

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

Modul 3 : Fungsi



Oleh:

ANGGUN WAHYU WIDIYANA

103112480280

12-IF-01

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
TAHUN AJARAN 2025/2026**

I. DASAR TEORI

Dalam pengembangan algoritma dan pemrograman, fungsi merupakan salah satu konsep paling fundamental yang menjadi landasan bagi pemrograman modular dan terstruktur. Fungsi memungkinkan pemrogram untuk memecah masalah kompleks menjadi bagian-bagian yang lebih kecil dan lebih mudah dikelola, sehingga meningkatkan keterbacaan, pemeliharaan, dan efisiensi kode.

A. Konsep Dasar Fungsi

Fungsi adalah serangkaian instruksi yang dikelompokkan dan diberi nama, dirancang untuk melakukan tugas spesifik, dan dapat dipanggil berulang kali dari bagian lain program. Berbeda dengan prosedur, fungsi selalu mengembalikan nilai setelah dieksekusi.

Suatu subprogram dikategorikan sebagai fungsi jika memenuhi dua kriteria utama:

1. Memiliki deklarasi tipe nilai yang dikembalikan
2. Mengandung kata kunci `return` dalam badan subprogram

Fungsi biasanya digunakan dalam konteks:

1. Penugasan (assignment) nilai ke variabel
2. Sebagai bagian dari ekspresi
3. Sebagai argumen dalam pemanggilan subprogram lain

Fungsi memiliki beberapa karakteristik penting:

1. Modularitas: Memungkinkan pemisahan logika program menjadi unit-unit yang dapat dikelola
2. Reusabilitas: Dapat dipanggil berulang kali dari berbagai bagian program
3. Abstraksi: Menyembunyikan implementasi detail dari pemanggil
4. Encapsulation: Membungkus data & operasi yang bekerja pada data tsb.
5. Keteraturan: Memungkinkan organisasi kode yang logis dan terstruktur

B. Deklarasi dan Struktur Fungsi

Deklarasi fungsi memiliki struktur umum sebagai berikut:

```
function <nama_fungsi> (<parameter>) : <tipe_data>
kamus
    <deklarasi variabel lokal>
algoritma
    <badan algoritma fungsi>
    return <nilai/variabel>
endfunction
```

Tabel 1.1 Notasi Algoritma

```
```go
func <nama_fungsi> (<parameter>) <tipe_data> {
 // deklarasi variabel lokal
 // badan algoritma fungsi
 return <nilai/variabel>
}
```

*Tabel 1.2 Notasi dalam Go*

Elemen penting dalam deklarasi fungsi:

1. Nama fungsi: Identifier yang digunakan untuk memanggil fungsi
2. Parameter: Daftar variabel yang diterima fungsi sebagai input
3. Tipe data return: Tipe data nilai yang dikembalikan fungsi
4. Variabel lokal: Variabel yang hanya dikenali dalam lingkup fungsi
5. Badan fungsi: Instruksi yang dieksekusi saat fungsi dipanggil
6. Return statement: Pernyataan yang mengembalikan nilai dan mengakhiri eksekusi fungsi.

Rekursi adalah teknik dimana fungsi memanggil dirinya sendiri untuk menyelesaikan masalah. Fungsi rekursif biasanya terdiri dari:

1. Base case: Kondisi sederhana yang menghentikan rekursi
2. Recursive case: Pemanggilan fungsi itu sendiri dengan parameter yang berbeda

Berikut adalah contoh fungsi untuk menghitung volume tabung jika diketahui jari-jari alas dan tinggi:

```
```go
func volumeTabung(jari_jari, tinggi int) float64 {
    var luasAlas, volume float64
    luasAlas = 3.14 * float64(jari_jari * jari_jari)
    volume = luasAlas * float64(tinggi)
    return volume
}
```
```

### C. Pemanggilan Fungsi

Fungsi dapat dipanggil dengan menuliskan nama fungsi diikuti dengan argumen yang diperlukan dalam tanda kurung. Berbeda dengan prosedur, fungsi dapat:

1. Diassign ke variabel
2. Menjadi bagian dari ekspresi
3. Digunakan sebagai argumen untuk fungsi atau prosedur lain

```
program ContohProsedur
kamus
 r, t: integer
 v1, v2: real
algoritma
 r ← 5
 t ← 10
 v1 ← volumeTabung(r, t) // cara pemanggilan #1
 v2 ← volumeTabung(r, t) + volumeTabung(15, t) // cara
pemanggilan #2
 output(volumeTabung(14, 100)) // cara pemanggilan #3
endprogram
```

Tabel 2.1 Notasi Algoritma

```
```go
func main() {
```

<pre> var r, t int var v1, v2 float64 r = 5 t = 10 v1 = volumeTabung(r, t) // cara pemanggilan #1 v2 = volumeTabung(r, t) + volumeTabung(15, t) // cara pemanggilan #2 fmt.Println(volumeTabung(14, 100)) // cara pemanggilan #3 } ... </pre>

Tabel 2.2 Notasi dalam Go

D. Jenis-Jenis Fungsi

Fungsi tanpa Parameter	Fungsi dengan Parameter
<pre> ```go func getCurrentTime() string { return time.Now().Format("15:04:05") } </pre>	<pre> ```go func power(base, exponent int) int { result := 1 for i := 0; i < exponent; i++ { result *= base } return result } ... </pre>

Tabel 3.1 Fungsi tanpa dan dengan Parameter

<pre> ```go func greeting(name string, formal bool) string { if formal { return "Good day, " + name } return "Hey, " + name } </pre>
--

Tabel 3.2 Fungsi dengan Parameter Default

Fungsi dengan Return Tunggal	Fungsi dengan Multiple Return
<pre> ```go func isEven(number int) bool { return number%2 == 0 } ... </pre>	<pre> ```go func divideAndRemainder(a, b int) (int, int) { return a / b, a % b } ... </pre>

II. GUIDED

Contoh Soal 1

```
//Nama : Anggun Wahyu W. (103112480280)
package main

import "fmt"

func main() {
    var a, b int
    fmt.Scan(&a,&b)

    if a >= b {
        fmt.Println(permutasi(a,b))
    } else {
        fmt.Println(permutasi(b,a))
    }
}

func faktorial(n int) int {
    hasil := 1
    for i := 1; i<=n; i++ {
        hasil *= i
    }
    return hasil
}

func permutasi(n,r int) int {
    if r > n {
        return 0
    }
    return faktorial(n) / faktorial(n-r)
}
```

Output:

```
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Guided3\ContohSoal1.go"
5 3
60
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Guided3\ContohSoal1.go"
4 2
12
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Guided3\ContohSoal1.go"
6 6
720
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Guided3\ContohSoal1.go"
3 5
60
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Guided3\ContohSoal1.go"
10 4
5040
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Guided3\ContohSoal1.go"
```

Deskripsi Program

Program ini menghitung nilai permutasi dari dua bilangan bulat yang dimasukkan oleh pengguna. Pengguna diminta untuk memasukkan dua angka, a dan b. Program kemudian memeriksa mana yang lebih besar dan menghitung permutasi menggunakan fungsi permutasi(n, r), yang didefinisikan sebagai $P(n, r) = \frac{n!}{(n-r)!}$

Fungsi faktorial(n) digunakan untuk menghitung faktorial dari bilangan bulat n. Jika r lebih besar dari n, fungsi permutasi akan mengembalikan 0. Hasil perhitungan permutasi ditampilkan ke layar. Singkatnya, program ini adalah kalkulator sederhana untuk menghitung permutasi berdasarkan input pengguna.

Contoh Soal 2

```
//Nama : Anggun Wahyu W. (103112480280)
package main

import "fmt"

// Fungsi untuk mengonversi suhu dari Celsius ke Fahrenheit
func celsiusToFahrenheit(celsius float64) float64 {
    return (9.0/5.0)*celsius + 32
}

func main() {
    var N int
    fmt.Print("Masukkan jumlah data: ")
    _, err := fmt.Scan(&N)
    if err != nil || N <= 0 {
        fmt.Println("Input tidak valid, pastikan memasukkan angka positif.")
        return
    }

    temperatures := make([]float64, N)

    // Membaca suhu dalam Celsius
    fmt.Println("Masukkan suhu dalam Celsius:")
    for i := 0; i < N; i++ {
        _, err := fmt.Scan(&temperatures[i])
        if err != nil {
            fmt.Println("Input tidak valid, pastikan memasukkan angka.")
            return
        }
    }

    // Mengonversi ke Fahrenheit dan mencetak hasil
    fmt.Println("Suhu dalam Fahrenheit:")
    for _, temp := range temperatures {
        fmt.Printf("%.2f\n", celsiusToFahrenheit(temp))
    }
}
```

Output:

```
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODUL3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODUL3\103112480280_Guided3\ContohSoal2.go"
Masukkan jumlah data: 3
Masukkan suhu dalam Celsius:25
0
100
32.00
212.00
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODUL3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODUL3\103112480280_Guided3\ContohSoal2.go"
Masukkan jumlah data: 2
Masukkan suhu dalam Celsius:-10
20
Suhu dalam Fahrenheit:
14.00
68.00
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODUL3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODUL3\103112480280_Guided3\ContohSoal2.go"
Masukkan jumlah data: 4
Masukkan suhu dalam Celsius:30.5
15.2
-5.5
37.0
Suhu dalam Fahrenheit:
86.90
59.36
22.10
98.60
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODUL3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODUL3\103112480280_Guided3\ContohSoal2.go"
```

Deskripsi Program

Program ini mengonversi sejumlah suhu dari Celsius ke Fahrenheit. Pengguna diminta memasukkan jumlah data suhu yang akan dikonversi, kemudian memasukkan nilai suhu dalam Celsius. Program membaca setiap suhu, mengonversinya ke Fahrenheit menggunakan fungsi `celsiusToFahrenheit`, dan mencetak hasil konversi dengan format desimal dua angka. Jika input tidak valid, program akan menampilkan pesan kesalahan dan berhenti.

Contoh Soal 3

```
//Nama : Anggun Wahyu W. (103112480280)
package main

import (
    "fmt"
    "math"
)

// Fungsi untuk menghitung luas permukaan tabung
func luasPermukaanTabung(r, t float64) float64 {
    return 2 * math.Pi * r * (r + t)
}

// Fungsi untuk menghitung volume tabung
func volumeTabung(r, t float64) float64 {
    return math.Pi * math.Pow(r, 2) * t
}

func main() {
    var r, t float64
```

```

// Input jari-jari dan tinggi tabung dengan validasi
fmt.Print("Masukkan jari-jari tabung: ")
_, errR := fmt.Scan(&r)
fmt.Print("Masukkan tinggi tabung: ")
_, errT := fmt.Scan(&t)

// Memeriksa apakah input valid
if errR != nil || errT != nil {
    fmt.Println("Input tidak valid! Harap masukkan angka yang benar.")
    return
}
// Memeriksa apakah jari-jari dan tinggi bernilai positif
if r <= 0 || t <= 0 {
    fmt.Println("Jari-jari dan tinggi tabung harus lebih dari nol.")
    return
}
// Menghitung luas permukaan dan volume
luas := luasPermukaanTabung(r, t)
volume := volumeTabung(r, t)

// Menampilkan hasil
fmt.Println("=====")
fmt.Printf("Luas Permukaan Tabung: %.2f satuan²\n", luas)
fmt.Printf("Volume Tabung: %.2f satuan³\n", volume)
fmt.Println("=====")
}

```

Output:

```

PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_guided3\ContohSoal3.go"
Masukkan jari-jari tabung: 5
Masukkan tinggi tabung: 10
=====
Luas Permukaan Tabung: 471.24 satuan²
Volume Tabung: 785.40 satuan³
=====
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_guided3\ContohSoal3.go"
Masukkan jari-jari tabung: -4
Masukkan tinggi tabung: 8
Jari-jari dan tinggi tabung harus lebih dari nol.
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_guided3\ContohSoal3.go"
Masukkan jari-jari tabung: abc
Masukkan tinggi tabung: Input tidak valid! Harap masukkan angka yang benar.
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_guided3\ContohSoal3.go"
Masukkan jari-jari tabung: 3
Masukkan tinggi tabung: 7
=====
Luas Permukaan Tabung: 188.50 satuan²
Volume Tabung: 197.92 satuan³
=====
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_guided3\ContohSoal3.go"

```

Deskripsi Program

Program ini menghitung luas permukaan dan volume tabung berdasarkan jari-jari dan tinggi yang dimasukkan oleh pengguna. Program memvalidasi input untuk memastikan nilai yang dimasukkan berupa angka positif. Jika valid, program menggunakan fungsi `luasPermukaanTabung` dan `volumeTabung` untuk melakukan perhitungan dengan rumus matematika. Hasil perhitungan ditampilkan dalam format desimal dua angka. Jika input tidak valid, program akan menampilkan pesan kesalahan dan berhenti.

III. UNGUIDED

Latihan Soal 1

Minggu ini, mahasiswa Fakultas Informatika mendapatkan tugas dari mata kuliah matematika diskrit untuk mempelajari kombinasi dan permutasi. Jonas salah seorang mahasiswa, iseng untuk mengimplementasikannya ke dalam suatu program. Oleh karena itu bersediakah kalian membantu Jonas?

Masukan terdiri dari empat buah bilangan asli a, b, c , dan d yang dipisahkan oleh spasi, dengan syarat $a \geq c$ dan $b \geq d$.

Keluaran terdiri dari dua baris. Baris pertama adalah hasil permutasi dan kombinasi a terhadap c , sedangkan baris kedua adalah hasil permutasi dan kombinasi b terhadap d .

Catatan: permutasi (P) dan kombinasi (C) dari n terhadap r ($n \geq r$) dapat dihitung dengan menggunakan persamaan berikut!

$$P(n, r) = \frac{n!}{(n-r)!}, \text{ sedangkan } C(n, r) = \frac{n!}{r!(n-r)!}$$

(Contoh input/output, Teks bergaris bawah adalah input dari user):

No	Masukan	Keluaran	Penjelasan
1.	5 10 3 10	60 10 3628800 1	$P(5, 3) = 5!/2! = 120/2 = 60$ $C(5, 3) = 5!/(3! \times 2!) = 120/12 = 10$ $P(10, 10) = 10!/0! = 3628800/1 = 3628800$ $C(10, 10) = 10!/(10! \times 0!) = 10!/10! = 1$
2.	8 0 2 0	56 28 1 1	

Selesaikan program tersebut dengan memanfaatkan subprogram yang diberikan berikut ini!

```
function factorial(n: integer) à integer
{mengembalikan nilai faktorial dari n}

function permutation(n,r : integer) à integer
{Mengembalikan hasil n permutasi r, dan n >= r}

function combination(n,r : integer) à integer
{Mengembalikan hasil n kombinasi r, dan n >= r}
```

```
// Nama : Anggun Wahyu W. (103112480280)
package main

import "fmt"

func factorial(n int) int {
    if n == 0 {
        return 1
    }
}
```

```

    return n * factorial(n-1)
}

func permutation(n, r int) int {
    if n < r {
        return 0
    }
    return factorial(n) / factorial(n-r)
}

func combination(n, r int) int {
    if n < r {
        return 0
    }
    return factorial(n) / (factorial(r) * factorial(n-r))
}

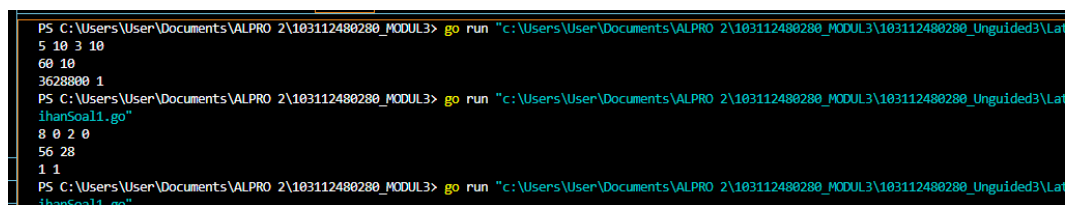
func main() {
    var a, b, c, d int
    fmt.Scan(&a, &b, &c, &d)

    permutasiA := permutation(a, c)
    kombinasiA := combination(a, c)
    permutasiB := permutation(b, d)
    kombinasiB := combination(b, d)

    fmt.Printf("%d %d\n", permutasiA, kombinasiA)
    fmt.Printf("%d %d\n", permutasiB, kombinasiB)
}

```

Output:



```

PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Unguided3\LatihanSoal1.go"
5 10 3 10
60 10
3628800 1
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Unguided3\LatihanSoal1.go"
8 0 2 0
56 28
1 1
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Unguided3\LatihanSoal1.go"

```

Deskripsi Program

Program ini berfungsi untuk membantu Jonas menghitung permutasi dan kombinasi dari dua pasang bilangan, sesuai dengan rumus matematika diskrit. Program menerima empat input angka: a, b, c, dan d. Kemudian, program menghitung permutasi dan kombinasi a terhadap c, serta permutasi dan kombinasi b terhadap d, menggunakan fungsi factorial, permutation, dan combination. Hasil perhitungan tersebut kemudian ditampilkan ke layar, masing-masing dalam dua baris terpisah, mengikuti format yang ditentukan oleh soal. Singkatnya, program ini adalah kalkulator sederhana untuk permutasi dan kombinasi.

Latihan Soal 2

Diberikan tiga buah fungsi matematika yaitu $f(x) = x^2$, $g(x) = x - 2$ dan $h(x) = x + 1$. Fungsi komposisi $(fogoh)(x)$ artinya adalah $f(g(h(x)))$. Tuliskan $f(x)$, $g(x)$ dan $h(x)$ dalam bentuk function.

Masukan terdiri dari sebuah bilangan bulat a, b dan c yang dipisahkan oleh spasi.

Keluaran terdiri dari tiga baris. Baris pertama adalah $(fogoh)(a)$, baris kedua $(gohof)(b)$, dan baris ketiga adalah $(hofog)(c)$!

Contoh

No	Masukan	Keluaran	Penjelasan
1.	7 2 10	36 3 65	$(fogog)(7) = 36$ $(gohof)(2) = 3$ $(hofog)(10) = 65$
2.	5 5 5	16 24 10	$(fogog)(5) = 16$ $(gohof)(5) = 24$ $(hofog)(5) = 10$
3.	3 8 4	4 63 5	$(fogog)(3) = 4$ $(gohof)(8) = 63$ $(hofog)(4) = 5$

```
//Nama : Anggun Wahyu W. (103112480280)
package main

import "fmt"

func f(x int) int {
    return x * x
}

func g(x int) int {
    return x - 2
}

func h(x int) int {
    return x + 1
}

func fogoh(x int) int {
    return f(g(h(x)))
}

func gohof(x int) int {
    return g(h(f(x)))
}

func hofag(x int) int {
```

```

    return h(f(g(x)))
}

func main() {
    var a, b, c int
    fmt.Scan(&a, &b, &c)

    resultFogoh := fogoh(a)
    resultGohof := gohof(b)
    resultHofag := hofag(c)

    fmt.Println(resultFogoh)
    fmt.Println(resultGohof)
    fmt.Println(resultHofag)
}

```

Output:

```

PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Unguided3\Lat
7 2 10
36
3
65
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Unguided3\Lat
5 5 5
16
24
10
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Unguided3\Lat
3 8 4
63
5
PS C:\Users\User\Documents\ALPRO 2\103112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\103112480280_MODULE3\103112480280_Unguided3\Lat

```

Deskripsi Program

Program ini adalah kalkulator fungsi komposisi. Pengguna memberikan tiga angka sebagai input (a, b, c). Program kemudian menghitung dan menampilkan tiga hasil: $f(g(h(a)))$, $g(h(f(b)))$, dan $h(f(g(c)))$ berdasarkan definisi fungsi $f(x) = x^2$, $g(x) = x - 2$, dan $h(x) = x + 1$. Hasil perhitungan ditampilkan dalam tiga baris terpisah.

Latihan Soal 3

[Lingkaran] Suatu lingkaran didefinisikan dengan koordinat titik pusat (cx, cy) dengan radius r . Apabila diberikan dua buah lingkaran, maka tentukan posisi sebuah titik sembarang (x, y) berdasarkan dua lingkaran tersebut.

Masukan terdiri dari beberapa tiga baris. Baris pertama dan kedua adalah koordinat titik pusat dan radius dari lingkaran 1 dan lingkaran 2, sedangkan baris ketiga adalah koordinat titik sembarang. Asumsi sumbu x dan y dari semua titik dan juga radius direpresentasikan dengan bilangan bulat.

Keluaran berupa string yang menyatakan posisi titik "Titik di dalam lingkaran 1 dan 2", "Titik di dalam lingkaran 1", "Titik di dalam lingkaran 2", atau "Titik

di luar lingkaran 1 dan 2".

Contoh

No	Masukan	Keluaran
1.	1 1 5 8 8 4 2 2	Titik di dalam lingkaran 1
2.	1 2 3 4 5 6 7 8	Titik di dalam lingkaran 2
3.	5 10 15 -15 4 20 0 0	Titik di dalam lingkaran 1 dan 2
4.	1 1 5 8 8 4 15 20	Titik di luar lingkaran 1 dan 2

Fungsi untuk menghitung jarak titik (a, b) dan (c, d) dimana rumus jarak adalah:

$$jarak = \sqrt{(a - c)^2 + (b - d)^2}$$

dan juga fungsi untuk menentukan posisi sebuah titik sembarang berada di dalam suatu lingkaran atau tidak.

```
function jarak(a,b,c,d : real) -> real
{Mengembalikan jarak antara titik (a,b) dan titik (c,d)}

function didalam(cx,cy,r,x,y : real) -> boolean
{Mengembalikan true apabila titik (x,y) berada di dalam lingkaran
yang memiliki titik pusat (cx,cy) dan radius r}
```

```
//Nama : Anggun Wahyu W. (103112480280)
package main

import (
    "fmt"
    "math"
)

func jarak(a, b, c, d float64) float64 {
    return math.Sqrt(math.Pow(a-c, 2) + math.Pow(b-d, 2))
}

func didalam(cx, cy, r, x, y float64) bool {
    return jarak(x, y, cx, cy) < r
}

func main() {
    var cx1, cy1, r1, cx2, cy2, r2, x, y int
    fmt.Scan(&cx1, &cy1, &r1)
    fmt.Scan(&cx2, &cy2, &r2)
    fmt.Scan(&x, &y)
```

```

fCx1, fCy1, fR1 := float64(cx1), float64(cy1), float64(r1)
fCx2, fCy2, fR2 := float64(cx2), float64(cy2), float64(r2)
fX, fY := float64(x), float64(y)

dalamLingkaran1 := didalam(fCx1, fCy1, fR1, fX, fY)
dalamLingkaran2 := didalam(fCx2, fCy2, fR2, fX, fY)

if dalamLingkaran1 && dalamLingkaran2 {
    fmt.Println("Titik di dalam lingkaran 1 dan 2")
} else if dalamLingkaran1 {
    fmt.Println("Titik di dalam lingkaran 1")
} else if dalamLingkaran2 {
    fmt.Println("Titik di dalam lingkaran 2")
} else {
    fmt.Println("Titik di luar lingkaran 1 dan 2")
}
}

```

Output:

```

PS C:\Users\User\Documents\ALPRO 2\183112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\183112480280_MODULE3\183112480280_Unguided3\LatihanSoal3.go"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS C:\Users\User\Documents\ALPRO 2\183112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\183112480280_MODULE3\183112480280_Unguided3\LatihanSoal3.go"
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS C:\Users\User\Documents\ALPRO 2\183112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\183112480280_MODULE3\183112480280_Unguided3\LatihanSoal3.go"
5 10 15
-15 4 20
0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\User\Documents\ALPRO 2\183112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\183112480280_MODULE3\183112480280_Unguided3\LatihanSoal3.go"
1 1 5
8 8 4
15 20
Titik di luar lingkaran 1 dan 2
PS C:\Users\User\Documents\ALPRO 2\183112480280_MODULE3> go run "c:\Users\User\Documents\ALPRO 2\183112480280_MODULE3\183112480280_Unguided3\LatihanSoal3.go"

```

Deskripsi Program

Program ini menentukan posisi suatu titik terhadap dua lingkaran. Pengguna memberikan koordinat pusat dan radius untuk masing-masing lingkaran, serta koordinat titik yang akan diuji. Program menghitung jarak antara titik tersebut dengan pusat setiap lingkaran. Berdasarkan jarak ini, program menentukan apakah titik berada di dalam lingkaran pertama, lingkaran kedua, keduanya, atau di luar kedua lingkaran, lalu mencetak hasilnya ke layar. Singkatnya, program ini adalah alat untuk memposisikan titik relatif terhadap dua lingkaran.

IV. KESIMPULAN

Fungsi adalah komponen penting dalam pengembangan algoritma dan pemrograman yang menyediakan mekanisme untuk modularitas, reusabilitas, dan abstraksi. Fungsi memungkinkan pemrogram untuk membagi kode menjadi bagian-bagian yang lebih kecil dan terorganisir, meningkatkan keterbacaan dan pemeliharaan program. Melalui contoh-contoh yang telah diberikan diatas, seperti perhitungan luas permukaan dan volume tabung, serta konversi suhu, terlihat bagaimana fungsi dapat digunakan untuk melakukan perhitungan matematis secara efisien. Fungsi juga mendukung penggunaan kembali kode (code reuse) dan modularitas, memungkinkan untuk mengimplementasikan logika yang kompleks dengan cara yang lebih sederhana. Selain itu, validasi input dalam fungsi memastikan bahwa program dapat menangani kesalahan dengan baik, memberikan pengalaman pengguna yang lebih baik.

V. REFERENSI

School of Computing. *Modul Praktikum 3 – Fungsi Algoritma dan Pemrograman 2 S1 Informatika*. 2024

Munir, Rinaldi. (2020). *Algoritma dan Pemrograman dalam Bahasa Go*. Bandung: Informatika.

Utami, Ema dan Sukrisno. (2019). *Konsep Dasar Pemrograman Go*. Yogyakarta: Graha Ilmu.