

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 04
I/O, TIPE DATA & VARIABEL



Disusun oleh:
NUR FITRI RACHMADILLA DEWANTI
109082500057
S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

Pemberi soal : Akhsan Sabili_109082500062

Soal 1

Deskripsi soal

Sebuah program Go digunakan untuk menghitung nilai rata-rata ujian siswa yang terdiri dari maple IPA, MTK, dan B.Indonesia dengan bilangan bulat. Program akan mengeluarkan output berupa rata-rata dari ketiga nilai tersebut

Contoh Input/Output

Input	Output
85 70 78	77.67
80 90 70	80.00
80 82 89	83.67

JAWAB

Source Code

```
package main

import "fmt"

func main() {
    var ipa, mtk, indo int

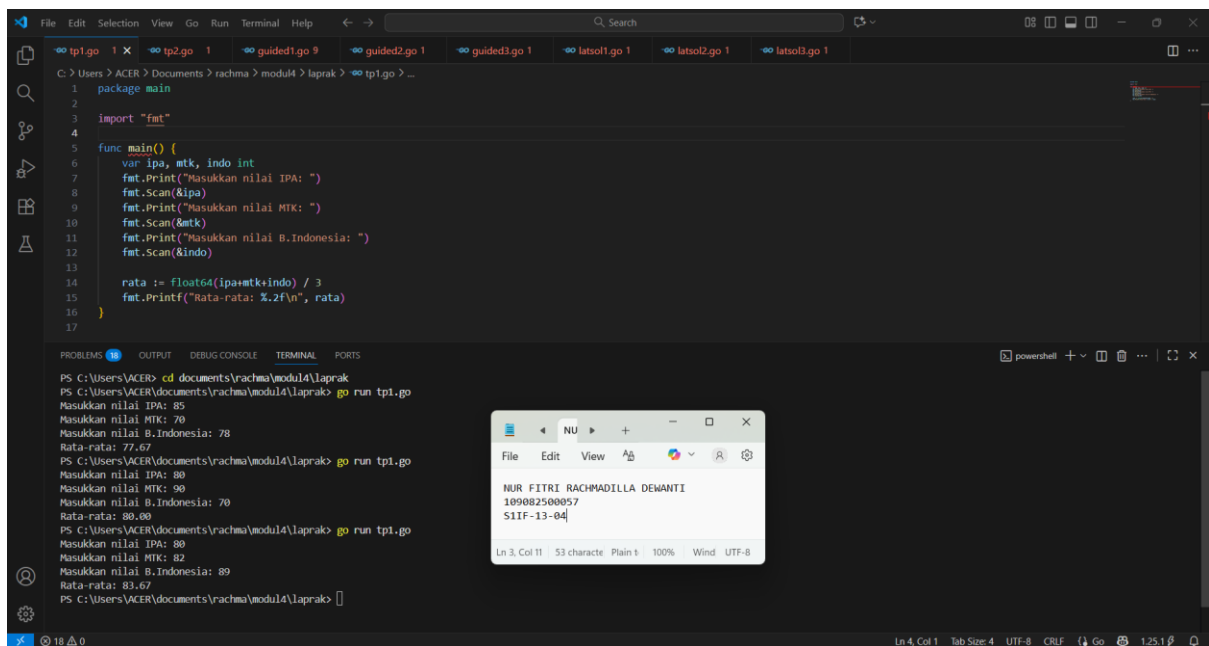
    fmt.Print("Masukkan nilai IPA: ")
    fmt.Scan(&ipa)

    fmt.Print("Masukkan nilai MTK: ")
    fmt.Scan(&mtk)

    fmt.Print("Masukkan nilai B.Indonesia: ")
    fmt.Scan(&indo)

    rata := float64(ipa+mtk+indo) / 3
    fmt.Printf("Rata-rata: %.2f\n", rata)
}
```

Screenshoot program



Soal 2

Deskripsi soal

Buatlah program dengan Bahasa Go untuk menentukan umur anak berdasarkan hari jumlah hari yang diketahui, jika 1 tahun adalah 365 hari, dan 1 tahun terdiri dari 12 bulan, sedangkan 1 bulan adalah 30 hari

Contoh Input/Output

Input	Output
765	Umur 2 tahun 1 bulan 5 hari
1339	Umur 3 tahun 8 bulan 4 hari
16890	Umur 46 tahun 3 bulan 10 hari

JAWAB

Source Code

```
package main

import "fmt"

func main() {

    var hari int

    fmt.Print("Masukkan jumlah hari: ")

    fmt.Scan(&hari)
```

```

    tahun := hari / 365

    sisa := hari % 365

    bulan := sisa / 30

    hariSisa := sisa % 30


    fmt.Printf("Umur %d tahun %d bulan %d hari\n", tahun,
bulan, hariSisa)

}

```

Screenshoot program

The screenshot shows a Go IDE with the following source code in `tp2.go`:

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var hari int
7     fmt.Print("Masukkan jumlah hari: ")
8     fmt.Scan(&hari)
9
10    tahun := hari / 365
11    sisa := hari % 365
12    bulan := sisa / 30
13    hariSisa := sisa % 30
14
15    fmt.Printf("Umur %d tahun %d bulan %d hari\n", tahun, bulan, hariSisa)
16
17 }

```

The terminal output shows the program being run multiple times with different inputs:

```

PS C:\Users\ACER> cd documents\rachma\modul4\laprak
PS C:\Users\ACER\documents\rachma\modul4\laprak> go run tp2.go
Masukkan jumlah hari: 765
Umur 2 tahun 1 bulan 5 hari
PS C:\Users\ACER\documents\rachma\modul4\laprak> go run tp2.go
Masukkan jumlah hari: 1339
Umur 3 tahun 8 bulan 4 hari
PS C:\Users\ACER\documents\rachma\modul4\laprak> go run tp2.go
Masukkan jumlah hari: 16890
Umur 46 tahun 3 bulan 10 hari
PS C:\Users\ACER\documents\rachma\modul4\laprak>

```

A small window titled "NU" is also visible, displaying the following text:

```

HUR FITRI RACHMADILLA DEWANTI
109082500057
S11F-13-04

```

LATIHAN KELAS – GUIDED

1. Guided 1

Source Code

```
package main

import "fmt"

func main() {

    var detik, jam, menit int

    fmt.Scan(&detik)

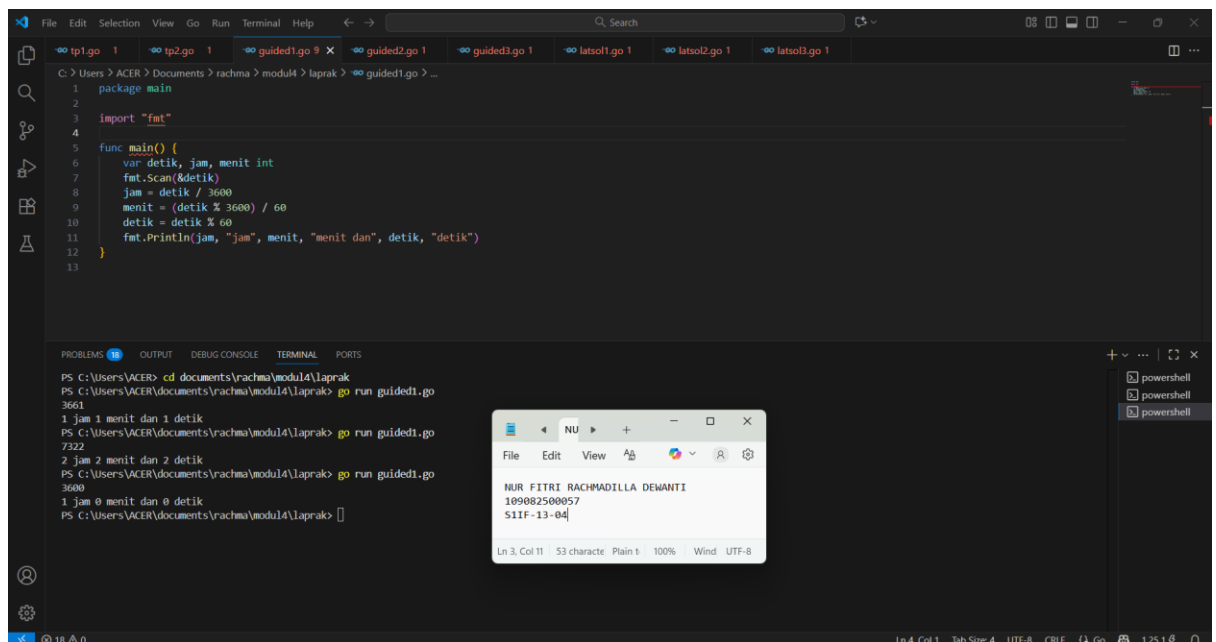
    jam = detik / 3600

    menit = (detik % 3600) / 60

    detik = detik % 60

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

Screenshoot program



Deskripsi program

Program ini dibuat untuk mengubah satuan waktu dari detik menjadi jam, menit, dan detik. Dengan memasukkan jumlah detik, maka program akan menghitung berapa jam dengan membagi jumlah detik dengan 3600 karena 1 jam = 3600 detik.

Setelah menghitung jam, sisa detiknya digunakan untuk mencari jumlah menit dengan cara membagi sisa itu dengan 60. Terakhir, adalah detik yang tersisa setelah konversi jam dan menit.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var bilangan, d1, d2, d3 int

    fmt.Scan(&bilangan)

    d1 = bilangan / 100

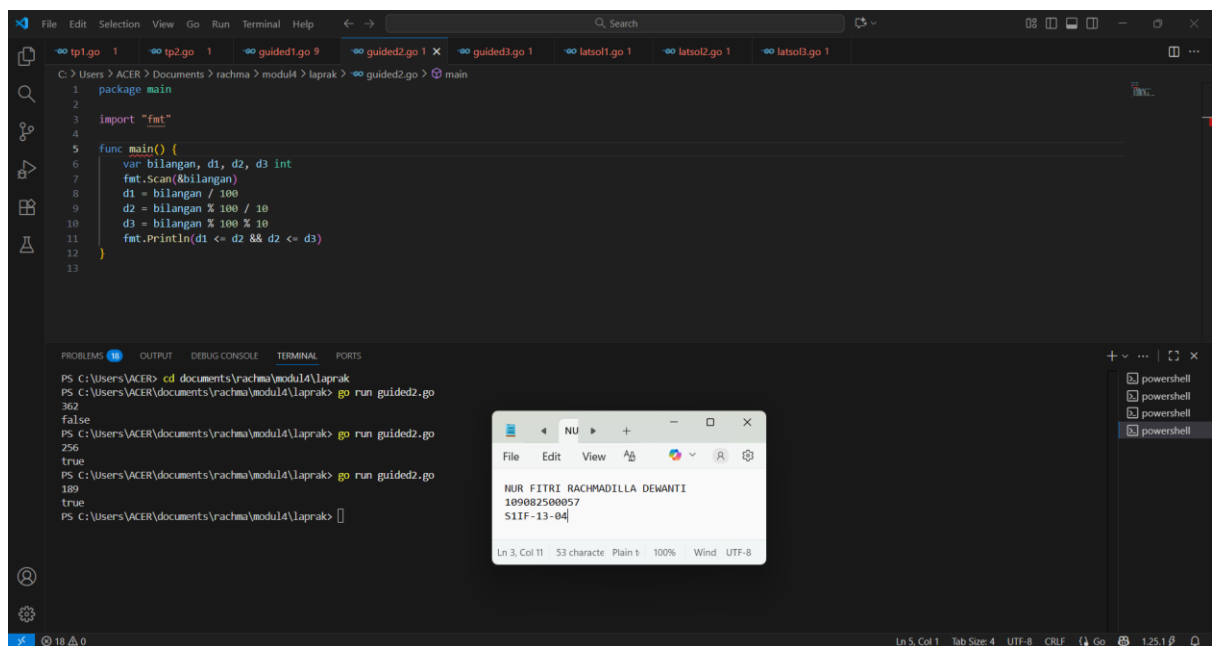
    d2 = bilangan % 100 / 10

    d3 = bilangan % 100 % 10

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

}
```

Screenshoot program



Deskripsi program

Program ini di buat untuk mengecek apakah 3 digit yang di input tersusun secara berurutan dari kecil ke besar. Misalnya, pengguna memasukkan sebuah bilangan 3 digit (362) lalu program akan memsiahkan tiap digitnya

d1: angka ratusan (bilangan / 100)

d2: angka puluhan (bilangan % 100 / 100)

d3: angka satuan (bilangan % 100 % 100)

Jika urutan digitnya naik atau sama dari kiri ke kanan (angkanya tersusun dari kecil ke besar), maka hasilnya true. Jadi untuk input 362, program akan menampilkan false, karena urutannya tidak naik dari kiri ke kanan (3 → 6 → 2 itu turun di bagian akhir).

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 IDE with a file named `guided3.go` open. The code defines a `main` function that takes three command-line arguments: `beratBadan` (weight), `tinggiBadan` (height), and `bmi` (BMI). It uses `fmt.Scan` to read the first two arguments and `fmt.Printf` to print the BMI value with two decimal places.

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

The terminal output shows the program being run with the following inputs: `70`, `1.75`, and `22.86`. The output is `22.86`.

```
PS C:\Users\ACER> cd documents\rachma\modul4\laprak
PS C:\Users\ACER\documents\rachma\modul4\laprak> go run guided3.go
70 1.75
22.86
PS C:\Users\ACER\documents\rachma\modul4\laprak> go run guided3.go
69 1.6
23.44
PS C:\Users\ACER\documents\rachma\modul4\laprak> go run guided3.go
80 1.8
24.69
PS C:\Users\ACER\documents\rachma\modul4\laprak>
```

A small window titled `NU` is also visible, displaying the text: `NUR FITRI RACHMADILLA DEWANTI`, `109082500057`, and `S1IF-13-04`.

Deskripsi program

Program ini dibuat untuk menghitung nilai BMI seseorang. Inputan berupa berat badan dalam kilogram dan tinggi badan dalam meter. Program akan menghitung BMI dengan rumus:

$$\text{BMI} = \text{berat badan} / (\text{tinggi badan} \times \text{tinggi badan})$$

Setelah dihitung, program akan menampilkan hasil BMI dengan dua angka di belakang koma.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var harga float64

    var diskon float64


    fmt.Print("Masukkan harga barang: ")

    fmt.Scan(&harga)

    fmt.Print("Masukkan persentase diskon: ")

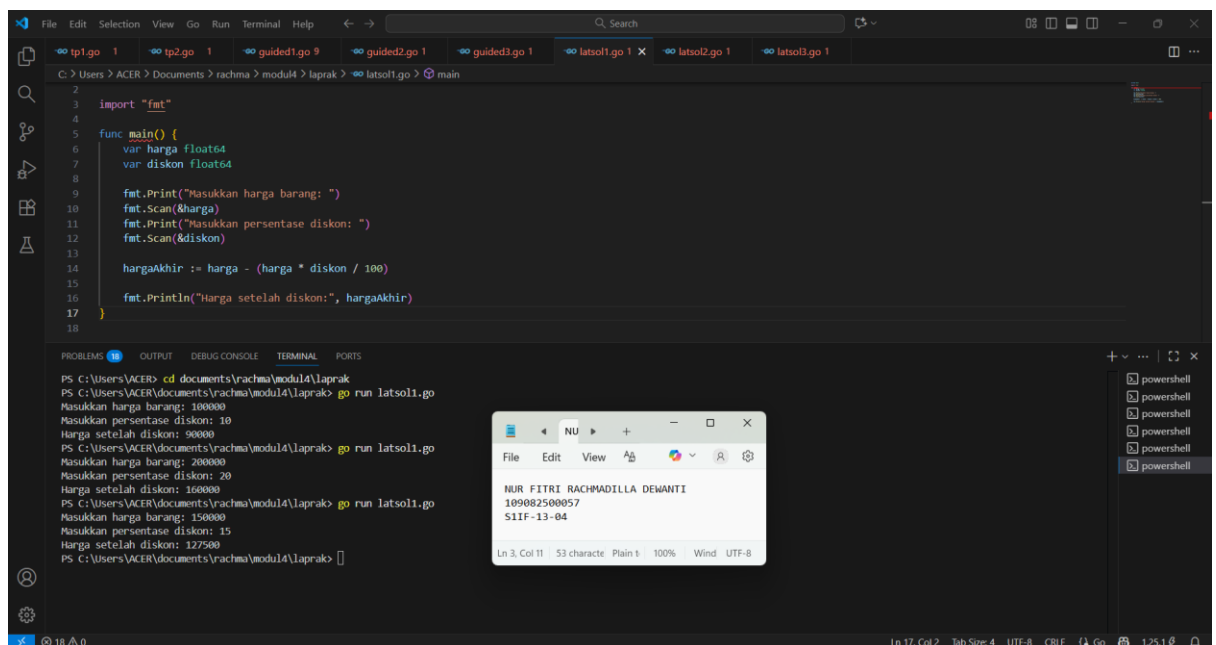
    fmt.Scan(&diskon)

    hargaAkhir := harga - (harga * diskon / 100)


    fmt.Println("Harga setelah diskon:", hargaAkhir)

}
```

Screenshoot program



Deskripsi program

Program ini dibuat untuk menghitung barang setelah diberi diskon dengan memasukkan harga barang dan persentase diskon. Lalu, program akan menghitung dengan rumus:

$$\text{harga} * \text{diskon} / 100$$

Setelah itu, program akan menghitung hasil akhir dengan cara:

$$\text{harga} - (\text{harga} * \text{diskon} / 100)$$

2. Tugas 2

Source code

```
package main

import (

    "fmt"

    "math"

)
```

```

func main() {

    var bmi, tinggi float64


    fmt.Print("Masukkan nilai BMI: ")

    fmt.Scan(&bmi)

    fmt.Print("Masukkan tinggi badan (meter): ")

    fmt.Scan(&tinggi)


    berat := bmi * (tinggi * tinggi)

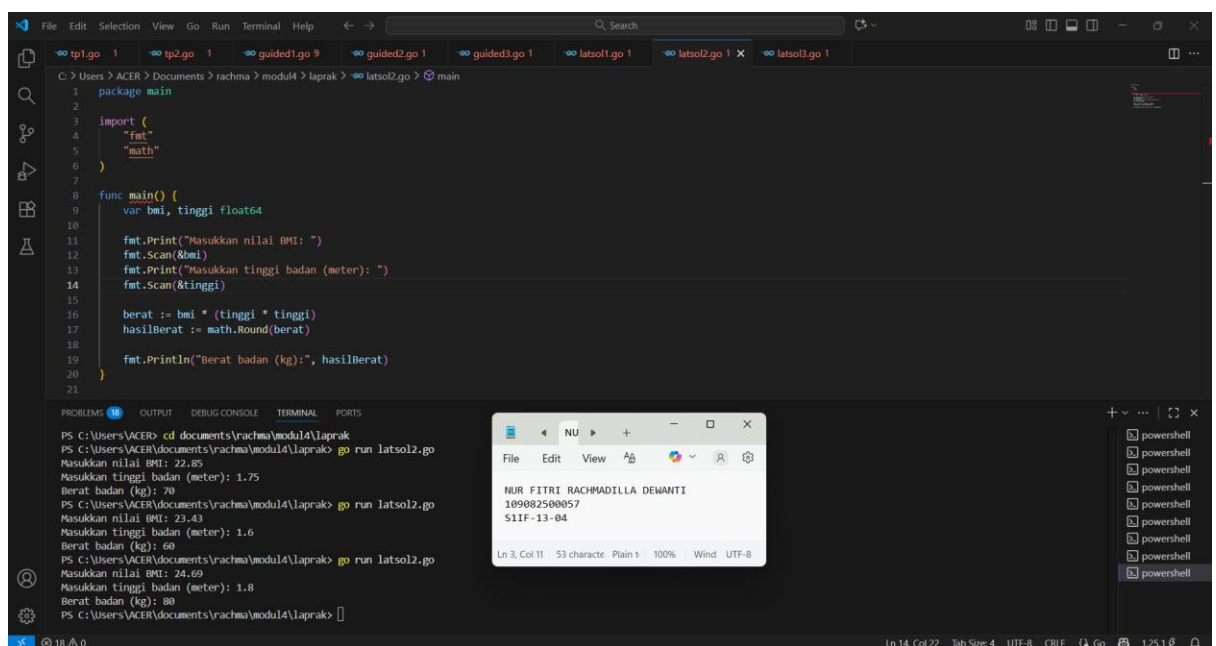
    hasilBerat := math.Round(berat)


    fmt.Println("Berat badan (kg):", hasilBerat)

}

```

Screenshoot program



Deskripsi program

Program ini dibuat untuk menghitung berat badan berdasarkan nilai BMI dan tinggi badan. Inputan yang dilakukan adalah memasukkan nilai BMI dan tinggi badan dalam meter dengan rumus:

$\text{berat} = \text{BMI} \times (\text{tinggi} \times \text{tinggi})$

Setelah mendapatkan hasil berat, program akan menggunakan fungsi `math.Round()` untuk membulatkan hasilnya ke angka bulat terdekat dan menampilkan berat badan dalam satuan kilogram.

3. Tugas 3

Source code

```
package main

import (
    "fmt"
    "math"
)

func distance(x1, y1, x2, y2 float64) float64 {
    return math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))
}

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

    fmt.Scan(&x1, &y1)
    fmt.Scan(&x2, &y2)
    fmt.Scan(&x3, &y3)

    ab := distance(x1, y1, x2, y2)
    bc := distance(x2, y2, x3, y3)
    ca := distance(x3, y3, x1, y1)

    max := ab
    if bc > max {
        max = bc
    }
}
```

```

    if ca > max {
        max = ca
    }

    fmt.Printf("%.2f\n", max)
}

```

Screenshoot program

The screenshot shows the VS Code editor with the file 'latso3.go' open. The code defines a 'main' package with imports for 'fmt' and 'math'. It includes a 'distance' function that calculates the Euclidean distance between two points. The 'main' function scans for three points (x1, y1), (x2, y2), and (x3, y3), calculates the distances between them (ab, bc, ca), and then determines the maximum distance using a series of if-statements. The result is printed using 'fmt.Printf'.

A terminal window is open at the bottom, showing the command 'go run latso3.go' being executed. The output of the program is displayed in a separate window:

```

NUR FITRI RACHMADILLA DENAWATI
109082500057
S11F-13-04

```

This screenshot shows the same VS Code editor with 'latso3.go' open. The terminal window at the bottom shows the command 'go run latso3.go' being executed multiple times. The output of the program is displayed in a separate window, showing the same information as the first screenshot:

```

NUR FITRI RACHMADILLA DENAWATI
109082500057
S11F-13-04

```

Deskripsi program

Program ini dibuat untuk menentukan sisi terpanjang dari sebuah segitiga berdasarkan tiga titik koordinat (A, B, C)

Inputan berupa tiga titik koordinat yaitu titik A (x_1, y_1), B(x_2, y_2), dan C (x_3, y_3).

Setelah semua sisi di hitung, program akan membandingkan panjangnya untuk mencari sisi yang paling panjang. Hasil yang di tampilkan berupa panjang sisi terpanjang dengan dua angka di belakang koma.

Hasil program saya berbeda dengan contoh di modul (modul menghasilkan 4.47 sedangkan program menghasilkan 5.00).

Setelah dihitung manual, hasil 5.00 adalah nilai yang benar untuk jarak terpanjang antara titik B(4,1) dan C(1,5).

Perbedaan ini kemungkinan disebabkan oleh kesalahan penulisan hasil pada modul atau perbedaan urutan titik saat perhitungan dilakukan.