

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 4
I/O, TIPE DATA DAN VARIABEL**



Disusun oleh:

Andromeda Alike Ramadhani

109082500156

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 bilangan_positif int

    fmt.Scan(&bilangan_positif)

    jam := bilangan_positif / 3600

    menit := bilangan_positif / 60 % 60

    detik := bilangan_positif % 60

    fmt.Printf("%d jam %d menit %d detik", jam, menit,
detik)

}
```

Screenshoot program



```
soal1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan_positif int
7     fmt.Scan(&bilangan_positif)
8
9     jam := bilangan_positif / 3600
10    menit := bilangan_positif / 60 % 60
11    detik := bilangan_positif % 60
12
13    fmt.Printf("%d jam %d menit %d detik", jam, menit, detik)
14 }
15
```

109082500156
IF-13-02
Andromeda Alika Ramadhani

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan M
odul 4 Input Output\soal1.go"
3661
1 jam 1 menit 1 detik
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan M
odul 4 Input Output\soal1.go"
7322
2 jam 2 menit 2 detik
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan M
odul 4 Input Output\soal1.go"
7115
1 jam 58 menit 35 detik
```

Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk mengkonversi detik ke jam, menit, dan detik. Langkah untuk membuat program tersebut antara lain yaitu:

1. Deklarasikan variabel yang akan kita gunakan. Variabel yang kita gunakan yaitu variabel bilangan positif dengan tipe data integer, karena masukkan berupa bilangan bulat positif. Tipe data integer berfungsi untuk menyimpan data numerik non desimal seperti 5, 2, 1, -2.
2. Gunakan "fmt.Scan(&bilangan_positif) untuk menginputkan bilangan positif yang akan dikonversi.
3. Selanjutnya gunakan modulus (%) dan div (/) untuk mengkonversi bilangan bulat positif yang akan dikonversi ke jam, menit, dan detik. Modulus (%) digunakan untuk memberikan hasil dari sisa bagi, sedangkan div (/) digunakan untuk menuliskan hasil bagi dalam bentuk bilangan bulat.
4. Terakhir, gunakan "fmt.Printf("%d jam %d menit %d detik", jam, menit, detik)" untuk menampilkan hasil dari bilangan bulat positif yang telah dikonversi ke jam, menit, dan detik. "fmt.Printf" berfungsi untuk menampilkan output dengan format tertentu. %d berfungsi untuk menampilkan format variabel yang digunakan, disini kita menggunakan tipe data integer. Oleh karena itu, kita menggunakan format %d. Terakhir, jangan lupa untuk mendeklarasikan variabel jam, menit, dan detik yang sudah dibuat sebelumnya di akhir.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var bilangan int
```

```

    fmt.Printf("Masukkan bilangan bulat positif: ")

    fmt.Scan(&bilangan)

    d1 := bilangan / 100

    d2 := bilangan / 100 % 10

    d3 := bilangan % 10

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

}

```

Screenshoot program

```

1  package main
2
3  import "fmt"
4
5  func main() {
6      var bilangan int
7      fmt.Printf("Masukkan bilangan bulat positif: ")
8      fmt.Scan(&bilangan)
9
10     d1 := bilangan / 100
11     d2 := bilangan / 100 % 10
12     d3 := bilangan % 10
13
14     fmt.Print(d1 <= d2 && d2 <= d3)
15 }
16

```

109082500156
IF-13-02
Andromeda Alika Ramadhani

```

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output\soal2\soal2.go"
Masukkan bilangan bulat positif: 362
false
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output\soal2\soal2.go"
Masukkan bilangan bulat positif: 425
true
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output\soal2\soal2.go"
Masukkan bilangan bulat positif: 256
true

```

Deskripsi program

Pada program diatas, kita diminta untuk menentukan apakah setiap digit pada suatu bilangan terurut membesar atau tidak. Jika terurut membesar, maka akan bernilai

true, sebaliknya jika tidak maka akan bernilai false. Langkah untuk membuat program tersebut antara lain yaitu:

1. Deklarasikan variabel yang akan kita gunakan. Karena disini kita akan menentukan apakah suatu bilangan itu terurut membesar atau mengecil, maka variabel yang kita butuhkan adalah bilangan dengan tipe data integer karena data bilangan yang digunakan adalah bilangan bulat positif. Tipe data integer berfungsi untuk menyimpan data numerik non desimal seperti 5, 2, 1, -2.
2. Gunakan "fmt.Scan(&bilangan)" untuk menginputkan data bilangan bulat positif
3. Selanjutnya gunakan modulus (%) dan div (/) untuk memecah bilangan yang akan kita input. Bilangan dipecah dari ratusan, puluhan, dan satuan. Modulus (%) digunakan untuk memberikan hasil dari sisa bagi, sedangkan div (/) digunakan untuk menuliskan hasil bagi dalam bentuk bilangan bulat.
4. Terakhir, gunakan "fmt.Print(d1 <= d2 && d2 <= d3)" untuk menentukan apakah bilangan tersebut terurut membesar atau tidak. Jika bilangan terurut membesar, maka akan bernilai true, sebaliknya jika tidak maka akan bernilai false. "fmt.Print" berfungsi untuk menampilkan output dari kode yang telah kita buat pada terminal.

3. Guided 3

Source Code

```
package main

import (

    "fmt"

)

func main() {

    var berat, tinggi float64

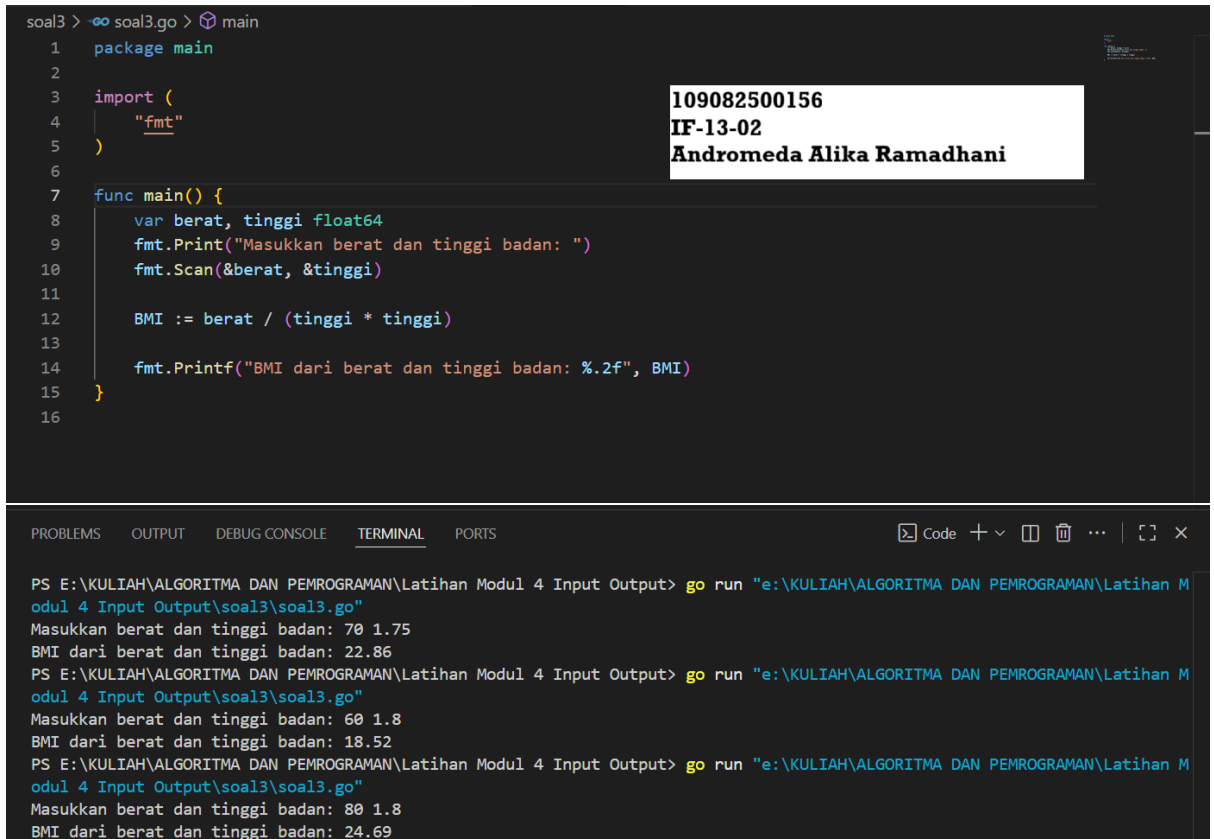
    fmt.Print("Masukkan berat dan tinggi badan: ")

    fmt.Scan(&berat, &tinggi)

    BMI := berat / (tinggi * tinggi)
```

```
        fmt.Printf("BMI dari berat dan tinggi badan: %.2f",  
        BMI)  
    }  
}
```

Screenshoot program



The screenshot shows a Go program in VS Code. The editor displays the source code for `soal3.go`, which calculates BMI. The terminal shows the execution of the program with three test cases. A student ID and name are visible in the top right corner of the editor window.

```
soal3 > go run soal3.go > main  
1 package main  
2  
3 import (  
4     "fmt"  
5 )  
6  
7 func main() {  
8     var berat, tinggi float64  
9     fmt.Print("Masukkan berat dan tinggi badan: ")  
10    fmt.Scan(&berat, &tinggi)  
11  
12    BMI := berat / (tinggi * tinggi)  
13  
14    fmt.Printf("BMI dari berat dan tinggi badan: %.2f", BMI)  
15 }  
16
```

109082500156
IF-13-02
Andromeda Alike Ramadhani

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output\soal3\soal3.go"
Masukkan berat dan tinggi badan: 70 1.75
BMI dari berat dan tinggi badan: 22.86
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output\soal3\soal3.go"
Masukkan berat dan tinggi badan: 60 1.8
BMI dari berat dan tinggi badan: 18.52
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Latihan Modul 4 Input Output\soal3\soal3.go"
Masukkan berat dan tinggi badan: 80 1.8
BMI dari berat dan tinggi badan: 24.69

Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk menghitung BMI berdasarkan berat dan tinggi badan. Langkah untuk membuat program tersebut antara lain yaitu:

1. Deklarasikan variabel yang akan kita gunakan. Karena disini kita akan menghitung BMI dari berat dan tinggi badan, maka variabel yang kita butuhkan adalah variabel berat dan tinggi dengan tipe data float. Tipe data float64 berfungsi untuk menyimpan data numerik desimal seperti 5.2, 1.2.
2. Gunakan "fmt.Scan(&berat, &tinggi)" untuk menginputkan data berat dan tinggi badan yang akan digunakan untuk menghitung BMI.
3. Selanjutnya masukkan rumus untuk menghitung BMI dari berat dan tinggi badan yang diinputkan dengan rumus BMI := berat / (tinggi * tinggi)
4. Terakhir, gunakan "fmt.Printf("BMI dari berat dan tinggi badan: %.2f", BMI)" untuk menampilkan hasil BMI yang telah dihitung sebelumnya. "fmt.Printf" berfungsi untuk menampilkan output dengan format tertentu. %.2f berfungsi

untuk menampilkan format variabel yang digunakan, disini kita menggunakan tipe float. Oleh karena itu, kita menggunakan format %f. Angka 2 pada format tersebut berfungsi untuk menambahkan 2 angka di belakang koma. Terakhir, jangan lupa untuk mendeklarasikan variabel BMI yang sudah dibuat sebelumnya di akhir.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var total, diskon int

    fmt.Print("Total Belanjaan: ")

    fmt.Scan(&total)

    fmt.Print("Diskon (dalam persen): ")

    fmt.Scan(&diskon)

    Total_Belanja := total - (total * diskon / 100)

    fmt.Printf("Total belanja anda adalah %d", Total_Belanja)
}
```

Screenshoot program

The image shows a Go program in a text editor and its execution output in a terminal. The program is a simple shopping cart calculator. It prompts the user for total purchase and discount, then calculates the final price after a 10% discount.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var total, diskon int
7     fmt.Print("Total Belanjaan: ")
8     fmt.Scan(&total)
9     fmt.Print("Diskon (dalam persen): ")
10    fmt.Scan(&diskon)
11
12    Total_Belanja := total - (total * diskon / 100)
13
14    fmt.Printf("Total belanja anda adalah %d", Total_Belanja)
15 }
16
```

The terminal output shows three runs of the program with increasing total purchase amounts and a constant 20% discount:

```
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4\soal1.go"
Total Belanjaan: 100000
Diskon (dalam persen): 10
Total belanja anda adalah 90000

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4\soal1.go"
Total Belanjaan: 250000
Diskon (dalam persen): 20
Total belanja anda adalah 200000

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4\soal1.go"
Total Belanjaan: 500000
Diskon (dalam persen): 50
Total belanja anda adalah 250000
```

1. Deklarasikan variabel yang akan kita gunakan. Disini kita menggunakan variabel total dan diskon karena kita akan mencari total harga setelah kita memperoleh diskon dengan besaran tertentu dengan menggunakan tipe data integer. Tipe data integer berfungsi untuk menyimpan data numerik non desimal seperti 5, 2, 1, -2.
2. Gunakan `"fmt.Scan(&total)"` dan `"fmt.Scan(&diskon)"` untuk menginputkan harga dan diskon.
3. Selanjutnya masukkan rumus untuk menghitung total belanja setelah mendapatkan diskon dengan rumus $\text{Total_Belanja} := \text{total} - (\text{total} * \text{diskon} / 100)$
4. Terakhir, gunakan `"fmt.Printf("Total belanja anda adalah %d", Total_Belanja)"` untuk menampilkan hasil dari harga total belanja yang diperoleh setelah mendapatkan diskon tertentu. `"fmt.Printf"` berfungsi untuk menampilkan output dengan format tertentu. `%f` berfungsi untuk menampilkan format

variabel yang digunakan, Karena disini kita menggunakan tipe data integer maka format yang kita gunakan yaitu format %d. Terakhir jangan lupa untuk mendeklarasikan variabel total_belanja di akhir.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var bmi, tinggi_badan float64

    fmt.Print("Nilai BMI: ")

    fmt.Scan(&bmi)

    fmt.Print("Tinggi badan: ")

    fmt.Scan(&tinggi_badan)

    berat_badan := bmi * (tinggi_badan * tinggi_badan)

    fmt.Printf("Berat Badan: %.f", berat_badan)

}
```

Screenshoot program

```
nomor2 > -o soal2.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bmi, tinggi_badan float64
7     fmt.Print("Nilai BMI: ")
8     fmt.Scan(&bmi)
9     fmt.Print("Tinggi badan: ")
10    fmt.Scan(&tinggi_badan)
11
12    berat_badan := bmi * (tinggi_badan * tinggi_badan)
13
14    fmt.Printf("Berat Badan: %.f", berat_badan)
15 }
16
```

109082500156
IF-13-02
Andromeda Alike Ramadhani

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4\nomor2\soa
l2.go"
Nilai BMI: 24.69
Tinggi badan: 1.8
Berat Badan: 80
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4\nomor2\soa
l2.go"
Nilai BMI: 25.75
Tinggi badan: 1.7
Berat Badan: 74
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modul4>

Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk menghitung berat badan dari BMI dan tinggi badan. Langkah untuk membuat program tersebut antara lain yaitu:

1. Deklarasikan variabel yang akan kita gunakan. Karena disini kita akan menghitung berat dari BMI dan tinggi badan, maka variabel yang kita butuhkan adalah variabel bmi dan tinggi dengan tipe data float. Tipe data float64 berfungsi untuk menyimpan data numerik desimal seperti 5.2, 1.2.
2. Gunakan "fmt.Scan(&bmi, &tinggi_badan)" untuk menginputkan data bmi dan tinggi badan yang akan digunakan untuk menghitung berat badan.
3. Selanjutnya masukkan rumus untuk menghitung berat dari bmi dan tinggi badan yang diinputkan dengan rumus $\text{berat_badan} := \text{bmi} * (\text{tinggi_badan} * \text{tinggi_badan})$ karena disini kita mencari berat badan, maka kita gunakan rumus bmi yang kemudian dipindahruaskan.
4. Terakhir, gunakan "fmt.Printf("berat badan: %.f", BMI)" untuk menampilkan hasil BMI yang telah dihitung sebelumnya. "fmt.Printf" berfungsi untuk menampilkan output dengan format tertentu. %.f berfungsi untuk menampilkan

format variabel yang digunakan, disini kita menggunakan tipe float. Oleh karena itu, kita menggunakan format %f. Terakhir, jangan lupa untuk mendeklarasikan variabel berat yang sudah dibuat sebelumnya di akhir.

3. Tugas 3

Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var x1, y1, x2, y2, x3, y3 float64
    fmt.Scan(&x1, &y1)
    fmt.Scan(&x2, &y2)
    fmt.Scan(&x3, &y3)

    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(x3-x1, 2) + math.Pow(y3-y1, 2))

    panjang := math.Max(ab, math.Max(bc, ca))

    fmt.Printf("Panjang: %.2f", panjang)

}
```

Screenshoot program

```
nomor3 > -go soal3.go > main
1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9     var x1, y1, x2, y2, x3, y3 float64
10    fmt.Scan(&x1, &y1)
11    fmt.Scan(&x2, &y2)
12    fmt.Scan(&x3, &y3)
13
14    ab := math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))
15    bc := math.Sqrt(math.Pow(x3-x2, 2) + math.Pow(y3-y2, 2))
16    ca := math.Sqrt(math.Pow(x3-x1, 2) + math.Pow(y3-y1, 2))
17
18    panjang := math.Max(ab, math.Max(bc, ca))
19
20    fmt.Printf("Panjang: %.2f", panjang)
21
22 }
23
```

109082500156
IF-13-02
Andromeda Alike Ramadhani

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modu14> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modu14\nomor3\soa
l3.go"
1.0 1.0
5.0 1.0
1.0 4.0
Panjang: 5.00
PS E:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modu14> go run "e:\KULIAH\ALGORITMA DAN PEMROGRAMAN\Laprak Modu14\nomor3\soa
l3.go"
2.0 0.0
1.0 1.0
1.0 4.0
Panjang: 4.12
```

Deskripsi program

Pada program diatas, kita diminta untuk membuat program untuk menentukan sisi terpanjang dari segitiga menggunakan teorema Pythagoras. Langkah untuk membuat program tersebut antara lain yaitu:

1. Deklarasikan variabel yang akan kita gunakan. Variabel yang kita gunakan yaitu variabel x1, x2, x3 dan y1, y2, y3 untuk menyimpan 3 titik dari sisi segitiga yang akan kita hitung dengan tipe data float. Tipe data float64 berfungsi untuk menyimpan data numerik desimal seperti 5.2, 1.2.
2. Gunakan "fmt.Scan" untuk menginputkan titik dari sisi segitiga yang akan kita hitung.
3. Selanjutnya masukkan rumus untuk menghitung setiap sisi segitiga, mulai dari sisi a, b, dan c dengan rumus $\sqrt{(x2 - x1)^2 + (y2 - y1)^2}$. Import "math" untuk menghitung rumus akar kuadrat dan pangkat. Gunakan math.Pow untuk pemangkatan dan math.Sqrt untuk akar kuadrat.
4. Selanjutnya gunakan rumus teorema pythagoras untuk menghitung sisi terpanjang segitiga.

5. Terakhir, gunakan `“fmt.Printf("Panjang: %.2f", Panjang)”` untuk menampilkan sisi terpanjang segitiga yang telah dihitung menggunakan rumus teorema Pythagoras. `“fmt.Printf”` berfungsi untuk menampilkan output dengan format tertentu. `%.2f` berfungsi untuk menampilkan format variabel yang digunakan, disini kita menggunakan tipe float. Oleh karena itu, kita menggunakan format `%f`. Angka 2 pada format tersebut berfungsi untuk menambahkan 2 angka di belakang koma. Terakhir, jangan lupa untuk mendeklarasikan variabel panjang yang sudah dibuat sebelumnya di akhir.