

**LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK**

**MODUL 3
FUNCTION**



Oleh:

M. AZKA HERMAWAN

2311102230

IF – 11- 02

**S1 TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

2024

I. DASAR TEORI

1. Definisi Function

Fungsi merupakan satu kesatuan rangkaian instruksi yang memberikan atau menghasilkan suatu nilai dan biasanya memetakan input ke suatu nilai yang lain. Oleh karena itu, fungsi selalu menghasilkan/mengembalikan nilai. Suatu subprogram dikatakan fungsi apabila:

- a. Ada deklarasi tipe nilai yang dikembalikan, dan
- b. Terdapat kata kunci return dalam badan subprogram.

Maka fungsi digunakan jika suatu nilai biasanya diperlukan, seperti

- Assignment nilai ke suatu variable
- Bagian dari ekspresi
- Bagian dari argumen suatu subprogram, dsb.

Karena itu selalu pilih nama fungsi yang menggambarkan nilai, seperti kata benda dan kata sifat. Contoh nama-nama fungsi: median, rerata, nilaiTerbesar, ketemu, selesai.

2. Deklarasi Function

Deklarasi fungsi sama dengan prosedur yaitu berada pada blok yang terpisah dengan program utama.

	Notasi Algoritma
1	function <nama function> (<params>) -> <type>
2	kamus
3	{deklarasi variabel lokal dari fungsi}
4	...
5	algoritma
6	{badan algoritma fungsi}
7	...
8	return <value/variabel>
9	endfunction
	Notasi dalam bahasa Go
10	func <nama function> (<params>) <type> {
11	/* deklarasi variabel lokal dari fungsi */
12	...
13	/* badan algoritma fungsi*/
14	...
15	return <value/variabel>
16	}
17	

Pada bagian deklarasi terlihat setelah parameter terdapat tipe data dari nilai yang dikembalikan, sedangkan pada bagian badan fungsi terdapat return dari nilai yang dikembalikan.

Berikut adalah contoh fungsi untuk menghitung volume dari tabung apabila jari-jari alas dan tinggi tabung diketahui.

	Notasi Algoritma
1	function volumeTabung(jari_jari,tinggi : integer) -> real
2	kamus
3	luasAlas, volume: real
4	algoritma
5	luasAlas <- 3.14 * (jari_jari * jari_jari)
6	volume <- luasAlas * tinggi
7	return volume
8	endfunction
	Notasi dalam bahasa Go
10	func volumeTabung(jari_jari,tinggi int) float64 {
11	var luasAlas,volume float64
12	luasAlas = 3.14 * float64(jari_jari * jari_jari)
13	volume = luasAlas * tinggi
14	return volume
15	}

3. Cara Pemanggilan Function

Sama halnya dengan prosedur cukup dilakukan dengan penulisan nama fungsi beserta argumen yang diminta oleh parameter dari fungsi. Perbedaannya dengan R adalah fungsi bisa di-assign ke suatu

variabel, menjadi bagian dari ekspresi, dan argumen dari suatu subprogram.

Notasi Algoritma	
1	program ContohProsedur
2	kamus
3	r, t : integer
4	v1, v2 : real
5	algoritma
6	r <- 5;
7	t <- 10
8	v1 <- volumeTabung(r, t) {cara pemanggilan #1}
9	v2 <- volumeTabung(r, t) + volumeTabung(15, t) {cara pemanggilan #2}
10	output(volumeTabung(14, 100)) {cara pemanggilan #3}
11	endprogram
Notasi dalam bahasa Go	
12	func main() {
13	var r, t int
14	var v1, v2 float64
15	r = 5
16	t = 10
17	v1 = volumeTabung(r, t) // cara pemanggilan #1
18	v2 = volumeTabung(r, t) + volumeTabung(15, t) // cara pemanggilan #2
19	fmt.Println(volumeTabung(14, 100)) // cara pemanggilan #3
20	}

Pada contoh pemanggilan fungsi di atas terlihat tidak ada perbedaan pada saat pemanggilan fungsi pada pseudocode ataupun GoLang. Di sini terlihat fungsi bisa di-assign ke suatu variabel pada saat pemanggilan, bisa dioperasikan sesuai dengan tipe data yang dikembalikan, dan juga bisa langsung ditampilkan dengan perintah output ataupun print.

II. GUIDED

1. Guided 1

Source Code

```
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 {
    var hasil int = 1
    var i int
    for i = 1; i <= n; i++ {
        hasil = hasil * i
    }
    return hasil
}

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

Screenshot



5 4
120

Deskripsi

Program diatas merupakan program untuk menghitung permutasi 2 angka yang diinputkan oleh pengguna. Program diatas menggunakan fungsi factorial dan fungsi permutasi yang bertujuan untuk menghitung dari faktorial dan rumus dari sebuah permutasi.

III. UNGUIDED

1. Unguided 1

Source Code

```
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(cx, cy, x, y) <= r
}

func main() {
    var cx1, cy1, r1, cx2, cy2, r2, x, y float64

    fmt.Print("Lingkaran 1: ")
    fmt.Scan(&cx1, &cy1, &r1)

    fmt.Print("Lingkaran 2: ")
    fmt.Scan(&cx2, &cy2, &r2)

    fmt.Print("Titik sembarang: ")
    fmt.Scan(&x, &y)

    insideLingkaran1 := diDalam(cx1, cy1, r1, x, y)
    insideLingkaran2 := diDalam(cx2, cy2, r2, x, y)
    if insideLingkaran1 && insideLingkaran2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if insideLingkaran1 {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if insideLingkaran2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}
```

Screenshot

```
Lingkaran 1: 1 1 5  
Lingkaran 2: 8 8 4  
Titik sembarang: 2 2  
Titik di dalam lingkaran 1
```

Deskripsi

Program berikut merupakan program untuk menentukan sebuah titik sembarang yang berada didalam dua lingkaran yang telah diberikan. Didalam program ini mempunyai beberapa fungsi, diantaranya adalah: *func jarak*: berfungsi untuk menghitung jarak antara 2 titik.

Func dalamLingkaran: berfungsi untuk menentukan apakah (x, y) berada didalam atau tidak, caranya dengan membandingkan jarak dari titik ke pusat lingkaran dengan radius lingkaran. Jika jarak tersebut lebih kecil atau sama dengan radius, titik tersebut berada di dalam lingkaran.