

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL III  
FUNGSI**



**Disusun Oleh :**

**Liya Khoirunnisa / 2311102124**

**IF-11-05**

**Dosen Pengampu :**

**Arif Amrulloh, S.Kom., M.Kom**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## I. DASAR TEORI

### a. Fungsi

Fungsi adalah sekumpulan blok kode yang dibungkus dengan nama tertentu. Penerapan fungsi yang tepat akan menjadikan kode lebih modular dan juga dry (kependekan dari don't repeat yourself), karena tak perlu menuliskan banyak proses berkali-kali, cukup sekali saja dan tinggal panggil jika dibutuhkan.

### b. Penerapan fungsi

Cara membuat fungsi cukup mudah, yaitu dengan menuliskan keyword `func`, diikuti setelahnya nama fungsi, kurung yang berisikan parameter, dan kurung kurawal untuk membungkus blok kode. Parameter sendiri adalah variabel yang disisipkan pada saat pemanggilan fungsi. Sebuah fungsi bisa didesain tidak mengembalikan apa-apa (`void`), atau bisa mengembalikan suatu nilai.

Fungsi dibagi menjadi 2:

1. Tidak mengembalikan nilai (`void`)
2. Dapat mengembalikan nilai

### c. Fungsi dengan nilai balik

Fungsi yang memiliki nilai kembalian, harus ditentukan tipe data nilai baliknya pada saat deklarasi. Cara menentukan tipe data nilai balik fungsi adalah dengan menuliskan tipe data yang diinginkan setelah kurung parameter. Sedangkan cara untuk mengembalikan nilainya adalah dengan menggunakan keyword `return` diikuti data yang ingin dikembalikan. Eksekusi keyword `return` akan menjadikan proses dalam blok fungsi berhenti pada saat itu juga. Semua statement setelah keyword tersebut tidak akan dieksekusi.

### d. Parameter bertipe data sama

Khusus untuk fungsi yang tipe data parameternya sama, bisa ditulis dengan gaya yang unik. Tipe datanya dituliskan cukup sekali saja di akhir.

Contoh :

```
func nameOfFunc (parA type, parB type, parC type) returnType  
func nameOfFunc (parA, parB, parC type) returnType
```

### e. Fungsi multiple return

Umumnya fungsi hanya memiliki satu buah nilai balik saja. Jika ada kebutuhan dimana data yang dikembalikan harus banyak, biasanya digunakanlah tipe seperti `map`, `slice`, atau `struct` sebagai nilai balik. Cara membuat fungsi yang memiliki banyak nilai balik tidaklah sulit. Tinggal tulis saja pada saat deklarasi fungsi semua tipe data nilai yang dikembalikan, dan pada keyword `return` tulis semua data yang ingin dikembalikan. Karena fungsi tersebut memiliki banyak nilai balik, maka pada pemanggilannya harus disiapkan juga banyak variabel untuk menampung nilai kembalian yang ada (sesuai jumlah nilai balik fungsi).

## II. GUIDED

1. 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 permutasi a untuk kemungkinan yang lain

### Source code

```
package main

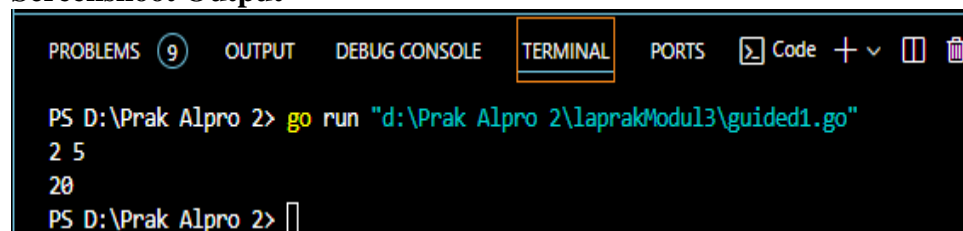
import "fmt"

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 Output



```
PS D:\Prak Alpro 2> go run "d:\Prak Alpro 2\laprakModul3\guided1.go"
2 5
20
PS D:\Prak Alpro 2> 
```

### Deskripsi Program

Program di atas dibuat untuk menghitung permutasi dari dua bilangan bulat. Pertama meminta pengguna untuk memasukkan dua angka dan membandingkan dua angka tersebut menggunakan percabangan if.

Perhitungan diawali dengan menghitung faktorial dari suatu angka setelah itu menghitung permutasi dengan rumus yang ada. Setelah perhitungan selesai, program akan mencetak hasilnya ke layar.

**\*Note:** Pada terminal terdapat 9 problem dikarenakan package dan func main digunakan juga di file lain

## 2. Program untuk menghitung keliling dan luas dari persegi

### Source code

```
package main

import (
    "fmt"
)

// Fungsi untuk menghitung luas persegi
func hitungLuas(sisi float64) float64 {
    return sisi * sisi
}

// Fungsi untuk menghitung keliling persegi
func hitungKeliling(sisi float64) float64 {
    return 4 * sisi
}

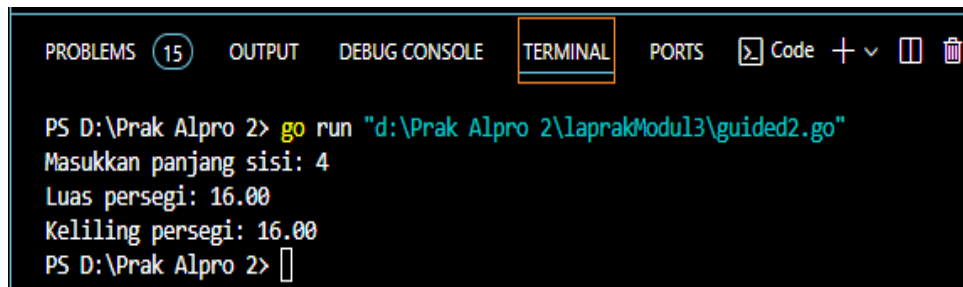
func main() {
    var sisi float64

    // Input panjang sisi persegi
    fmt.Print("Masukkan panjang sisi: ")
    fmt.Scan(&sisi)

    // Menghitung luas dan keliling menggunakan fungsi
    luas := hitungLuas(sisi)
    keliling := hitungKeliling(sisi)

    // Output hasil
    fmt.Printf("Luas persegi: %.2f\n", luas)
    fmt.Printf("Keliling persegi: %.2f\n", keliling)
}
```

### Screenshot Output



```
PROBLEMS 15 OUTPUT DEBUG CONSOLE TERMINAL PORTS Code + v [ ] [ ]
PS D:\Prak Alpro 2> go run "d:\Prak Alpro 2\laprakModul3\guided2.go"
Masukkan panjang sisi: 4
Luas persegi: 16.00
Keliling persegi: 16.00
PS D:\Prak Alpro 2> [ ]
```

### Deskripsi Program

Program di atas dibuat untuk menghitung luas dan keliling sebuah persegi. Pengguna diminta untuk memasukkan panjang sisi persegi. Sisi yang telah diinputkan digunakan untuk menghitung luas dengan rumus sisi x sisi dan keliling dengan rumus  $4 \times \text{sisi}$ . Hasil perhitungan tersebut ditampilkan dengan desimal yang memiliki 2 angka dibelakang koma.

**\*Note:** Pada terminal terdapat 15 problem dikarenakan package dan func main digunakan juga di file lain

### III. UNGUIDED

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!

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

#### Source code

```
/* Liya Khoirunnisa - 2311102124 */

package main

import "fmt"

func main() {
    // Deklarasi variabel
    var a, b, c, d int

    // Input nilai a
    fmt.Print("Masukkan nilai a: ")
    fmt.Scan(&a)

    // Input nilai b
    fmt.Print("Masukkan nilai b: ")
    fmt.Scan(&b)

    // Input nilai c
    fmt.Print("Masukkan nilai c: ")
    fmt.Scan(&c)

    // Input nilai d
    fmt.Print("Masukkan nilai d: ")
    fmt.Scan(&d)

    // Cek inputan
    if a >= c && b >= d {
        // Hitung permutasi dan kombinasi untuk a dan c
        fmt.Printf("\nP(%d,%d) = %d!/(%d)! = %d/%d = %d\n",
            a, c, a, a-c, faktorial(a), faktorial(a-c), permutasi(a, c))
        fmt.Printf("C(%d,%d) = %d!/(%d!x%d!) = %d/(%d x %d)
            = %d\n", a, c, a, c, a-c, faktorial(a), faktorial(c),
            faktorial(a-c), kombinasi(a, c))

        // Hitung permutasi dan kombinasi untuk b dan d
    }
}
```

```

        fmt.Printf("P(%d,%d) = %d!/(%d)! = %d/%d = %d\n", b,
d, b, b-d, faktorial(b), faktorial(b-d), permutasi(b, d))
        fmt.Printf("C(%d,%d) = %d!/(%d!x%d!) = %d/(%d x %d)
= %d\n", b, d, b, d, b-d, faktorial(b), faktorial(d),
faktorial(b-d), kombinasi(b, d))
    } else {
        fmt.Println("Input tidak valid, inputam harus a >= c
dan b >= d")
    }
}

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

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

// Fungsi kombinasi
func kombinasi(n, r int) int {
    return faktorial(n) / (faktorial(r) * faktorial(n-r))
}

```

## Screenshoot Output

```

PROBLEMS 13 OUTPUT DEBUG CONSOLE TERMINAL PORTS Code + -
PS D:\Prak Alpro 2\laprakModul3> go run "d:\Prak Alpro 2\laprakModul3\unguided1
.go"
Masukkan nilai a: 5
Masukkan nilai b: 10
Masukkan nilai c: 3
Masukkan nilai d: 10

P(5,3) = 5!/(2)! = 120/2 = 60
C(5,3) = 5!/(3!x2!) = 120/(6 x 2) = 10
P(10,10) = 10!/(0)! = 3628800/1 = 3628800
C(10,10) = 10!/(10!x0!) = 3628800/(3628800 x 1) = 1
PS D:\Prak Alpro 2\laprakModul3>

```

## Deskripsi Program

Program di atas dibuat untuk menghitung permutasi dan kombinasi berdasarkan input pengguna. Pengguna diminta menginputkan angka dengan syarat  $a \geq c$  dan  $b \geq d$ . Program akan mengecek apakah angka yang diinputkan sesuai dengan syarat yang ada menggunakan percabangan if. Jika angka tersebut sudah sesuai, maka program akan menghitung

faktorial, permutasi dan kombinasi dengan rumus yang ada. Untuk menghitung faktorial, permutasi, dan kombinasi menggunakan fungsi pengembalian nilai.

**\*Note:** Pada terminal terdapat 13 problem dikarenakan package dan func main digunakan juga di file lain

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

#### Source code

```
/* Liya Khoirunnisa - 2311102124*/

package main

import (
    "fmt"
)

// Fungsi untuk f(x)
func f(x int) int {
    return x * x
}

// Fungsi untuk g(x)
func g(x int) int {
    return x - 2
}

// Fungsi untuk h(x)
func h(x int) int {
    return x + 1
}

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

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

// Fungsi komposisi h(f(g(x)))
```



```

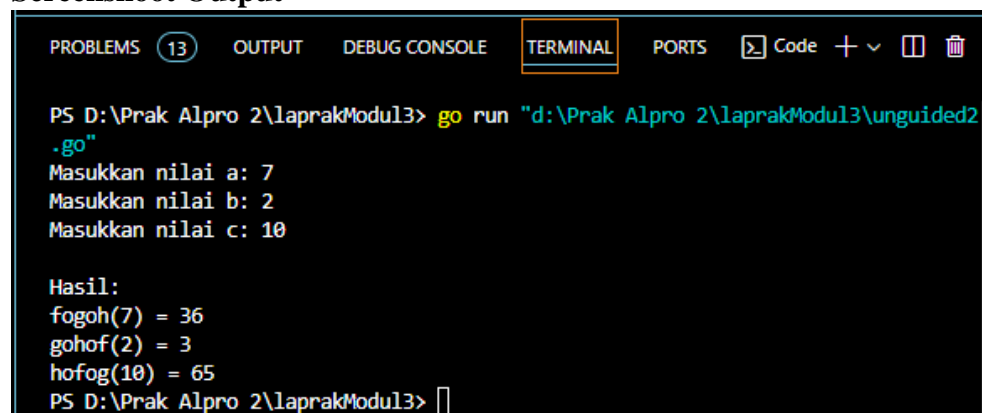
func hofog(x int) int {
    return h(f(g(x)))
}

func main() {
    var a, b, c int
    // Membaca input dari pengguna
    fmt.Print("Masukkan nilai a: ")
    fmt.Scan(&a)
    fmt.Print("Masukkan nilai b: ")
    fmt.Scan(&b)
    fmt.Print("Masukkan nilai c: ")
    fmt.Scan(&c)

    // Menghitung hasil komposisi fungsi
    fmt.Printf("\nHasil: \n")
    fmt.Printf("fogoh(%d) = %d\n", a, fogoh(a)) //
(fogoh) (a)
    fmt.Printf("gohof(%d) = %d\n", b, gohof(b)) //
(gohof) (b)
    fmt.Printf("hofog(%d) = %d\n", c, hofog(c)) //
(hofog) (c)
}

```

## Screenshoot Output



```

PROBLEMS 13 OUTPUT DEBUG CONSOLE TERMINAL PORTS Code + v
PS D:\Prak Alpro 2\laprakModul3> go run "d:\Prak Alpro 2\laprakModul3\unguided2
.go"
Masukkan nilai a: 7
Masukkan nilai b: 2
Masukkan nilai c: 10

Hasil:
fogoh(7) = 36
gohof(2) = 3
hofog(10) = 65
PS D:\Prak Alpro 2\laprakModul3>

```

## Deskripsi Program

Program di atas dibuat untuk menghitung komposisi fungsi dengan fungsi dasar  $f(x) = x^2$ ,  $g(x) = x - 2$  dan  $h(x) = x + 1$ . Pengguna diminta untuk menginputkan tiga angka. Tiga angka yang diinputkan dihitung dengan komposisi fungsi fogoh, gohof, dan hofog. Untuk menghitung  $f(x)$ ,  $g(x)$ ,  $h(x)$ , fogoh, gohof, dan hofog menggunakan fungsi pengembalian nilai. Hasil perhitungan akan dicetak ke layar.

**\*Note:** Pada terminal terdapat 13 problem dikarenakan package dan func main digunakan juga di file lain

3. 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". 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. Catatan: lihat paket math dalam lampiran untuk menggunakan fungsi math.sqrt() untuk menghitung akar kuadrat

### Source code

```
/*Liya Khoirunnisa - 2311102124*/

package main

import (
    "fmt"
    "math" // Untuk operasi matematika
)

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

// Fungsi untuk menentukan apakah titik berada di dalam lingkaran
func cekDalamLingkaran(cx, cy, r, x, y int) bool {
    return jarak(cx, cy, x, y) <= float64(r)
}

func main() {
    // Deklarasi variabel
    var cx1, cy1, r1, cx2, cy2, r2, x, y int

    // Input lingkaran 1
    fmt.Println("Lingkaran 1")
    fmt.Print("Masukkan titik pusat (cx1, cy1) dan radius r1: ")
    fmt.Scan(&cx1, &cy1, &r1)

    // Input lingkaran 2
    fmt.Println("\nLingkaran 2")
    fmt.Print("Masukkan titik pusat (cx2, cy2) dan radius
```

```

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

    // Input titik sembarang
    fmt.Print("\nMasukkan titik sembarang (x, y): ")
    fmt.Scan(&x, &y)

    // Mengecek apakah titik berada di dalam lingkaran 1
    dan/atau lingkaran 2
    titikDalam1 := cekDalamLingkaran(cx1, cy1, r1, x, y)
    titikDalam2 := cekDalamLingkaran(cx2, cy2, r2, x, y)

    // Menentukan output berdasarkan posisi titik
    fmt.Println("\nPosisi titik")
    if titikDalam1 && titikDalam2 {
        // Jika titik berada di dalam kedua lingkaran
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if titikDalam1 {
        // Jika titik hanya berada di dalam lingkaran 1
        fmt.Println("Titik di dalam lingkaran 1")
    } else if titikDalam2 {
        // Jika titik hanya berada di lingkaran 2
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        // Jika titik berada di luar kedua lingkaran
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}

```

## Screenshoot Output

```

PROBLEMS (12) OUTPUT DEBUG CONSOLE TERMINAL PORTS Code + - [ ] [ ] ...
PS D:\Prak Alpro 2\laprakModul3> go run "d:\Prak Alpro 2\laprakModul3\unguided3.go"
Lingkaran 1
Masukkan titik pusat (cx1, cy1) dan radius r1: 1 2 3

Lingkaran 2
Masukkan titik pusat (cx2, cy2) dan radius r2: 4 5 6

Masukkan titik sembarang (x, y): 7 8

Posisi titik
Titik di dalam lingkaran 2
PS D:\Prak Alpro 2\laprakModul3>

```

## Deskripsi Program

Program di atas dibuat untuk menentukan apakah sebuah titik sembarang berada di dalam lingkaran satu, dua, atau tidak ada dari kedua lingkaran. Pengguna diminta untuk memasukkan koordinat pusat dan radius dari dua lingkaran, serta titik sembarang. Selanjutnya program menghitung jarak antara titik sembarang dengan pusat masing-masing lingkaran dengan rumus  $jarak = \sqrt{(a - c)^2 + (b - d)^2}$ . Berdasarkan hasil perhitungan,

program akan memberikan output yang menyatakan apakah titik tersebut berada di dalam salah satu atau kedua lingkaran, atau berada di luar keduanya.

**\*Note:** Pada terminal terdapat 12 problem dikarenakan package dan func main digunakan juga di file lain

## DAFTAR PUSTAKA

- [1] Agung, N. (2017). *Dasar Pemrograman Golang*. Dinkes Provinsi Jatim. Tersedia dari <https://dinkes.jatimprov.go.id/userfile/dokumen/Dasar%20Pemrograman%20Golang.pdf>