

LAPORAN PRAKTIKUM
ALGORITMA PEMROGRAMAN 2
MODUL III
FUNGSI



Oleh :

Geranada Saputra Priambudi

2311102008

IF-11-06

Dosen Pengampu :

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

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY

I. DASAR TEORI

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, ...

Deklarasi Function

Deklarasi function dalam Go (Golang) dimulai dengan keyword `func`, diikuti oleh nama function yang biasanya menggambarkan tugas atau fungsinya. Parameter function ditulis di dalam tanda kurung setelah nama function, dan setiap parameter harus memiliki nama serta tipe datanya. Jika function mengembalikan nilai, tipe data dari nilai yang dikembalikan diletakkan setelah parameter. Isi dari function atau instruksi-instruksi yang akan dieksekusi ditulis di dalam kurung kurawal `{}`. Misalnya, function `tambah(a int, b int) int` menerima dua parameter `a` dan `b` bertipe `int` dan mengembalikan hasil penjumlahan dari keduanya yang juga bertipe `int`. Selain itu, jika suatu function tidak mengembalikan nilai, tidak perlu ada deklarasi return type, seperti pada function `tampilkanPesan()` yang hanya mencetak pesan tanpa mengembalikan nilai. Kata kunci `return` digunakan ketika function mengembalikan suatu nilai. Dengan struktur yang sederhana ini, Golang memudahkan pengelolaan kode yang modular dan reusable.

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.

II. GUIDED

1. Source Code:

```
package main

import "fmt"

func main() {
    var a, b int
    fmt.Println("Masukkan dua bilangan: ")
}
```

```

    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
    for i := 1; i <= n; i++ {
        hasil = hasil * i
    }
    return hasil
}

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

```

Output:

```

PS C:\Users\Asus\Documents\Alpro2> go run
Masukkan dua bilangan:
12 4
11880
PS C:\Users\Asus\Documents\Alpro2> 

```

Penjelasan:

Program diatas meminta pengguna untuk memasukkan dua bilangan bulat, kemudian menghitung permutasi dari kedua bilangan tersebut. Jika bilangan pertama lebih besar atau sama dengan bilangan kedua, program menghitung permutasi dari bilangan pertama memilih bilangan kedua, yaitu dengan rumus permutasi. Pada contoh input 12 dan 4, program menghitung permutasi $12P_4$, yang menghasilkan nilai 11,880. Fungsi faktorial() digunakan untuk menghitung nilai faktorial, dan fungsi permutasi() memanfaatkan hasil faktorial untuk menghitung nilai permutasi.

2. Source Code:

```

package main

import "fmt"

func luasBalok(p, l, t int) int {
    hasil := 2 * (p*l + p*t + l*t)
    return hasil
}

```

```

func volumeBalok(p, l, t int) int {
    volume := p * l * t
    return volume
}

func main() {
    var p, l, t int

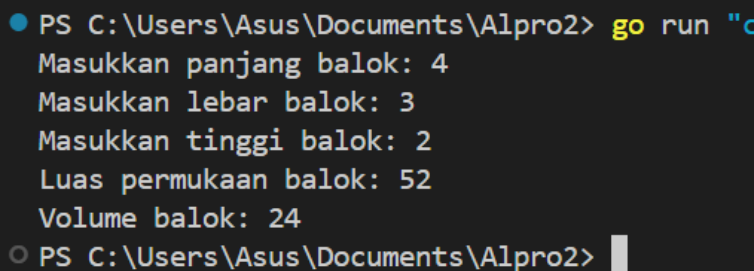
    fmt.Print("Masukkan panjang balok: ")
    fmt.Scan(&p)
    fmt.Print("Masukkan lebar balok: ")
    fmt.Scan(&l)
    fmt.Print("Masukkan tinggi balok: ")
    fmt.Scan(&t)

    luasPermukaan := luasBalok(p, l, t)
    volume := volumeBalok(p, l, t)

    fmt.Printf("Luas permukaan balok: %d\n", luasPermukaan)
    fmt.Printf("Volume balok: %d\n", volume)
}

```

Output:



```

PS C:\Users\Asus\Documents\Alpro2> go run "c:\Users\Asus\Documents\Alpro2\main.go"
Masukkan panjang balok: 4
Masukkan lebar balok: 3
Masukkan tinggi balok: 2
Luas permukaan balok: 52
Volume balok: 24
PS C:\Users\Asus\Documents\Alpro2>

```

Penjelasan:

Program diatas merupakan program dalam bahasa Go yang menghitung luas permukaan dan volume dari sebuah balok berdasarkan input panjang, lebar, dan tinggi yang dimasukkan oleh pengguna. Fungsi luasBalok menghitung luas permukaan balok, sedangkan fungsi volumeBalok menghitung volume balok. Setelah pengguna memasukkan nilai untuk panjang, lebar, dan tinggi balok, program akan memanggil kedua fungsi tersebut untuk menghitung hasilnya, kemudian mencetak luas permukaan dan volume balok ke layar. Pada contoh di atas, untuk balok dengan panjang 4, lebar 3, dan tinggi 2, hasil perhitungan menunjukkan luas permukaan balok adalah 52 dan volumenya adalah 24.

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 kalla membantu Jonas? (tidak tentunya ya :p)

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

```
package main

import "fmt"

var a, b, c, d int

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

func permutasi(n, r int) int {
    return faktorial(n) / faktorial(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  
>= d6 5 ")  
    }  
}
```

Output:

```
PS C:\Users\Asus\Documents\Alpro2> go run "c:\  
Masukkan input = 5 10 3 10  
60, 10  
3628800, 1  
PS C:\Users\Asus\Documents\Alpro2> █
```

Penjelasan:

Program di atas merupakan implementasi dalam bahasa Go (Golang) yang menghitung permutasi dan kombinasi dari dua pasang bilangan. Permutasi dan kombinasi adalah konsep matematika yang digunakan untuk menentukan urutan atau pengelompokan dari suatu himpunan. Program tersebut meminta input empat bilangan, yaitu a, b, c, dan d, dengan syarat bahwa a harus lebih besar atau sama dengan c, dan b lebih besar atau sama dengan d. Fungsi faktorial digunakan untuk menghitung nilai faktorial dari sebuah bilangan, yang merupakan dasar dari perhitungan permutasi dan kombinasi. Fungsi permutasi menghitung jumlah urutan berbeda yang dapat dibentuk dari elemen-elemen yang tersedia, sedangkan fungsi kombinasi menghitung jumlah cara untuk memilih sekumpulan elemen tanpa memperhatikan urutannya. Jika syarat terpenuhi, program akan mencetak hasil permutasi dan kombinasi untuk dua pasang bilangan tersebut. Jika tidak, program akan menampilkan pesan bahwa syarat tidak terpenuhi.

Pada contoh program tersebut menerima input berupa empat bilangan a = 5, b = 10, c = 3, dan d = 10, kemudian menghitung permutasi dan kombinasi berdasarkan pasangan (a, c) dan (b, d). Untuk pasangan pertama (a, c) yaitu (5, 3), permutasi dihitung dengan rumus, sehingga hasilnya adalah 60, 10. Untuk pasangan kedua (b, d) yaitu (10, 10), sehingga hasilnya adalah 3628800, 1. Hasil tersebut dicetak dalam dua baris: baris pertama untuk pasangan (5, 3) dan baris kedua untuk pasangan (10, 10).

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 $(hofog)(c)$!

Source Code:

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

func main() {
    var a, b, c int
    fmt.Print("Masukkan tiga bilangan bulat: ")
    fmt.Scan(&a, &b, &c)

    fmt.Println(fogoh(a))
    fmt.Println(gohof(b))
    fmt.Println(hofog(c))
}
```

Output:

```
PS C:\Users\Asus\Documents\Alpro2> go run "c:\u
Masukkan tiga bilangan bulat: 7 2 10
36
3
65
PS C:\Users\Asus\Documents\Alpro2> go run "c:\u
Masukkan tiga bilangan bulat: 5 5 5
16
24
10
PS C:\Users\Asus\Documents\Alpro2> go run "c:\u
Masukkan tiga bilangan bulat: 3 8 4
4
63
5
PS C:\Users\Asus\Documents\Alpro2> 
```

Penjelasan:

Program ini mengimplementasikan tiga fungsi matematika, yaitu $f(x) = x^2$, $g(x) = x - 2$, dan $h(x) = x + 1$, dan mengombinasikannya menjadi tiga komposisi fungsi berbeda: $(fogoh)(x)$, $(gohof)(x)$, dan $(hofog)(x)$. Setiap komposisi melibatkan pemanggilan beruntun fungsi, misalnya $(fogoh)(x)$ berarti menghitung $f(g(h(x)))$ dengan menjalankan fungsi $h(x)$ terlebih dahulu, kemudian $g(x)$ dari hasil $h(x)$, dan akhirnya $f(x)$ dari hasil $g(x)$. Program ini menerima tiga input bilangan bulat, yaitu a , b , dan c , lalu menghitung masing-masing komposisi fungsi berdasarkan input tersebut.

Sebagai contoh, jika pengguna memasukkan 7, 2, dan 10, hasil dari fungsi-fungsi tersebut adalah 36 untuk $(fogoh)(7)$, 3 untuk $(gohof)(2)$, dan 65 untuk $(hofog)(10)$. Program menggunakan pendekatan modular dengan memisahkan setiap fungsi untuk mempermudah implementasi dan memastikan komposisi fungsi dieksekusi secara benar. Output dari tiap komposisi fungsi dicetak pada baris terpisah, sehingga memberikan hasil sesuai dengan input yang dimasukkan.

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 dalam lingkaran 1 dan 2"

Source Code:

```
package main

import (
    "fmt"
    "math"
)

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

func dalamLingkaran(cx, cy, r, x, y float64) bool {
    distance := hitungJarak(cx, cy, x, y)
    return distance < r
}

func main() {
    var cx1, cy1, r1 float64
    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 1\n(cx1 cy1 r1): ")
    fmt.Scan(&cx1, &cy1, &r1)

    var cx2, cy2, r2 float64
    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 2\n(cx2 cy2 r2): ")
    fmt.Scan(&cx2, &cy2, &r2)

    var x, y float64
    fmt.Print("Masukkan koordinat titik sembarang (x y): ")
    fmt.Scan(&x, &y)

    dalamLingkaran1 := dalamLingkaran(cx1, cy1, r1, x, y)
    dalamLingkaran2 := dalamLingkaran(cx2, cy2, r2, x, y)

    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\Asus\Documents\Alpro2> go run "c:\Users\Asus\Documents\Alpro2\M
• Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 1 1 5
Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2): 8 8 4
Masukkan koordinat titik sembarang (x y): 2 2
Titik di dalam lingkaran 1
• PS C:\Users\Asus\Documents\Alpro2> go run "c:\Users\Asus\Documents\Alpro2\M
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
• PS C:\Users\Asus\Documents\Alpro2> go run "c:\Users\Asus\Documents\Alpro2\M
Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 5 10 5
Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2): -15 4 20
Masukkan koordinat titik sembarang (x y): 0 0
Titik di dalam lingkaran 2
• PS C:\Users\Asus\Documents\Alpro2> go run "c:\Users\Asus\Documents\Alpro2\M
Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 1 1 5
Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2): 8 8 4
Masukkan koordinat titik sembarang (x y): 15 20
Titik di luar lingkaran 1 dan 2
○ PS C:\Users\Asus\Documents\Alpro2> |
```

Penjelasan:

Program diatas digunakan untuk menentukan posisi sebuah titik sembarang (x, y) berdasarkan dua lingkaran yang masing-masing memiliki koordinat pusat (cx1, cy1) dan (cx2, cy2) serta radius r1 dan r2. Program menghitung jarak antara titik sembarang dan pusat lingkaran dengan menggunakan rumus jarak Euclidean. Jika jarak titik ke pusat lingkaran lebih kecil dari radius lingkaran, maka titik tersebut berada di dalam lingkaran. Program kemudian mengecek apakah titik berada di dalam lingkaran 1, lingkaran 2, atau kedua lingkaran sekaligus. Setelah menerima input koordinat pusat dan radius kedua lingkaran serta titik sembarang, program akan menentukan posisi titik tersebut dengan menguji jaraknya terhadap masing-masing lingkaran. Hasilnya bisa berupa empat kemungkinan: "Titik di dalam lingkaran 1 dan 2" jika titik berada di dalam kedua lingkaran, "Titik di dalam lingkaran 1" jika hanya di dalam lingkaran 1, "Titik di dalam lingkaran 2" jika hanya di dalam lingkaran 2, atau "Titik di luar lingkaran 1 dan 2" jika titik berada di luar kedua lingkaran.