

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL III
FUNGSI**



Disusun Oleh :

Deshan Rafif Alfarisi / 2311102326

S1-IF-11-06

Dosen Pengampu :

Abednego Dwi Septiadi S.Kom., M.Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

a. DefintstFunctton

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 digunakanjika 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, nllalTerbesar, ketemu, selesai, ...

b. Deklarasi Functlon

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

Notas1 Algoritma	
1	function nama function» (<params>) -> <type>
2	kamus
3	(deklarasi variabel lokal dan 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> (/* deklarasi variabel lokal dari fungsi */ /* badan algoritma fungsi*/

	return <value/variabel>
)

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) volume <- luasAlas * tinggi
6	return volume endfunction
7	
8	
	Notasi dalam bahasa Go
10	func volumeTabung(jari_jari,tinggi int) float64 { var luasAlas,volume
U	float64
72	luasAlas = 3.14 * float64(jari_jari ^ jari_jari) volume = luasAlas * tinggi
13	return volume
14	}
15	

3.1 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 penanggilan #1)
9	v2 <- volumeTabung(r, t) + volumeTabung(15, t) (cara penanggilan #2)
10	output(volumeTabung(14, 100)) (cara penanggilan #3)

11	endprogram	
Notasi dalam bahasa Go		
12	func main() {	
13	var r, t int	
14	var v1, v2 float64	
15	5	
16	t 10	
17	v1 volumeTabung(r, t)	// cara penanggilan #1
18	v2 = volumeTabung(r, t) + volumeTabung(15, t)	fmt // cara penanggilan #2
19	.Println(volumeTabung(14, 100))	// cara penanggilan #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.

3.1 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 permutasi b apabila $1 \leq b \leq a$ atau b permutasi untuk kemungkinan yang lain.



Fakultas Informatika
School of Computing
Telkom University



```

1 package main
2
3 import "fmt"
4 func main(){
5
6     vara,bint
7
8     fmt.Scan(&a, &b)
9     if a >= b {
10
11         fmt.Println(permutasi(a,b))
12
13     }else{
14
15         fmt.Println(permutasi(b,a))
16     }
17
18 }
19
20 func faktorial(n int) int{
21
22     var hasil int = 7

```

```

D:\DEV\DEMO>go build
contoh.go
D:\DEV\DEMO>contoh.exe
25
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

1. Soal Studi Case

Permutasi

Sourcecode

```
package main

import "fmt"

func main() {
    var a, b int
    fmt.Print("Masukkan nilai a dan b: ")
    fmt.Scan(&a, &b)

    // Memastikan a >= b
    if a >= b {
        fmt.Println("Hasil permutasi:", permutasi(a, b))
    } else {
        fmt.Println("Hasil permutasi:", permutasi(b, a))
    }
}

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

func permutasi(n, r int) int {
    if n < r {
        fmt.Println("Error: n harus lebih besar atau sama dengan r")
        return -1
    }
    return faktorial(n) / faktorial(n-r)
}
```

Screenshoot Output



Masukkan nilai a dan b: 2 3
Hasil permutasi: 6
|

Deskripsi Program

Program ini dirancang untuk menghitung permutasi dari dua bilangan bulat yang dimasukkan pengguna. Pertama, program meminta pengguna untuk memasukkan dua bilangan bulat, yaitu a dan b. Selanjutnya, program akan memastikan bahwa nilai a lebih besar atau sama dengan b. Setelah itu, program akan memanggil fungsi permutasi untuk menghitung permutasi dari nilai a dan b. Fungsi permutasi menggunakan fungsi faktorial untuk menghitung faktorial dari suatu bilangan. Hasil perhitungan permutasi kemudian akan ditampilkan ke layar. Jika nilai a lebih kecil dari b, maka nilai a dan b akan ditukar posisinya sebelum perhitungan permutasi dilakukan.

2. Soal Studi Case

Luas permukaan balok dan volume balok

Sourcecode

```
package main

import "fmt"

func main() {
    var panjang, lebar, tinggi float64

    fmt.Print("Masukkan panjang balok: ")
    fmt.Scan(&panjang)

    fmt.Print("Masukkan lebar balok: ")
    fmt.Scan(&lebar)

    fmt.Print("Masukkan tinggi balok: ")
    fmt.Scan(&tinggi)

    // Hitung luas permukaan
    luasPermukaan := 2 * ((panjang * lebar) + (panjang * tinggi) + (lebar * tinggi))

    // Hitung volume
    volume := panjang * lebar * tinggi

    fmt.Printf("Luas permukaan balok: %.2f satuan luas\n", luasPermukaan)
    fmt.Printf("Volume balok: %.2f satuan volume\n", volume)
}
```

Screenshoot Output

```
Masukkan panjang balok: 2
Masukkan lebar balok: 3
Masukkan tinggi balok: 4
Luas permukaan balok: 52.00 satuan luas
Volume balok: 24.00 satuan volume
|
```

Deskripsi Program

Program ini dirancang untuk menghitung luas permukaan dan volume sebuah balok. Pengguna akan diminta memasukkan nilai panjang, lebar, dan tinggi balok. Setelah itu, program akan menghitung luas permukaan balok dengan menjumlahkan luas semua sisinya dan menghitung volume balok dengan mengalikan panjang, lebar, dan tingginya. Hasil perhitungan luas permukaan dan volume kemudian akan ditampilkan di layar. Program ini sangat berguna bagi siapa saja yang ingin mengetahui luas permukaan dan volume sebuah balok dengan cepat dan mudah.

III. UNGUIDED

1. Soal Studi Case

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? (tidak tentunya ya :p)

Masukan terdiri dari empat buah bilangan asli a , b , c , dan d yang dipisahkan oleh spasi, dengan syarat $a \geq 1$ dan $b \geq 1$.

Keluaran terdiri dari dua baris. Baris pertama adalah hasil permutasi dan kombinasi 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!

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

Sourcecode

```
package main

import "fmt"

var a,b,c,d int

// Fungsi untuk menghitung faktorial dari bilangan n
func faktorial(n int) int {
    hasil := 1
    // Loop untuk menghitung faktorial, dimulai dari 1 hingga n
    for i := 1; i <= n; i++ {
        hasil = hasil * i
    }
    return hasil
}

// Fungsi untuk menghitung permutasi  $P(n, r) = n! / (n-r)!$ 
func permutasi(n, r int) int {
    return faktorial(n) / faktorial(n-r)
}

// Fungsi untuk menghitung kombinasi  $C(n, r) = n! / (r! * (n-r)!)$ 
func kombinasi(n, r int) int {
    return faktorial(n) / (faktorial(r) * faktorial(n-r))
}

func main() {

    fmt.Print("Masukkan input = ")
    fmt.Scan(&a,&b,&c,&d)

    if a >= c && b >= d {
```

```

// Baris pertama: Permutasi dan Kombinasi a terhadap c
fmt.Printf("%d, %d\n", permutasi(a, c), kombinasi(a, c))

// Baris kedua: Permutasi dan Kombinasi b terhadap d
fmt.Printf("%d, %d\n", permutasi(b, d), kombinasi(b, d))
} else {
    fmt.Println("Syarat tidak terpenuhi: a harus >= c dan b harus >= d")
}
}

```

Screenshoot Output

```

Masukkan input = 3 4 1 2
3, 3
12, 6

```

Deskripsi Program

Program ini untuk menghitung permutasi dan kombinasi dari dua pasang bilangan bulat yang diberikan pengguna. Pertama, pengguna diminta memasukkan empat bilangan bulat, yaitu a, b, c, dan d. Program kemudian memeriksa apakah a lebih besar sama dengan c dan b lebih besar sama dengan d. Jika syarat ini terpenuhi, maka program akan menghitung permutasi dan kombinasi dari a terhadap c serta permutasi dan kombinasi dari b terhadap d. Hasil perhitungan kemudian akan ditampilkan. Fungsi faktorial digunakan untuk menghitung faktorial dari suatu bilangan, yang kemudian digunakan dalam fungsi permutasi dan kombinasi untuk menghitung nilai permutasi dan kombinasi. Jika syarat awal tidak terpenuhi, program akan menampilkan pesan kesalahan.

2. Soal Studi Case

Diberikan tiga buah fungsi matematika yaitu $f(x) = x^2$, $g(x) = x - 2$ dan $h(x) = x + 1$. Fungsi komposisi $(f \circ g \circ h)(x)$ 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 $(f \circ g \circ h)(a)$, baris kedua $(g \circ h \circ f)(b)$, dan baris ketiga adalah $(h \circ f \circ g)(c)$!

Sourcecode

```

package main

import (
    "fmt"
)

// Fungsi-fungsi yang diberikan
func f(x int) int {

```

```

        return x * x
    }

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

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

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

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

    // Fungsi komposisi (hofog)
    func hofog(x int) int {
        return h(f(g(x)))
    }

    func main() {
        var a, b, c int

        // Input tiga bilangan bulat a, b, c
        fmt.Println("Masukkan tiga bilangan bulat (a b c) yang dipisahkan oleh spasi: ")

        fmt.Scanf("%d %d %d", &a, &b, &c)

        // Output hasil komposisi fungsi
        fmt.Println("Hasil (fogoh)(a):", fogoh(a))
        fmt.Println("Hasil (gohof)(b):", gohof(b))
        fmt.Println("Hasil (hofog)(c):", hofog(c))
    }
}

```

Screenshoot Output

```

Masukkan tiga bilangan bulat (a b c) yang dipisahkan oleh spasi:
2 3 4
Hasil (fogoh)(a): 1
Hasil (gohof)(b): 8
Hasil (hofog)(c): 5

```

Deskripsi Program

Program ini dirancang untuk menghitung komposisi dari tiga fungsi matematika yaitu $f(x)$, $g(x)$, dan $h(x)$. Setelah pengguna memasukkan tiga bilangan bulat a , b , dan c , program akan menghitung dan menampilkan hasil dari komposisi fungsi $(f \circ g \circ h)(a)$, $(g \circ h \circ f)(b)$, dan $(h \circ f \circ g)(c)$. Fungsi $f(x)$ mengkuadratkan nilai x , $g(x)$ mengurangi nilai x dengan 2, dan $h(x)$ menambahkan 1 ke nilai x . Komposisi fungsi ini melibatkan penerapan satu fungsi ke hasil fungsi lainnya secara berurutan.

3. Soal Studi Case

[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".

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

// Fungsi untuk menghitung jarak antara dua titik (x1, y1) dan (x2, y2)
func distance(x1, y1, x2, y2 int) float64 {
    return math.Sqrt(math.Pow(float64(x2-x1), 2) + math.Pow(float64(y2-y1), 2))
}

// Fungsi untuk mengecek apakah titik (x, y) berada di dalam lingkaran dengan pusat
// (cx, cy) dan radius r
func isInsideCircle(x, y, cx, cy, r int) bool {
    return distance(x, y, cx, cy) <= float64(r)
}

func main() {
    var cx1, cy1, r1 int // Koordinat dan radius lingkaran 1
    var cx2, cy2, r2 int // Koordinat dan radius lingkaran 2
    var x, y int         // Koordinat titik sembarang
```

```

// Input untuk lingkaran 1
fmt.Println("Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1):")
fmt.Scanf("%d %d %d", &cx1, &cy1, &r1)

// Input untuk lingkaran 2
fmt.Println("Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2):")
fmt.Scanf("%d %d %d", &cx2, &cy2, &r2)

// Input untuk titik sembarang
fmt.Println("Masukkan koordinat titik sembarang (x y):")
fmt.Scanf("%d %d", &x, &y)

// Mengecek apakah titik (x, y) berada di dalam lingkaran 1 dan/atau
lingkaran 2
inCircle1 := isInsideCircle(x, y, cx1, cy1, r1)
inCircle2 := isInsideCircle(x, y, cx2, cy2, r2)

// Menentukan output berdasarkan posisi titik terhadap lingkaran 1 dan
lingkaran 2
if inCircle1 && inCircle2 {
    fmt.Println("Titik di dalam lingkaran 1 dan 2")
} else if inCircle1 {
    fmt.Println("Titik di dalam lingkaran 1")
} else if inCircle2 {
    fmt.Println("Titik di dalam lingkaran 2")
} else {
    fmt.Println("Titik di luar lingkaran 1 dan 2")
}
}

```

Screenshoot Output

```

Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1):
1 2 3
Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2):
4 5 6
Masukkan koordinat titik sembarang (x y):
7 8
Titik di dalam lingkaran 2

```

Deskripsi Program

Program Go ini dirancang untuk menentukan posisi suatu titik terhadap dua lingkaran yang berbeda. Pertama, program akan meminta pengguna untuk memasukkan koordinat pusat dan jari-jari dari kedua lingkaran, serta koordinat dari sebuah titik sembarang. Kemudian, program akan menghitung jarak antara titik tersebut dengan pusat masing-masing lingkaran. Jika jarak antara titik dengan pusat lingkaran lebih kecil atau sama dengan jari-jari lingkaran, maka titik tersebut berada di dalam lingkaran. Terakhir, program akan mencetak hasil apakah titik tersebut berada di dalam lingkaran pertama, lingkaran kedua, keduanya, atau tidak satupun.

