

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 4**

**I/O Tipe Data & Variable**



**Disusun oleh:**

**Ahmad Malik Arrayyan**

**109082500116**

**S1IF-13-02**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## LATIHAN KELAS – GUIDED

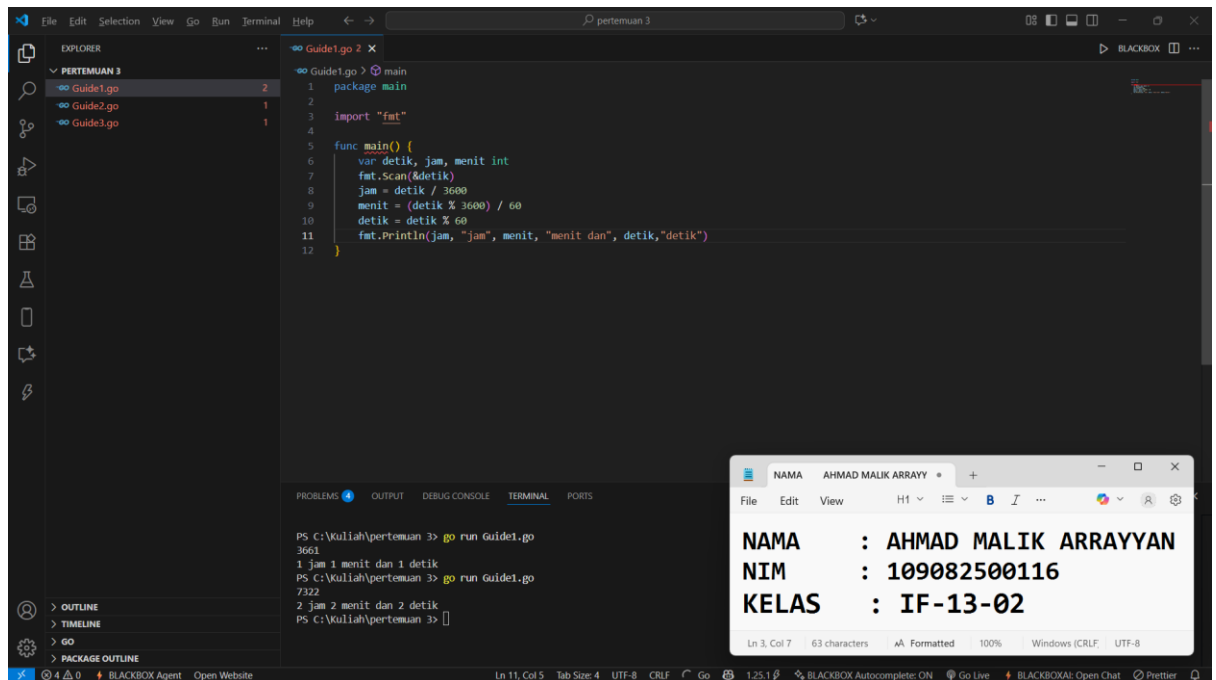
### 1. Guided 1 Source Code

```
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 dan",
detik,"detik")
}
```

### Screenshoot program



### Deskripsi program

Program ini berfungsi untuk mengonversi total detik menjadi jam, menit, dan detik dengan menggunakan operasi pembagian dan modulus. Struktur logika-nya sangat umum digunakan untuk masalah konversi waktu di banyak bahasa pemrograman.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {

    var bilangan, d1, d2, d3 int

    fmt.Scan(&bilangan)

    d1 = bilangan / 100

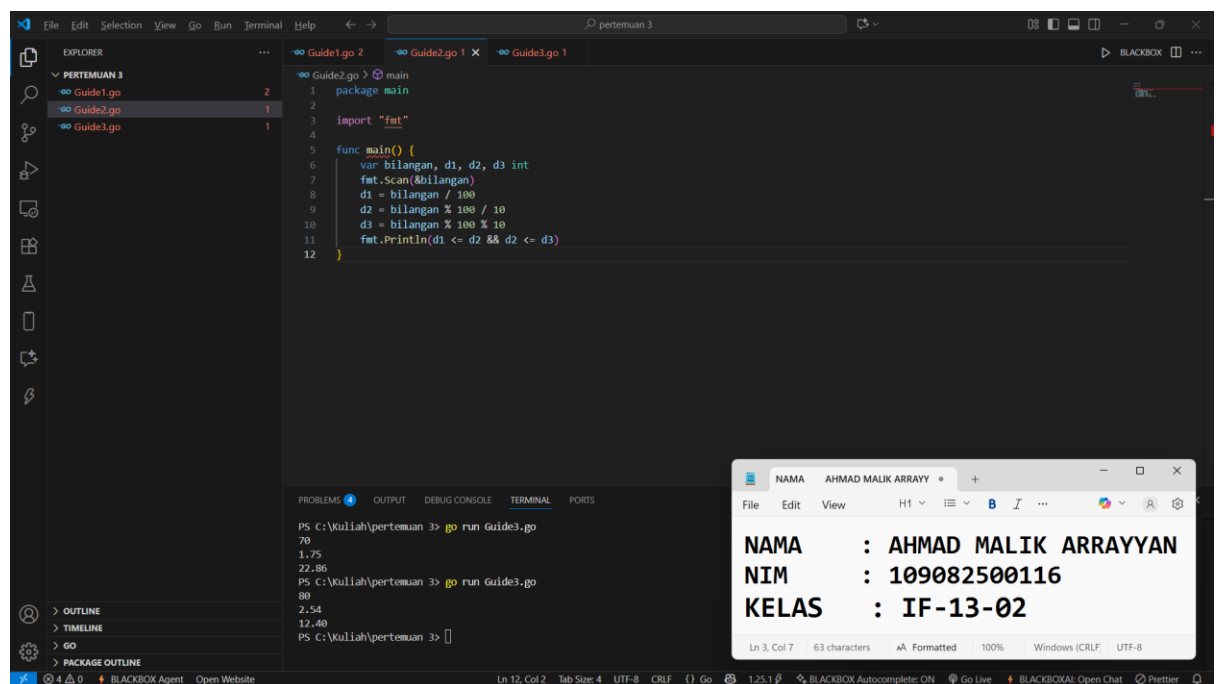
    d2 = bilangan % 100 / 10

    d3 = bilangan % 100 % 10

    fmt.Println(d1 <= d2 && d2 <= d3)

}
```

### Screenshoot program



### Deskripsi program

Program ini berfungsi untuk:

- Membaca bilangan tiga digit dari input.
- Memecah bilangan tersebut menjadi tiga digit terpisah (ratusan, puluhan, satuan).
- Mengecek apakah ketiga digit tersebut berurutan menaik atau tetap (non-decreasing).
- Menampilkan hasil berupa true atau false.

### 3. Guided 3

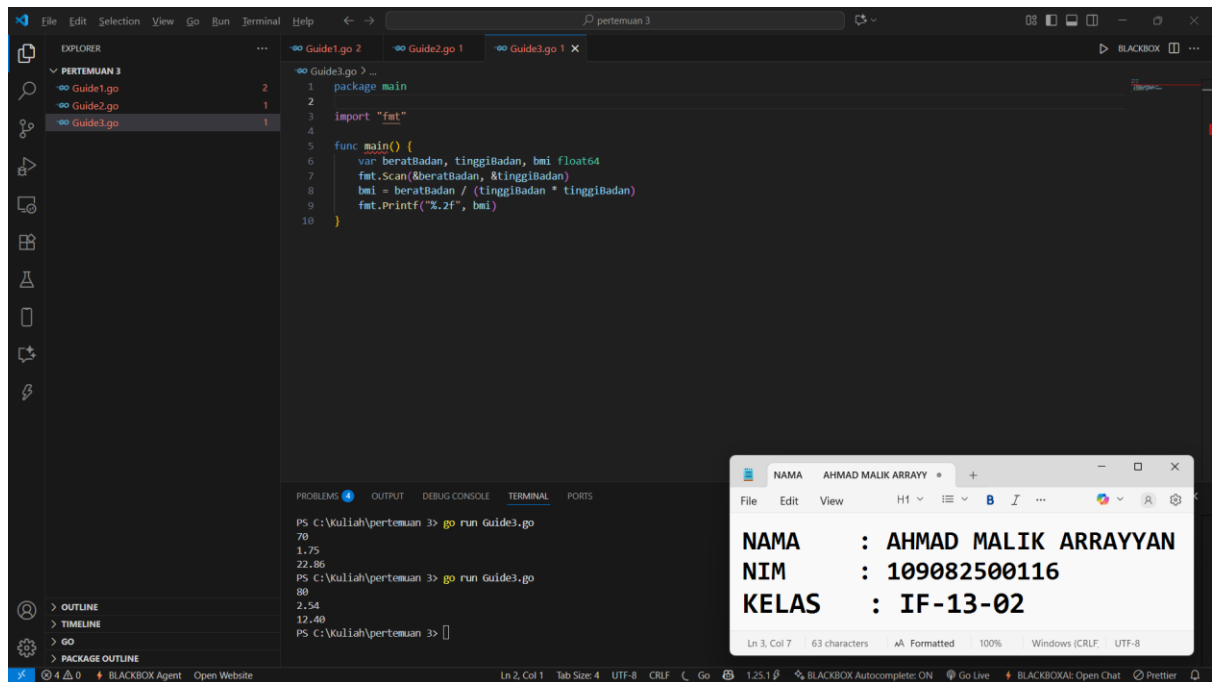
#### Source Code

```
package main

import "fmt"

func main() {
    var beratBadan, tinggiBadan, bmi float64
    fmt.Scan(&beratBadan, &tinggiBadan)
    bmi = beratBadan / (tinggiBadan * tinggiBadan)
    fmt.Printf("%.2f", bmi)
}
```

#### Screenshoot program



## Deskripsi program

Program ini digunakan untuk menghitung **BMI (Body Mass Index)** dengan langkah:

1. Membaca berat badan dan tinggi badan dari input.
2. Menghitung nilai BMI menggunakan rumus standar.
3. Menampilkan hasilnya dalam dua digit desimal.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import (
    "fmt"
)

func main() {
    var totalAwal int
    var diskon int
```

```

// Input

fmt.Scanln(&totalAwal)

fmt.Scanln(&diskon)


// Hitung total setelah diskon

totalAkhir := totalAwal - (totalAwal * diskon / 100)


// Output

fmt.Println(totalAkhir)

}

```

## Screenshoot program

The screenshot shows a Go IDE with the following components:

- EXPLORER:** Shows a project named 'TUGAS 1' with a file 'Tugas1.go'.
- EDITOR:** Displays the source code of 'Tugas1.go', which is identical to the code block provided above.
- TERMINAL:** Shows the execution of the program. The user runs 'go run Tugas1.go' and enters the following input:
 

```

19000
10
15300

```

 The output of the program is:
 

```

20000

```
- OUTPUT WINDOW:** Displays the program's output, which is a formatted string:
 

```

NAMA      : AHMAD MALIK ARRAYYAN
NIM       : 109082500116
KELAS    : IF-13-02

```

## Deskripsi program

Program ini sangat sederhana

fungsi utamanya adalah menghitung harga setelah diskon,

dengan cara mengurangi harga awal dengan potongan harga sesuai persentase diskon yang kamu masukkan.

## 2. Tugas 2

### Source code

```
package main

import (
    "fmt"
)

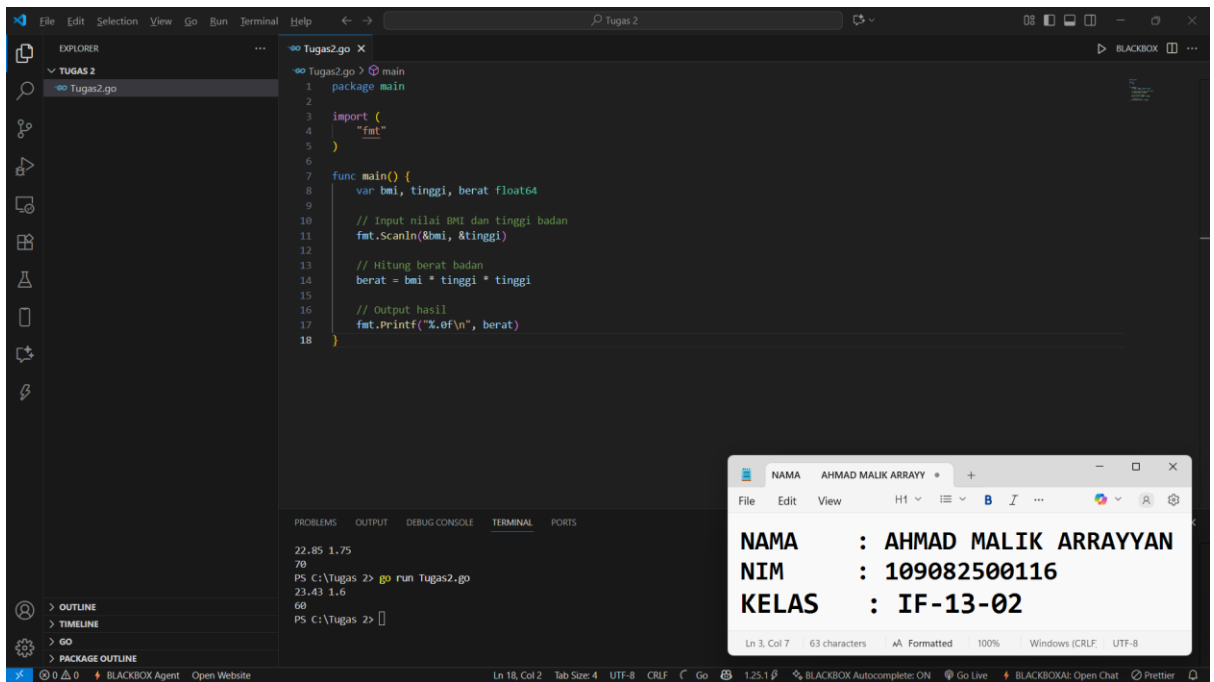
func main() {
    var bmi, tinggi, berat float64

    // Input nilai BMI dan tinggi badan
    fmt.Scanln(&bmi, &tinggi)

    // Hitung berat badan
    berat = bmi * tinggi * tinggi

    // Output hasil
    fmt.Printf("%.0f\n", berat)
}
```

### Screenshoot program



### Deskripsi program

Program ini:

- Meminta kamu memasukkan BMI dan tinggi badan.
- Menghitung berat badan berdasarkan rumus BMI.
- Menampilkan hasilnya dengan dua angka di belakang koma.

Sederhananya, program ini adalah kebalikan dari program BMI sebelumnya — kalau yang sebelumnya mencari BMI dari berat dan tinggi, yang ini mencari berat dari BMI dan tinggi.

### 3. Tugas 3

#### Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var ax, ay, bx, by, cx, cy float64

    // Input koordinat titik A, B, dan C
    fmt.Scanln(&ax, &ay)
    fmt.Scanln(&bx, &by)
    fmt.Scanln(&cx, &cy)

    // Hitung panjang sisi-sisi segitiga
    ab := math.Sqrt(math.Pow(bx-ax, 2) + math.Pow(by-ay, 2))
```



```

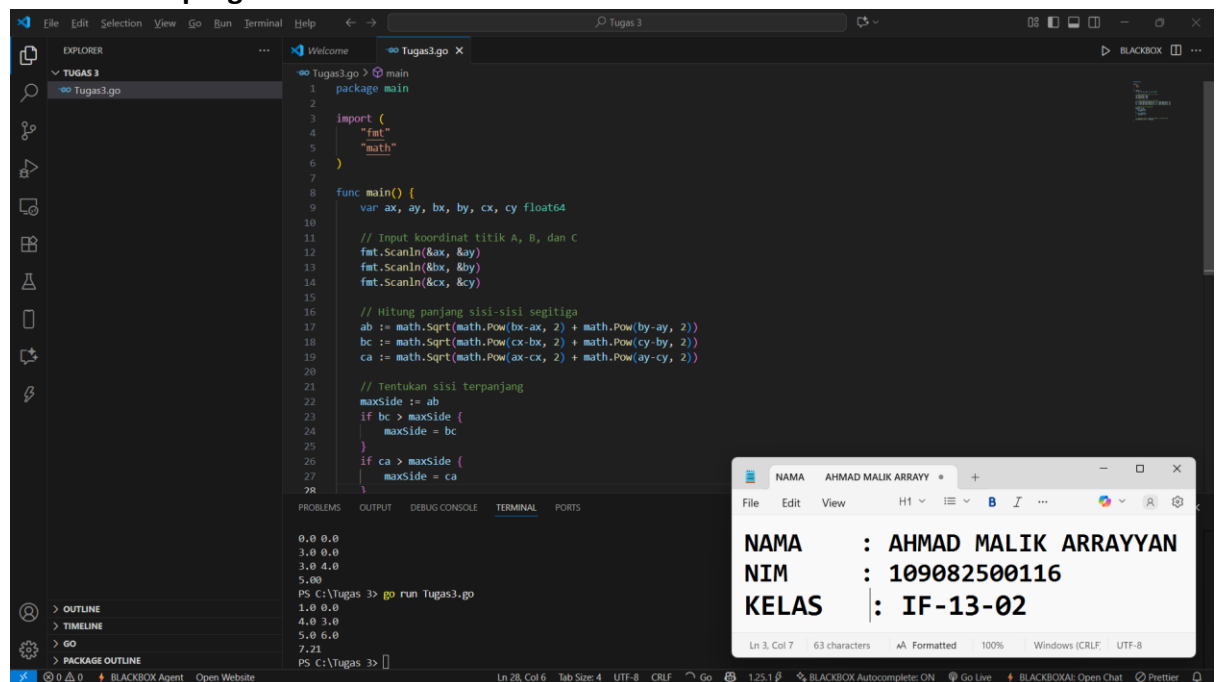
bc := math.Sqrt(math.Pow(cx-bx, 2) + math.Pow(cy-by, 2))
ca := math.Sqrt(math.Pow(ax-cx, 2) + math.Pow(ay-cy, 2))

// Tentukan sisi terpanjang
maxSide := ab
if bc > maxSide {
    maxSide = bc
}
if ca > maxSide {
    maxSide = ca
}

// Tampilkan hasil dengan dua angka di belakang koma
fmt.Printf("%.2f\n", maxSide)
}

```

### Screenshoot program



### Deskripsi program

Program ini membaca koordinat tiga titik di bidang (titik A, B, dan C), menghitung panjang ketiga sisi segitiga yang dibentuk oleh ketiga titik tersebut, lalu menampilkan panjang sisi terpanjang (nilai maksimal di antara ketiga sisi).

- float64 dipakai agar hasil jarak (yang bisa desimal) dapat disimpan.
- Penggunaan `math.Pow(..., 2)` bekerja, tapi sering lebih cepat dan lebih ringkas menulis `(dx*dx)` daripada `math.Pow(dx, 2)`.
- Kalau ingin, bisa juga menambahkan pengecekan apakah tiga titik itu membentuk segitiga valid (tidak kolinear) sebelum menghitung sisi.