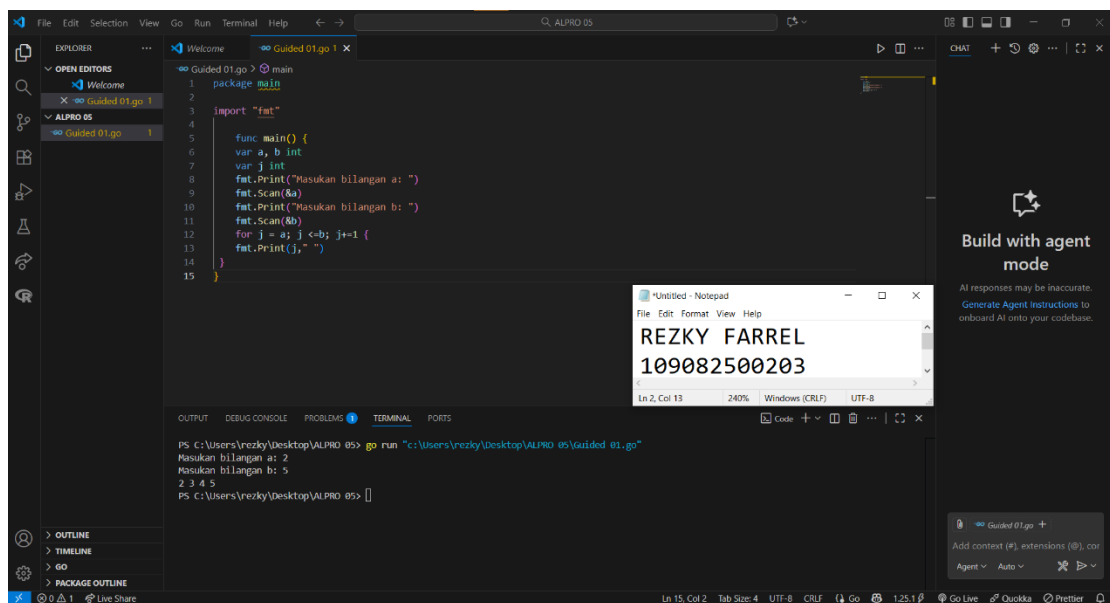
#### Source Code

```
package main

import "fmt"

func main() {
    var detik, jam, menit int
    fmt.Scan(&detik)
    jam = detik / 3600
    menit = (detik % 3600) / 60
    detik = detik % 60
    fmt.Println(jam, "jam", menit, "menit", detik, "detik")
}
```

#### Screenshoot program :



#### Deskripsi program :

Program ini berfungsi untuk mencetak deret bilangan berurutan berdasarkan input dari pengguna. Awalnya, program mendeklarasikan variabel integer a, b, dan j. Menggunakan fungsi dari paket fmt, program terlebih dahulu meminta pengguna memasukkan "bilangan a" (nilai awal) dan "bilangan b" (nilai akhir), yang kemudian disimpan ke dalam variabel masing-masing. Inti dari program ini adalah perulangan for yang dimulai dengan memberikan nilai a kepada j. Perulangan akan terus berjalan selama nilai j masih lebih kecil atau sama dengan b.

## 2. Guided 2

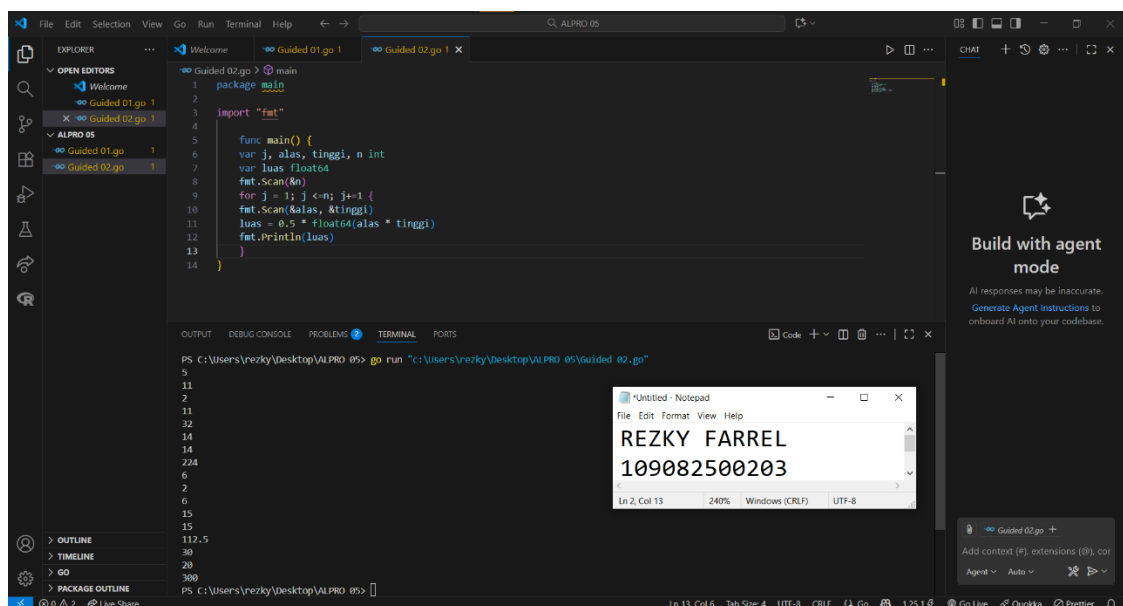
### Source Code

```
package main

import "fmt"

func main() {
    var j, alas, tinggi, n int
    var luas float64
    fmt.Scan(&n)
    for j = 1; j <=n; j+=1 {
        fmt.Scan(&alas, &tinggi)
        luas = 0.5 * float64(alas * tinggi)
        fmt.Println(luas)
    }
}
```

### Screenshoot program :



### Deskripsi program :

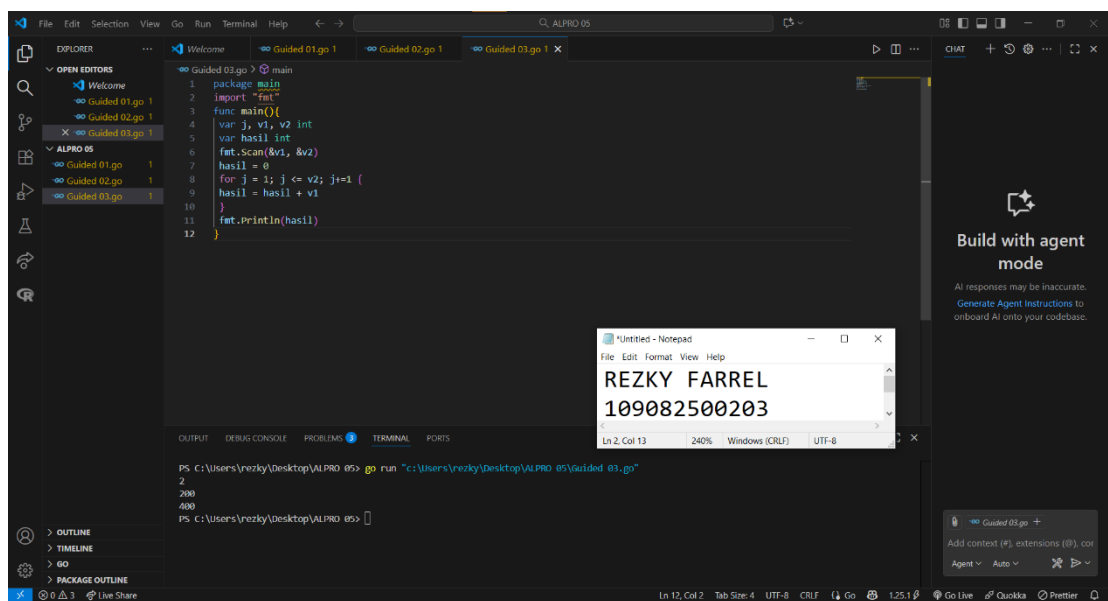
Program ini menjelaskan sebagai kalkulator untuk menghitung luas beberapa segitiga secara berurutan dalam satu kali eksekusi. Saat dijalankan, program pertama-tama akan meminta pengguna memasukkan sebuah angka (n) yang menentukan berapa kali proses perhitungan akan diulang. Setelah itu, program akan memasuki sebuah perulangan for yang berjalan sebanyak n kali. Di dalam setiap iterasi, program akan meminta pengguna untuk memasukkan dua nilai, yaitu alas dan tinggi segitiga. Menggunakan rumus matematika  $Luas = \frac{1}{2} \times \text{alas} \times \text{tinggi}$ . Hasil luas dari setiap perhitungan tersebut kemudian langsung dicetak ke baris baru pada terminal sebelum program melanjutkan ke iterasi berikutnya atau berhenti jika sudah mencapai jumlah yang ditentukan.

### 3. Guided 3

#### Source Code

```
package main
import "fmt"
func main(){
    var j, v1, v2 int
    var hasil int
    fmt.Scan(&v1, &v2)
    hasil = 0
    for j = 1; j <= v2; j+=1 {
        hasil = hasil + v1
    }
    fmt.Println(hasil)
}
```

#### Screenshoot program :



#### Deskripsi program :

Program ini merupakan asarnya mensimulasikan operasi perkalian dengan menggunakan metode penjumlahan berulang. Awalnya, program meminta pengguna untuk memasukkan dua bilangan bulat, yang disimpan dalam variabel v1 dan v2. Sebuah variabel hasil kemudian diinisialisasi dengan nilai nol. Inti dari program ini adalah sebuah perulangan for yang akan berjalan sebanyak v2 kali. Di dalam setiap putaran, nilai dari v1 akan ditambahkan secara akumulatif ke dalam variabel hasil. Setelah perulangan selesai, program akan mencetak nilai akhir dari hasil, yang secara efektif merupakan hasil perkalian antara v1 dan v2.

## TUGAS

### 1. Tugas 1

#### Source code

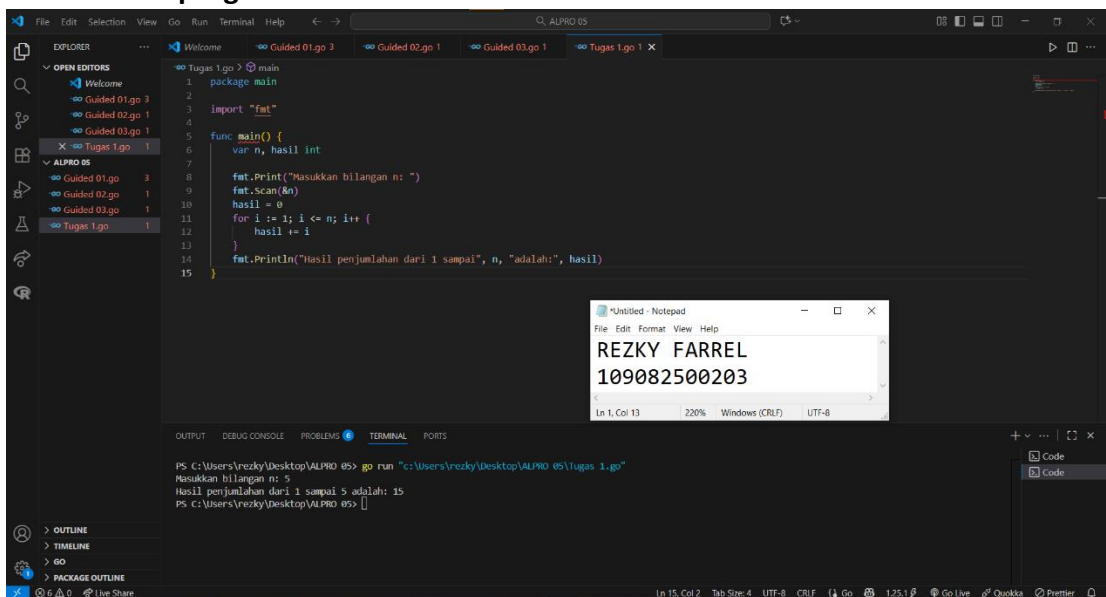
```
package main

import "fmt"

func main() {
    var n, hasil int

    fmt.Print("Masukkan bilangan n: ")
    fmt.Scan(&n)
    hasil = 0
    for i := 1; i <= n; i++ {
        hasil += i
    }
    fmt.Println("Hasil penjumlahan dari 1 sampai", n,
"adalah:", hasil)
}
```

#### Screenshoot program :



#### Deskripsi program :

Program ini merupakan perhitungan jumlah total dari deret bilangan bulat yang dimulai dari 1 hingga angka  $n$  yang diinputkan oleh pengguna. Setelah meminta dan menerima masukan  $n$ , program menginisialisasi sebuah variabel `hasil` dengan nilai nol yang berfungsi sebagai akumulator.

## Tugas 2

### Source code

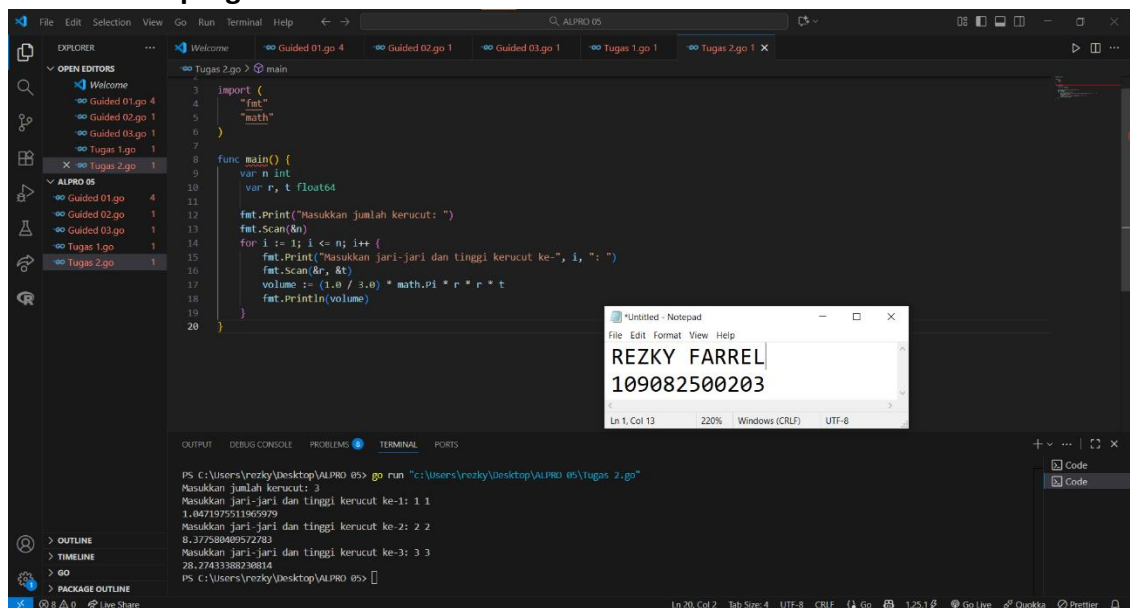
```
package main

import (
    "fmt"
    "math"
)

func main() {
    var n int
    var r, t float64

    fmt.Print("Masukkan jumlah kerucut: ")
    fmt.Scan(&n)
    for i := 1; i <= n; i++ {
        fmt.Print("Masukkan jari-jari dan tinggi kerucut ke-", i, ": ")
        fmt.Scan(&r, &t)
        volume := (1.0 / 3.0) * math.Pi * r * r * t
        fmt.Println(volume)
    }
}
```

### Screenshoot program



### Deskripsi program

Program ini berfungsi sebagai kalkulator untuk menghitung volume beberapa kerucut secara berurutan. Saat dieksekusi, program terlebih dahulu meminta pengguna untuk memasukkan jumlah total kerucut yang ingin dihitung volumenya. Berdasarkan jumlah tersebut, program akan menjalankan sebuah perulangan for. Di dalam setiap iterasi, pengguna akan diminta untuk memasukkan nilai jari-jari (r) dan tinggi (t) untuk kerucut yang bersangkutan. Hasil perhitungan volume dari setiap kerucut akan langsung dicetak ke layar sebelum program melanjutkan ke iterasi berikutnya hingga semua perhitungan selesai

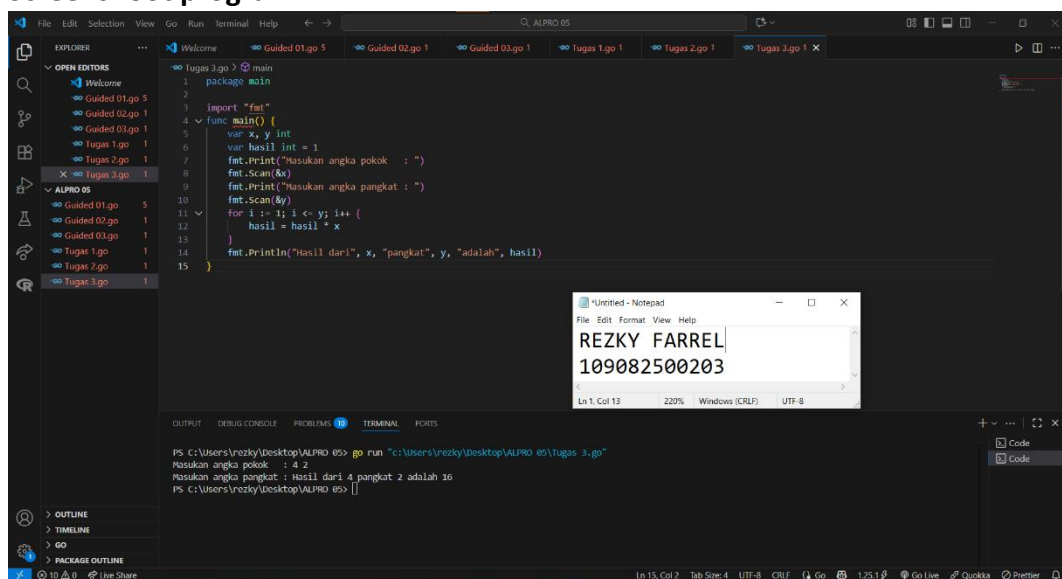
### Tugas 3

#### Source code

```
package main

import "fmt"
func main() {
    var x, y int
    var hasil int = 1
    fmt.Print("Masukan angka pokok : ")
    fmt.Scan(&x)
    fmt.Print("Masukan angka pangkat : ")
    fmt.Scan(&y)
    for i := 1; i <= y; i++ {
        hasil = hasil * x
    }
    fmt.Println("Hasil dari", x, "pangkat", y, "adalah",
hasil)
}
```

#### Screenshoot program :



#### Deskripsi program

Program merupakan kalkulator perpangkatan yang bekerja dengan metode perkalian berulang. Saat dijalankan, program akan meminta pengguna untuk memasukkan dua nilai: "angka pokok" (basis, disimpan sebagai x) dan "angka pangkat" (eksponen, disimpan sebagai y). Program menginisialisasi sebuah variabel hasil dengan nilai 1, yang akan bertindak sebagai akumulator.

## Tugas 4

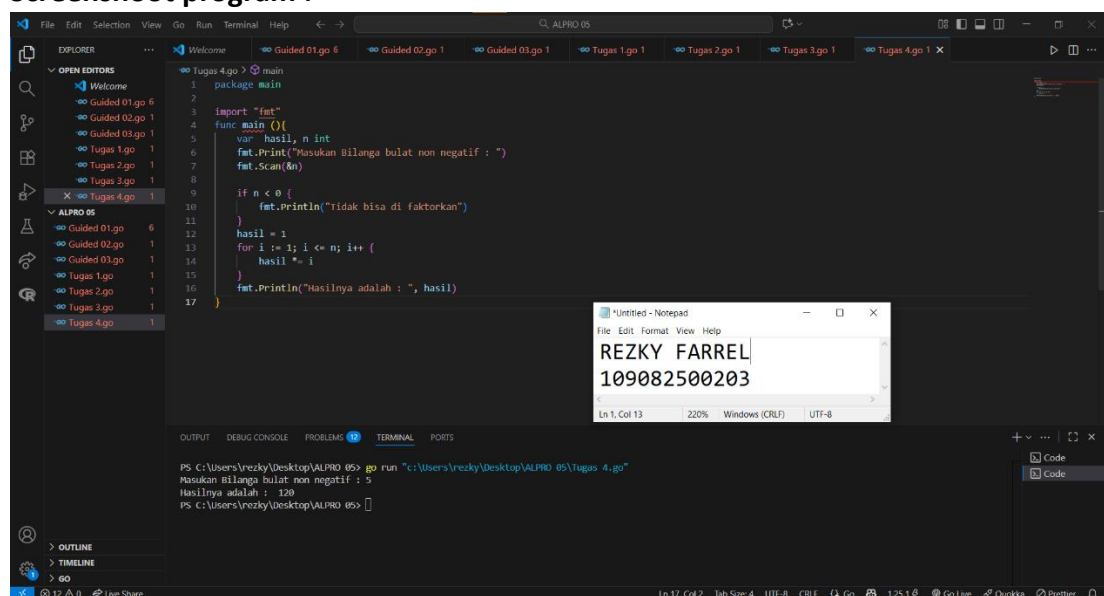
### Source code

```
package main

import "fmt"
func main () {
    var hasil, n int
    fmt.Print("Masukan Bilangan bulat non negatif : ")
    fmt.Scan(&n)

    if n < 0 {
        fmt.Println("Tidak bisa di faktorkan")
    }
    hasil = 1
    for i := 1; i <= n; i++ {
        hasil *= i
    }
    fmt.Println("Hasilnya adalah : ", hasil)
}
```

### Screenshoot program :



### Deskripsi program



Program merupakan perhitungan nilai faktorial dari sebuah bilangan bulat non-negatif yang dimasukkan oleh pengguna. Setelah menerima input, program pertama-tama melakukan validasi untuk memastikan angka tersebut tidak negatif; jika negatif, program akan menampilkan pesan kesalahan. Apabila input valid, program menginisialisasi sebuah variabel hasil dengan nilai 1. Kemudian, sebuah perulangan for dieksekusi, dimulai dari 1 hingga angka yang diinputkan. Di dalam setiap iterasi, nilai hasil akan dikalikan secara akumulatif dengan angka iterasi saat ini. Setelah perulangan selesai, program akan mencetak nilai akhir dari hasil, yang merupakan jawaban faktorial dari bilangan tersebut.