

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

**MODUL 04  
I/O, TIPE DATA & VARIABEL**



**Disusun oleh:**  
**FARID HERDIYANTO VITASANDI**  
**109082500123**  
**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 int

    fmt.Print("Masukkan detik: ")

    fmt.Scan(&detik)

    konversiJam := detik / 3600

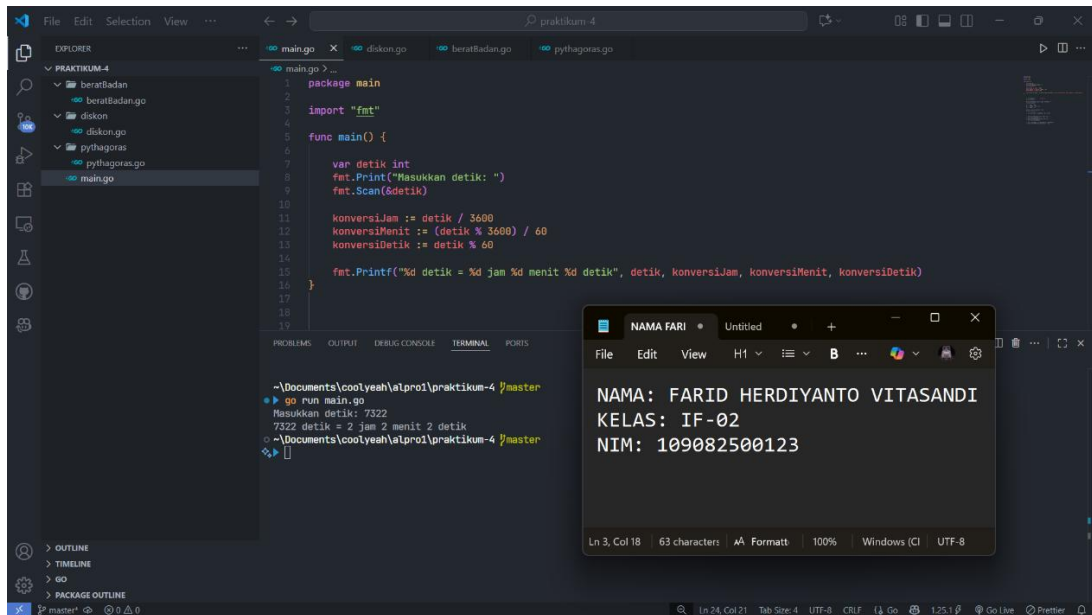
    konversiMenit := (detik % 3600) / 60

    konversiDetik := detik % 60

    fmt.Printf("%d detik = %d jam %d menit %d detik", detik,
konversiJam, konversiMenit, konversiDetik)

}
```

## Screenshoot program



The screenshot shows a Go program in a VS Code editor. The program is located in `main.go` within a project named `praktikum-4`. The code defines a `main` function that prompts the user for an integer value representing seconds. It then calculates the equivalent time in hours, minutes, and seconds using integer division and modulo operations. The results are printed using `fmt.Printf` in a formatted string.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var detik int
8     fmt.Print("Masukkan detik: ")
9     fmt.Scan(&detik)
10
11     konversiJam := detik / 3600
12     konversiMenit := (detik % 3600) / 60
13     konversiDetik := detik % 60
14
15     fmt.Printf("%d detik = %d jam %d menit %d detik", detik, konversiJam, konversiMenit, konversiDetik)
16 }
17
18
19
```

The terminal output shows the program execution with the input `7322` and the resulting output: `7322 detik = 2 jam 2 menit 2 detik`.

A separate window titled `NAMA FARI` displays the following information:

```
NAMA: FARID HERDIYANTO VITASANDI
KELAS: IF-02
NIM: 109082500123
```

## Deskripsi program

Program di atas adalah program yang dibuat untuk mengonversi satuan waktu dari detik menjadi jam, menit, dan detik. Program akan meminta input berupa bilangan bulat dari User yang kemudian akan disimpan ke dalam variable *detik*. Setelah itu akan dilakukan perhitungan yang dimana jumlah jam diperoleh dengan  $\text{detik} / 3600$ , menit dihitung dari sisa detik dengan  $(\text{detik} \% 3600) / 60$ , dan detik sisanya diperoleh dari  $\text{detik} \% 60$ . Hasil konversi ini nantinya akan ditampilkan menggunakan `fmt.Printf` dalam format jam, menit, dan detik seperti yang ada pada hasil screenshoot di atas.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {

    var angka int

    var status bool

    fmt.Print("Masukkan deret angka (100-999): ")

    fmt.Scan(&angka)

    a1 := angka / 100

    a2 := (angka % 100) / 10

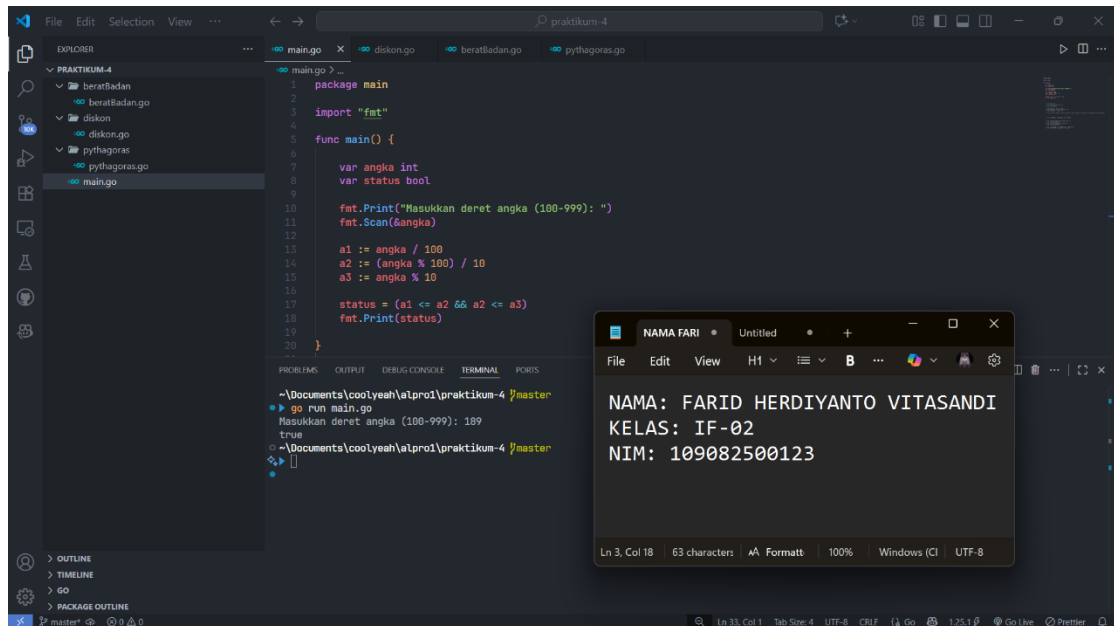
    a3 := angka % 10

    status = (a1 <= a2 && a2 <= a3)

    fmt.Print(status)

}
```

## Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var angka int
7     var status bool
8
9
10    fmt.Print("Masukkan deret angka (100-999): ")
11    fmt.Scan(&angka)
12
13    a1 := angka / 100
14    a2 := (angka % 100) / 10
15    a3 := angka % 10
16
17    status = (a1 <= a2 && a2 <= a3)
18    fmt.Print(status)
19 }
20
```

```
~\Documents\coolyeah\alpro1\praktikum-4 /master
go run main.go
Masukkan deret angka (100-999): 189
true
~\Documents\coolyeah\alpro1\praktikum-4 /master
```

NAMA: FARID HERDIYANTO VITASANDI  
KELAS: IF-02  
NIM: 109082500123

## Deskripsi program

Program di atas adalah program yang dibuat untuk mengecek apakah tiga digit sebuah angka berurutan dari kecil ke besar atau tidak. User diminta untuk menginput angka bilangan bulat antara 100 sampai 999 yang nantinya akan disimpan di variable *angka*. Angka ratusan diperoleh dengan  $angka / 100$ , puluhan dengan  $(angka \% 100) / 10$ , dan satuan dengan  $angka \% 10$ . Setelah itu dilakukan pengecekan dengan kondisi  $a1 \leq a2 \ \&\& \ a2 \leq a3$ , artinya digit pertama harus lebih kecil atau sama dengan digit kedua, dan digit kedua lebih kecil atau sama dengan digit ketiga. Hasilnya akan disimpan di variable *status* yang bertipe boolean dan akan ditampilkan sebagai *true* atau *false* seperti yang terlihat pada hasil screenshoot di atas.

### 3. Guided 3

#### Source Code

```
package main

import "fmt"

func main() {

    var beratBadan, tinggiBadan, bmi float64

    fmt.Print("Masukkan berat badan (kg): ")
    fmt.Scan(&beratBadan)

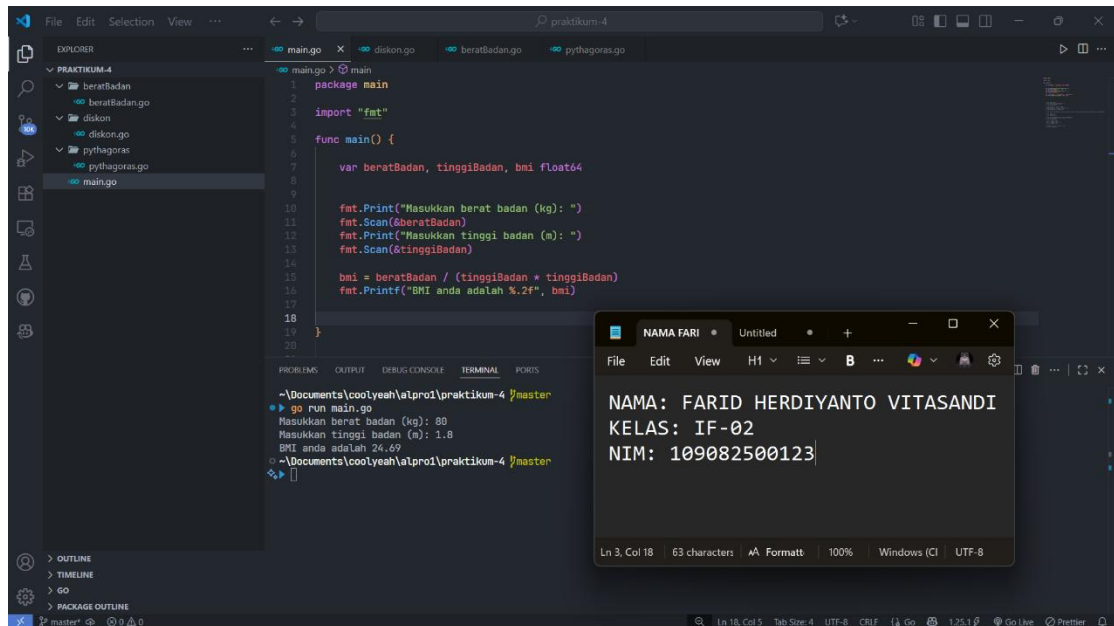
    fmt.Print("Masukkan tinggi badan (m): ")
    fmt.Scan(&tinggiBadan)

    bmi = beratBadan / (tinggiBadan * tinggiBadan)

    fmt.Printf("BMI anda adalah %.2f", bmi)

}
```

## Screenshoot program



The screenshot shows a Go program in a VS Code editor. The program is named `main.go` and is located in a directory named `praktikum-4`. The code defines three variables: `beratBadan`, `tinggiBadan`, and `bmi`, all of type `float64`. It prompts the user to enter their weight and height, calculates the BMI using the formula  $\text{BMI} = \frac{\text{beratBadan}}{(\text{tinggiBadan} * \text{tinggiBadan})}$ , and prints the result. The output shows the user's name as FARID HERDIYANTO VITASANDI, their class as IF-02, and their NIM as 109082500123.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var beratBadan, tinggiBadan, bmi float64
8
9
10    fmt.Print("Masukkan berat badan (kg): ")
11    fmt.Scan(&beratBadan)
12    fmt.Print("Masukkan tinggi badan (m): ")
13    fmt.Scan(&tinggiBadan)
14
15    bmi = beratBadan / (tinggiBadan * tinggiBadan)
16    fmt.Printf("BMI anda adalah %.2f", bmi)
17
18 }
19
20
```

Terminal output:

```
~\Documents\cool\yeah\alpro1\praktikum-4 $ go run main.go
Masukkan berat badan (kg): 80
Masukkan tinggi badan (m): 1.8
BMI anda adalah 24.69
```

## Deskripsi program

Program di atas adalah program yang dibuat untuk menghitung BMI (Body Mass Index) seseorang. Program ini menggunakan tiga variabel yang bertipe *float64* yaitu, *beratBadan*, *tinggiBadan*, dan *bmi*. Program akan meminta User untuk memasukkan berat badan dalam satuan kilogram dan tinggi badan dalam satuan meter. Setelah itu program akan mulai menghitung dengan rumus yang sudah ditetapkan yaitu  $\text{berat} / (\text{tinggiBadan} * \text{tinggiBadan})$ . Hasil perhitungan nantinya akan disimpan di variabel *bmi* dan ditampilkan dengan format dua angka di belakang koma menggunakan *fmt.Printf* seperti yang terlihat pada screenshoot di atas.

## TUGAS

### 1. Soal 1

#### Source Code

```
package main

import "fmt"

func main() {

    var hargaAwal, diskon, hargaAkhir int

    fmt.Print("Masukkan harga belanja awal: Rp")

    fmt.Scan(&hargaAwal)

    fmt.Print("Masukkan diskon (%): ")

    fmt.Scan(&diskon)

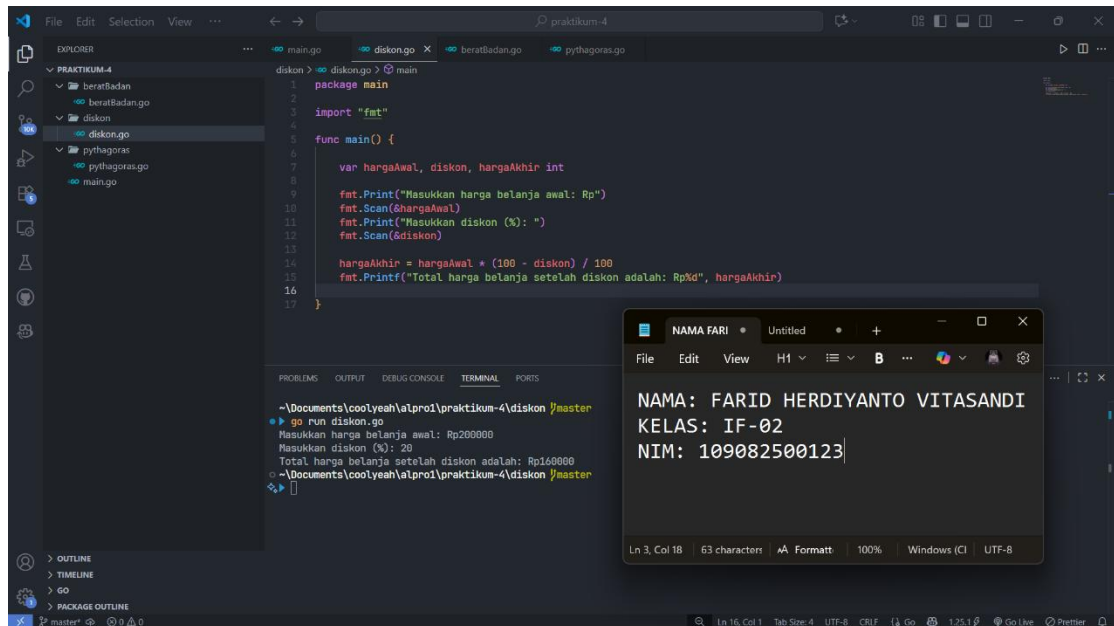
    hargaAkhir = hargaAwal * (100 - diskon) / 100

    fmt.Printf("Total harga belanja setelah diskon adalah: Rp%d", hargaAkhir)

}
```



## Screenshoot program



The screenshot shows a Go program in a VS Code editor. The file explorer on the left shows a project named 'PRAKTIKUM-4' with subdirectories 'beratBadan', 'diskon', and 'pythagoras'. The 'diskon.go' file is open in the editor. The code defines a `main` package and a `main` function that takes `hargaAwal`, `diskon`, and `hargaAkhir` as integers. It prompts the user for the initial price and discount percentage, calculates the final price using the formula  $hargaAkhir = hargaAwal * (100 - diskon) / 100$ , and prints the result.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var hargaAwal, diskon, hargaAkhir int
8
9     fmt.Print("Masukkan harga belanja awal: Rp")
10    fmt.Scan(&hargaAwal)
11    fmt.Print("Masukkan diskon (%): ")
12    fmt.Scan(&diskon)
13
14    hargaAkhir = hargaAwal * (100 - diskon) / 100
15    fmt.Printf("Total harga belanja setelah diskon adalah: Rp%d", hargaAkhir)
16
17 }
```

The terminal output shows the program being run from the `diskon` directory. The user enters `200000` for the initial price and `20` for the discount. The program outputs the total price after the discount: `Rp160000`.

```
~\Documents\coolYeah\alpro1\praktikum-4\diskon $go run diskon.go
Masukkan harga belanja awal: Rp200000
Masukkan diskon (%): 20
Total harga belanja setelah diskon adalah: Rp160000
~\Documents\coolYeah\alpro1\praktikum-4\diskon $
```

## Deskripsi program

Program di atas adalah program yang dibuat untuk menghitung harga sesuatu setelah diskon. Program akan meminta User untuk memasukkan harga awal dan persentase diskon, lalu program akan mulai menghitung harga akhir dengan rumus  $hargaAkhir = hargaAwal * (100 - diskon) / 100$ . Kemudian, hasilnya akan ditampilkan menggunakan `fmt.Printf` seperti yang terlihat pada screenshoot di atas.

## 2. Soal 2

### Source Code

```
package main

import (
    "fmt"
    "math"
)

func main() {

    var bmi, tinggiBadan, beratBadan float64

    fmt.Print("Masukkan nilai BMI: ")
    fmt.Scan(&bmi)

    fmt.Print("Masukkan tinggi badan (m): ")
    fmt.Scan(&tinggiBadan)

    beratBadan = bmi * (tinggiBadan * tinggiBadan)

    fmt.Print("Berat badan anda adalah: ",
math.Ceil(beratBadan), "Kg")
}
```

## Screenshoot program

The screenshot shows a Go IDE with a project named 'praktikum-4'. The file explorer on the left shows a directory structure with files like 'beratBadan.go', 'diskon.go', 'pythagoras.go', and 'main.go'. The main editor displays the code for 'beratBadan.go'.

```
1 package main
2 import (
3     "fmt"
4     "math"
5 )
6
7 func main() {
8     var bmi, tinggiBadan, beratBadan float64
9
10    fmt.Print("Masukkan nilai BMI: ")
11    fmt.Scan(&bmi)
12    fmt.Print("Masukkan tinggi badan (m): ")
13    fmt.Scan(&tinggiBadan)
14
15    beratBadan = bmi * (tinggiBadan * tinggiBadan)
16    fmt.Print("Berat badan anda adalah: ", math.Ceil(beratBadan), "Kg")
17 }
18
19
```

Below the code editor, the terminal window shows the execution of the program:

```
~\Documents\coolYeah\alpro1\praktikum-4\beratBadan /master
go run beratBadan.go
Masukkan nilai BMI: 24.69
Masukkan tinggi badan (m): 1.8
Berat badan anda adalah: 89Kg
~\Documents\coolYeah\alpro1\praktikum-4\beratBadan /master
```

Overlaid on the bottom right is a text editor window titled 'NAMA FARI' containing the following text:

```
NAMA: FARID HERDIYANTO VITASANDI
KELAS: IF-02
NIM: 109082500123
```

## Deskripsi program

Program di atas berfungsi untuk menghitung berat badan seseorang berdasarkan BMI (Body Mass Index) dan tinggi badannya. Pada kode di atas terdapat tiga variabel bertipe float64 yaitu: *bmi*, *tinggiBadan*, dan *beratBadan* yang berfungsi untuk menyimpan nilai. Kemudian, program akan meminta User untuk memasukkan nilai BMI dan tinggi badan dalam meter yang kemudian akan dihitung dengan rumus  $berat = bmi \times (tinggiBadan \times tinggiBadan)$ . Hasil perhitungan nantinya akan dibulatkan ke atas menggunakan fungsi *math.Ceil()* yang diimpor dari library “math” dan akan ditampilkan dengan satuan kilogram (Kg) seperti yang terlihat pada hasil screenshoot di atas.

### 3. Soal 3

#### Source Code

```
package main

import (
    "fmt"
    "math"
)

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

    fmt.Print("Masukkan titik koordinat A: ")
    fmt.Scan(&ax, &ay)
    fmt.Print("Masukkan titik koordinat B: ")
    fmt.Scan(&bx, &by)
    fmt.Print("Masukkan titik koordinat C: ")
    fmt.Scan(&cx, &cy)

    sisiAB := math.Sqrt(math.Pow(bx - ax, 2) + math.Pow(by - ay, 2))
    sisiBC := math.Sqrt(math.Pow(cx - bx, 2) + math.Pow(cy - by, 2))
    sisiCA := math.Sqrt(math.Pow(ax - cx, 2) + math.Pow(ay - cy, 2))
}
```

```

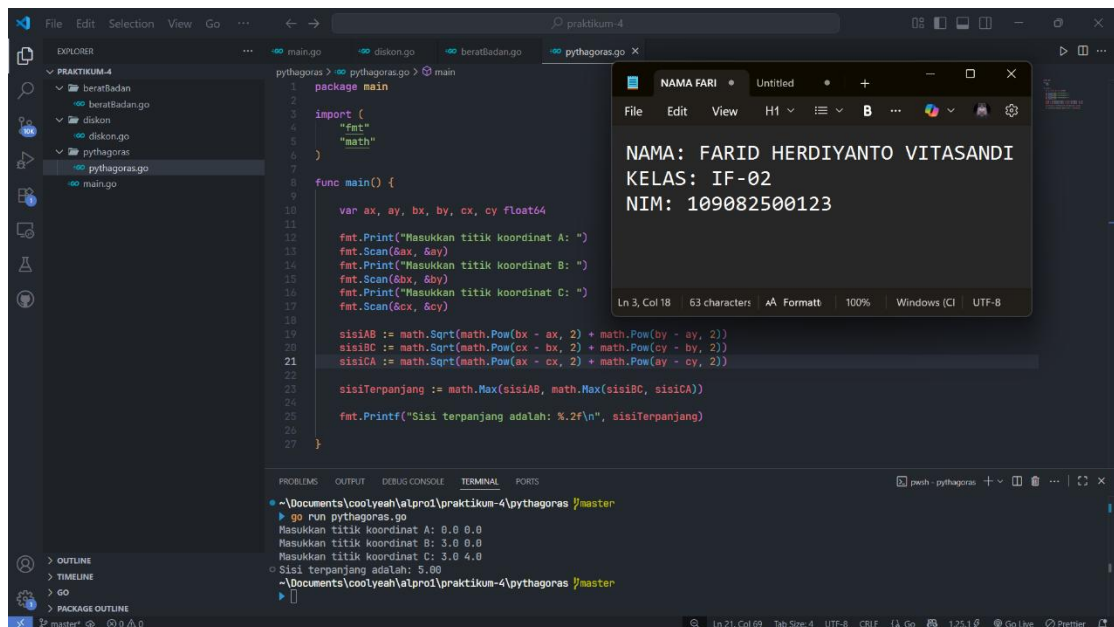
    sisiTerpanjang := math.Max(sisiAB, math.Max(sisiBC,
sisiCA))

    fmt.Printf("Sisi terpanjang adalah: %.2f\n",
sisiTerpanjang)

}

```

## Screenshoot program



## Deskripsi program

Program di atas dibuat untuk menentukan sisi terpanjang dari sebuah segitiga berdasarkan tiga titik koordinat (A, B, C). User akan diminta untuk memasukkan nilai koordinat masing-masing titik. Kemudian, program akan mulai menghitung panjang ketiga sisi segitiga menggunakan rumus jarak antar dua titik, yaitu,  $\sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}$ , dan kemudian akan disimpan pada variable `sisiAB`, `sisiBC`, `sisiCA`. Setelah semua sisi sudah diketahui, program kemudian akan menggunakan fungsi `math.Max()` yang diimpor dari library “math” untuk mencari sisi mana yang paling panjang. Selanjutnya, sisi terpanjang akan ditampilkan seperti yang terlihat pada gambar di atas.