

**LAPORAN PRAKTIKUM
ALGORITMA PEMROGRAMAN 2**

**MODUL 3
PENGENALAN FUNGSI**



Oleh:

PANDIA ARYA BRATA

2311102076

IF – 11 - 02

**S1 TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

2024

I. DASAR TEORI

3.1 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:

1. **Ada** deklarasi tipe nilai yang dikembalikan, dan
2. **Terdapat** kata kunci **return** dalam badan subprogram.

Maka fungsi digunakan jika suatu nilai biasanya diperlukan, seperti:

- Assignment nilai ke suatu variabel
- 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**, ...

3.2 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
4	... dari
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) 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) float64 {
11	var luasAlas,volume float64
12	luasAlas = 3.14 * float64(jari_jari * jari_jari)
13	volume = luasAlas * tinggi
14	return volume
15	}

3.3 Cara Pemanggilan Function

Sama halnya dengan prosedur, pemanggilan fungsi cukup dilakukan dengan penulisan nama fungsi beserta argumen yang diminta oleh parameter dari fungsi. Perbedaannya dengan prosedur 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	<code>v2 = volumetabung(r,t) + volumetabung(15,t)</code>	<code>// cara pemanggilan #2</code>
19	<code>fmt.Println(volumeTabung(14,100))</code>	<code>// cara pemanggilan</code>
20	<code>}</code>	

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.

3.4 Contoh Program dengan Function

Berikut ini adalah contoh penulisan fungsi pada suatu program lengkap.

Buatlah sebuah program beserta fungsi yang digunakan untuk menghitung nilai faktorial dan permutasi.

Masukan terdiri dari dua buah bilangan positif a dan b .

Keluaran berupa sebuah bilangan bulat yang menyatakan nilai a permutasi b apabila $a \geq b$ atau b permutasi a untuk kemungkinan yang lain.

```

1  package main
2  import "fmt"
3  func main(){
4      var a,b int
5      fmt.Scan(&a, &b)
6      if a >= b {
7          fmt.Println(permutasi(a,b))
8      }else{
9          fmt.Println(permutasi(b,a))
10     }
11 }
12 func faktorial(n int) int{
13     var hasil int = 1
14     var i int
15     for i = 1; i <= n; i++ {
16         hasil = hasil * i
17     }
18     return hasil
19 }
20 func permutasi(n,r int) int {
21     return faktorial(n) / faktorial(n-r)
22 }
```

D:\DEV\DEMO>go build contoh.go

D:\DEV\DEMO>contoh.exe

2 5
20

Pada contoh di atas fungsi faktorial dipanggil secara tidak langsung melalui fungsi permutasi, dan fungsi faktorial dan permutasi dipanggil sebagai ekspresi dari suatu statement.

II. GUIDED

Sourcecode :

```
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 program :

```
5 5
120
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul3>
```

Deskripsi program :

Program ini secara keseluruhan dirancang untuk menghitung permutasi dari dua bilangan bulat yang diinputkan oleh pengguna. Fungsi faktorial digunakan sebagai fungsi pendukung untuk menghitung faktorial, yang merupakan bagian dari rumus permutasi. Fungsi permutasi kemudian menggunakan hasil dari fungsi faktorial untuk menghitung nilai permutasi.

III. UNGUIDED

Sourcecode :

```
package main

import (
    "fmt"
    "math"
)

func priksa_titik(cx1, cx2, r1, cy1, cy2, r2, x, y float64) string
{
    titik1 := math.Sqrt(math.Pow(x-cx1, 2) + math.Pow(y-cy1, 2))
    titik2 := math.Sqrt(math.Pow(x-cx2, 2) + math.Pow(y-cy2, 2))

    if titik1 <= r1 && titik2 <= r2 {
        return ("Titik di dalam lingkaran 1 dan 2")
    } else if titik1 <= r1 {
        return ("Titik di dalam lingkaran 1")
    } else if titik2 <= r2 {
        return ("Titik di dalam lingkaran 2")
    } else {
        return ("Titik di luar lingkaran 1 dan 2")
    }
}

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

    fmt.Print("Masukan kordinat titik pusat dan radius lingkaran : ")
    fmt.Scan(&cx1, &cy1, &r1)

    fmt.Print("Masukan kordinat titik pusat dan radius lingkaran : ")
    fmt.Scan(&cx2, &cy2, &r2)

    fmt.Print("Masukan kordinat titik x dan y : ")
    fmt.Scan(&x, &y)

    hasil := priksa_titik(x, y, cx1, cy1, r1, cx2, cy2, r2)
    fmt.Print(hasil)
}
```

Screenshot program :

```
Masukan kordinat titik pusat dan radius lingkaran : 1 2 3
Masukan kordinat titik pusat dan radius lingkaran : 3 2 1
Masukan kordinat titik x dan y : 4 4
Titik di dalam lingkaran 2
```

Deskripsi program :

Program ini melakukan perhitungan matematis untuk menentukan jarak antara titik yang diberikan dengan pusat masing-masing lingkaran. Kemudian, hasil perhitungan jarak tersebut dibandingkan dengan jari-jari lingkaran untuk menentukan apakah titik tersebut berada di dalam, di luar, atau tepat pada lingkaran.