

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

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



Disusun oleh:

Yedija Johanan Siregar

109082500075

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

**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

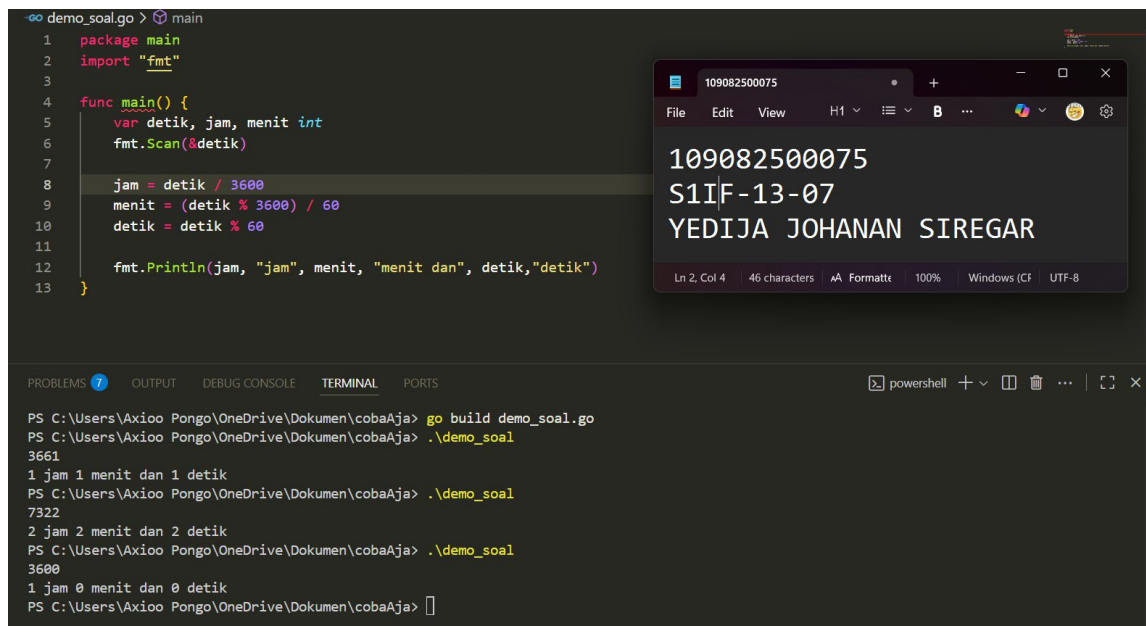
    fmt.Println(jam, "jam", menit, "menit dan",
detik,"detik")

}
```

Screenshoot program

//tambahkan tangkapan layar dari program (boleh lebih dari 1 jika diperlukan)

CONTOH TANGKAPAN LAYAR:



The screenshot shows a Go IDE with a source code editor and a terminal. The source code is a Go program that takes an input in seconds and converts it to hours, minutes, and seconds. The terminal shows the execution of the program with three test cases: 3661 seconds (1 hour, 1 minute, 1 second), 7322 seconds (2 hours, 2 minutes, 2 seconds), and 3600 seconds (1 hour, 0 minutes, 0 seconds).

```
demo_soal.go > main
1 package main
2 import "fmt"
3
4 func main() {
5     var detik, jam, menit int
6     fmt.Scan(&detik)
7
8     jam = detik / 3600
9     menit = (detik % 3600) / 60
10    detik = detik % 60
11
12    fmt.Println(jam, "jam", menit, "menit dan", detik, "detik")
13 }
```

Terminal Output:

```
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go build demo_soal.go
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> .\demo_soal
3661
1 jam 1 menit dan 1 detik
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> .\demo_soal
7322
2 jam 2 menit dan 2 detik
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> .\demo_soal
3600
1 jam 0 menit dan 0 detik
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja>
```

Deskripsi program

Program ini meminta pengguna untuk memasukkan jumlah detik, lalu mengubah detik tersebut menjadi satuan waktu yang lebih mudah dibaca, yaitu dalam jam, menit, dan detik. Kita perlu memasukkan perintah `package main`, ini menandakan program utama `go`. Lalu, kita perlu mengimpor paket `fmt` yang digunakan untuk input dan output, seperti `Print` dan `Scan`, lalu kita bisa munculkan perintah `func main()` untuk fungsi utama yang otomatis akan dijalankan saat program dijalankan. Kita masuk bagian input, maka dari itu, kita perlu menambahkan perintah `"var detik, jam, menit int"` (sesuai yang diminta), dan `"fmt.Scan(&detik)"`. Lalu kita masuk ke bagian proses perhitungan, masukkan rumus menghitung jam, lalu rumus menit, dan yang terakhir kita perlu mengambil sisa detik terakhir setelah jam dan menit dihitung menggunakan `"detik = detik % 60"`. Lalu kita masuk ke bagian yang terakhir bagian output, yang dimana untuk menampilkan hasil perhitungan ke layar dengan menggunakan `fmt.Println(jam, "jam", menit, "menit dan", detik, "detik")`.

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

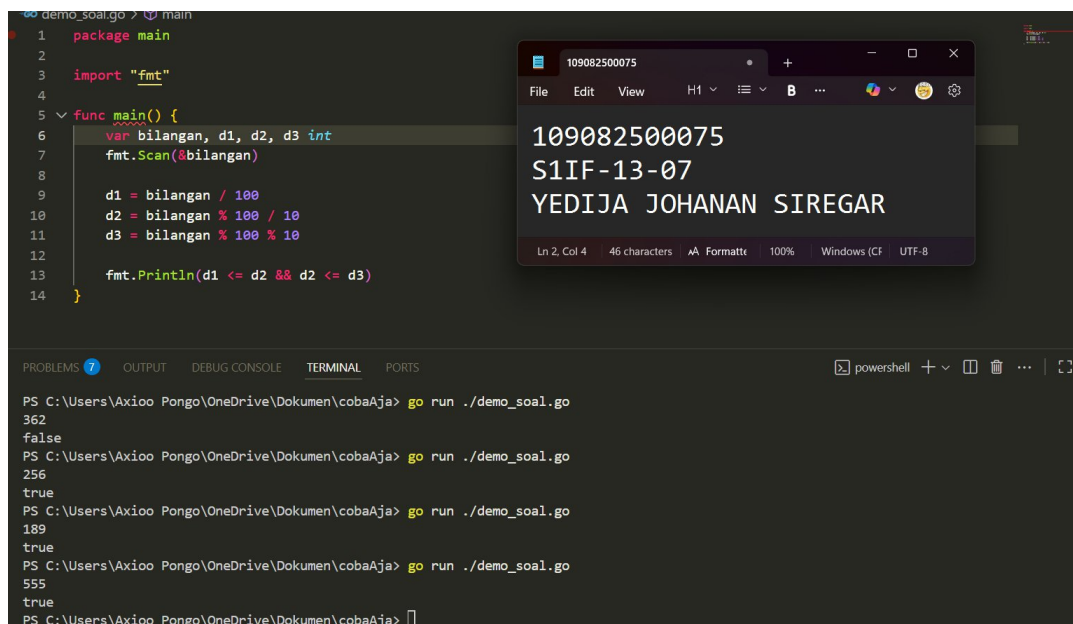
    d3 = bilangan % 100 % 10

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

}

```

Screenshoot program



The screenshot shows a Go program being executed in a terminal window. The program takes an integer input and checks if its digits are in non-decreasing order. The text editor window shows the source code of the program.

```

demo_soal.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan, d1, d2, d3 int
7     fmt.Scan(&bilangan)
8
9     d1 = bilangan / 100
10    d2 = bilangan % 100 / 10
11    d3 = bilangan % 100 % 10
12
13    fmt.Println(d1 <= d2 && d2 <= d3)
14 }

```

Terminal output:

```

PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
362
false
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
256
true
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
189
true
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
555
true
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja>

```

The text editor window shows the input and output of the program:

```

109082500075
S1IF-13-07
YEDIJA JOHANAN SIREGAR

```

Deskripsi program

Program ini meminta pengguna memasukkan bilangan tiga digit, lalu memeriksa apakah angka-angka di dalam bilangan tersebut tersusun secara menaik

(urut dari kecil ke besar). Untuk kita bisa tahu, apa yang di input oleh user itu bisa menunjukkan true dan false, maka kita bisa memunculkan perintah package main yang menandakan bahwa ini adalah program utama yang bisa dijalankan, lalu kita perlu mengimpor fmt untuk keperluan input dan output, seperti Println dan Scan, setelah kita munculkan perintah func main () {, yang menandakan ini adalah fungsi utama program, tempat semua perintah dijalankan. Lalu, kita masuk ke bagian deklarasi variabel, kita perlu munculkan perintah var bilangan, d1, d2, d3 int, lalu kita perlu munculkan perintah fmt.Scan(&bilangan), perintah ini meminta pengguna memasukkan sebuah bilangan bulat, nilai input ini akan disimpan dalam variabel bilangan. Lalu, kita masuk ke bagian penghitungan, dalam tahap ini program akan diminta untuk mengambil digit pertama (ratusan) dari bilangan, lalu mengambil digit kedua (puluhan), dan sampai digit satuan (d1 = bilangan/100 (ratusan), d2 = bilangan % 100/10 (puluhan), d3 = bilangan % 100 % 10 (satuan)). Lalu, kita masuk ke bagian proses output, dengan kita menampilkan fmt.Println(d1 <= d2 && d2 <= d3) untuk menampilkan hasil output ke layar.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var beratBadan, tinggiBadan, bmi float64

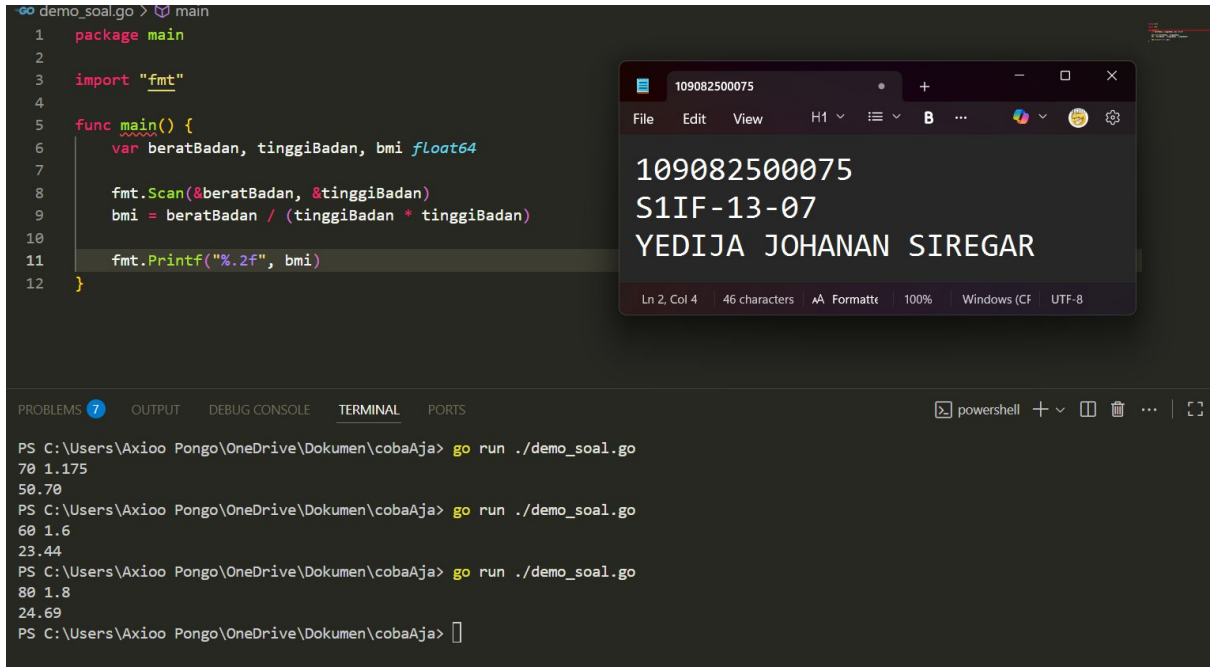
    fmt.Scan(&beratBadan, &tinggiBadan)

    bmi = beratBadan / (tinggiBadan * tinggiBadan)

    fmt.Printf("%.2f", bmi)
```

```
}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution results in a terminal. The code defines a `main` function that takes weight and height as input, calculates BMI, and prints the result. The terminal shows three test runs with different inputs and their corresponding BMI outputs.

```
demo_soal.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var beratBadan, tinggiBadan, bmi float64
7
8     fmt.Scan(&beratBadan, &tinggiBadan)
9     bmi = beratBadan / (tinggiBadan * tinggiBadan)
10
11     fmt.Printf("%.2f", bmi)
12 }
```

Terminal Output:

```
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
70 1.175
50.70
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
60 1.6
23.44
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
80 1.8
24.69
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja>
```

Output Window Content:

```
109082500075
S1IF-13-07
YEDIJA JOHANAN SIREGAR
```

Deskripsi program

Program ini digunakan untuk menghitung nilai BMI (Body Mass Index) seseorang berdasarkan berat badan (dalam kilogram) dan tinggi badan (dalam meter) yang dimasukkan oleh pengguna. Untuk kita bisa tahu berapa nilai BMI pada seseorang, maka kita perlu membuat kode pertama `package main`, menunjukkan bahwa program ini adalah program utama, lalu kita perlu mengimpor paket `fmt` untuk keperluan input dan output seperti `Printf` dan `Scan`, kita perlu juga memunculkan kode `func main() {`, ini menunjukkan fungsi utama seluruh perintah program dijalankan. Kita masuk ke bagian proses input, kita perlu memunculkan kode `var beratBadan, tinggiBadan, bmi float64` (`float 64` untuk menyimpan angka desimal), dan perlu memunculkan kode `fmt.Scan(&beratBadan, &tinggiBadan)`. Setelah itu, kita masuk ke bagian perhitungan, program akan diminta untuk menghitung berat badan dan tinggi badan seseorang untuk bisa melihat nilai BMI nya, maka dari itu kita masukkan rumus BMI yang bisa didapat di mesin pencarian (google, chrome, dll). Lalu, kita masuk ke bagian output, maka kita bisa memunculkan kode `fmt.Printf("%.2f", bmi)`, untuk menampilkan hasil BMI ke layar user.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var totalBelanja, diskon int

    fmt.Printf("Masukkan total belanja: ")

    fmt.Scan(&totalBelanja)

    fmt.Printf("Masukkan persentase diskon: ")

    fmt.Scan(&diskon)


    totalAkhir      :=      float64(totalBelanja)      -
(float64(totalBelanja) * float64(diskon) / 100)

    fmt.Printf("%.0f\n", totalAkhir)

}
```

Screenshoot program

The screenshot shows a Go program in a code editor and its execution in a terminal. The code defines a `main` function that prompts for a total purchase amount and a discount percentage, calculates the final amount after discount, and prints it. The terminal shows three test runs with different inputs. A browser window in the foreground displays the output of the program: the total purchase amount 109082500075, the discount code S1IF-13-07, and the name YEDIJA JOHANAN SIREGAR.

```
demo_soal.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var totalBelanja, diskon int
7     fmt.Printf("Masukkan total belanja: ")
8     fmt.Scan(&totalBelanja)
9     fmt.Printf("Masukkan persentase diskon: ")
10    fmt.Scan(&diskon)
11
12    // hitung total akhir setelah diskon
13    totalAkhir := float64(totalBelanja) - (float64(totalBelanja) * float64(diskon) / 100)
14
15    fmt.Printf("%.0f\n", totalAkhir)
16 }
```

```
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
Masukkan total belanja: 100000
Masukkan persentase diskon: 10%
90000
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
Masukkan total belanja: 200000
Masukkan persentase diskon: 20%
160000
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
Masukkan total belanja: 150000
Masukkan persentase diskon: 15%
127500
```

109082500075
S1IF-13-07
YEDIJA JOHANAN SIREGAR

Deskripsi program

Program ini meminta dua masukan: total belanja dan persen diskon dan menghitung total akhir setelah diskon menggunakan rumus $\text{total akhir} = \text{total belanja} - (\text{total belanja} \times \text{diskon} / 100)$ dan menampilkan hasil akhir nya dalam bilangan bulat. Kita bisa menggunakan kode go, dengan cara memunculkan perintah `package main`, ini menandakan bahwa program utama yang bisa dijalankan, lalu kita bisa mengimpor paket `fmt` untuk operasi input dan output, seperti `Printf` dan `Scan`, dan kita bisa memunculkan fungsi `main() {`, bagian pertama yang dijalankan Ketika program ini dimulai. Lalu, kita bisa masuk ke bagian input, dengan memunculkan `var totalBelanja, diskon int`, dan kita bisa menggunakan `fmt.Scan(&totalBelanja)` dan `fmt.Scan(&diskon)` untuk scan input dari pengguna, dan bisa menambahkan sedikit perintah input dengan menggunakan `fmt.Printf("masukkan perintah: ")`. Lalu, program akan melakukan proses perhitungan diskon dengan menggunakan rumus diatas. Dan, hasil akhirnya akan ditampilkan dengan memunculkan dengan kode `fmt.Printf("%.0f\n", totalAkhir)`.

2. Tugas 2

Source code


```

package main

import "fmt"

func main() {

    var bmi, tinggiBadan, beratBadan float64

    // Input dua bilangan: BMI dan tinggi badan (dalam meter)

    fmt.Printf("Masukkan BMI dan tinggi badan: ")

    fmt.Scan(&bmi, &tinggiBadan)

    // Rumus mencari berat badan

    beratBadan = bmi * (tinggiBadan * tinggiBadan)

    fmt.Printf("%.0f\n", beratBadan)

}

```

Screenshoot program

The screenshot shows a Go program being executed in a terminal window. The program calculates weight based on BMI and height. The code is as follows:

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var bmi, tinggiBadan, beratBadan float64
7
8     // Input dua bilangan: BMI dan tinggi badan (dalam meter)
9     fmt.Printf("Masukkan BMI dan tinggi badan: ")
10    fmt.Scan(&bmi, &tinggiBadan)
11
12    // Rumus mencari berat badan
13    beratBadan = bmi * (tinggiBadan * tinggiBadan)
14
15    fmt.Printf("%.0f\n", beratBadan)
16 }

```

The terminal output shows three successful runs of the program with the following inputs and outputs:

```

PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
Masukkan BMI dan tinggi badan: 22.85 1.75
70
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
Masukkan BMI dan tinggi badan: 23.43 1.6
60
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./demo_soal.go
Masukkan BMI dan tinggi badan: 24.69 1.8
80
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja>

```

In the background, a browser window displays the output of the program for the first run, showing the BMI and height values entered, and the calculated weight:

```

109082500075
S1IF-13-07
YEDIJA JOHANAN SIREGAR

```

Deskripsi program

Program ini meminta input berupa BMI dan tinggi badan (meter), dan menghitung berat badan dengan rumus $\text{berat} = \text{BMI} \times (\text{tinggi}^2)$, dan menampilkan hasil dalam kilogram, dibulatkan ke bilangan bulat. Kita bisa menggunakan bahasa pemrograman Go, dengan memunculkan package main, untuk menandakan program utama, lalu kita mengimpor paket fmt untuk input/output, seperti Printf dan Scan. Dan, kita masuk ke bagian input dengan menggunakan `fmt.Scan(&bmi, &tinggiBadan)`, program ini meminta dua input dari pengguna, berupa nilai BMI dan nilai tinggi badan (meter). Setelah itu kita masuk ke bagian proses penghitungan, program akan diminta untuk menghitung berat badan seseorang dengan menggunakan rumus $\text{beratBadan} = \text{bmi} \times (\text{tinggiBadan} \times \text{tinggiBadan})$. Dan untuk hasil output kita bisa memunculkan kode perintah `fmt.Printf("%.0f\n", beratBadan)`.

3. Tugas 3

Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var x1, y1, x2, y2, x3, y3 float64

    // Input tiga titik koordinat (masing-masing dua nilai: x
dan y)
    fmt.Scan(&x1, &y1)
    fmt.Scan(&x2, &y2)
    fmt.Scan(&x3, &y3)

    // Menghitung panjang sisi-sisi segitiga dengan rumus jarak
dua titik
    ab := math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))
    bc := math.Sqrt(math.Pow(x3-x2, 2) + math.Pow(y3-y2, 2))
}
```

```

ca := math.Sqrt(math.Pow(x1-x3, 2) + math.Pow(y1-y3, 2))

// Menentukan sisi terpanjang
max := ab
if bc > max {
    max = bc
}
if ca > max {
    max = ca
}

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

```

Screenshoot program

The screenshot shows a Go program being executed in a terminal. The program prompts the user to enter three 2D coordinates (x1, y1), (x2, y2), and (x3, y3). It then calculates the lengths of the three sides of the triangle formed by these points using the distance formula. It determines the longest side and displays the result with two decimal places.

The output of the program is displayed in a web browser window, showing the following text:

```

109082500075
S1IF-13-07
YEDIJA JOHANAN SIREGAR

```

Deskripsi program

Program ini meminta tiga titik koordinat 2D sebagai input, menghitung panjang ketiga sisi segitiga dengan rumus jarak antar titik, menentukan sisi yang paling panjang, dan menampilkan hasilnya dengan dua angka di belakang koma. Kita bisa menggunakan bahasa pemrograman Go dengan memunculkan kode perintah package

main, ini merupakan fondasi utama dari bahasa Go, lalu kita perlu mengimpor paket `fmt`, agar kita bisa memakai fungsi `math.Sqrt()` (akar kuadrat) dan `math.Pow()` (pemangkatan). Setelah itu, kita di bagian input kita perlu memunculkan kode perintah `fmt.Scan(&x1, &y1)` program ini meminta input dari tiga koordinat, titik A, B, dan C. Setelah itu, program akan diminta untuk menghitung masukan (input) dari pengguna menggunakan rumus jarak dua titik (teorema Pythagoras), lalu program akan membandingkan ketiga sisi tersebut (ab , bc , ca) dan mencari paling besar menggunakan `if`. Setelah itu, program akan menampilkan hasil dari perhitungan ke layar user/pengguna dengan memunculkan kode perintah `fmt.Printf("%.2f\n", max)`.