

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL III

FUNGSI



Disusun Oleh :

Daffa Aryaputra / 2311102272

IF-11-06

Dosen Pengampu :

ABEDNEGO DWI SEPTIADI

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Fungsi dalam Golang adalah blok kode yang dirancang untuk melakukan tugas tertentu. Fungsi memungkinkan kita untuk memecah program menjadi bagian-bagian yang lebih kecil dan lebih teratur, sehingga memudahkan pengelolaan kode. Dalam Golang, sebuah fungsi didefinisikan dengan kata kunci ``func``, diikuti oleh nama fungsi, parameter (jika ada), dan tipe data yang dikembalikan (jika ada). Fungsi juga bisa dipanggil di berbagai bagian program, sehingga meminimalkan pengulangan kode.

Setiap fungsi dalam Golang bisa menerima sejumlah parameter dan mengembalikan nilai. Parameter adalah data yang dikirimkan ke fungsi saat dipanggil, sedangkan nilai yang dikembalikan bisa berupa tipe data tunggal atau banyak tipe data sekaligus. Misalnya, kita bisa membuat fungsi yang menerima dua angka dan mengembalikan hasil penjumlahannya. Dengan adanya tipe pengembalian yang jelas, Golang memastikan bahwa fungsi hanya mengembalikan tipe data yang sudah ditentukan, sehingga membantu mengurangi bug dan kesalahan tipe data.

Fungsi sebagai nilai berarti bahwa kita bisa menyimpan fungsi ke dalam variabel, mengirimkannya sebagai parameter ke fungsi lain, atau mengembalikannya sebagai hasil dari fungsi lain. Fitur ini memberikan fleksibilitas yang lebih besar dalam menulis kode, sehingga memudahkan implementasi konsep seperti callback, closure, dan higher-order functions.

II. GUIDED

1.

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)
}
```

Screenshoot Output



```
PS C:\Users\ACER> go run "e:\golang semester 3\modul3_guided1.go"
2 5
20
```

Deskripsi Program

Program ini menghitung permutasi dari dua angka yang dimasukkan oleh pengguna. Pertama, pengguna diminta untuk memasukkan dua angka, `a` dan `b`. Program kemudian membandingkan kedua angka tersebut. Jika `a` lebih besar atau sama dengan `b`, program akan menghitung permutasi

dengan `a` sebagai angka pertama. Jika tidak, `b` yang digunakan sebagai angka pertama.

Untuk menghitung permutasi, program memiliki dua bagian utama: satu untuk menghitung faktorial dari sebuah angka (hasil kali semua angka dari 1 sampai angka tersebut) dan satu untuk menghitung permutasi itu sendiri. Setelah faktorial dihitung, program membagi hasilnya untuk mendapatkan jumlah cara mengatur beberapa elemen dari sekumpulan elemen.

Misalnya, jika pengguna memasukkan angka 2 dan 5, program akan menghasilkan angka 20. Ini menunjukkan ada 20 cara berbeda untuk memilih dan mengatur 2 elemen dari 5 elemen yang tersedia.

2.

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    // Membuat reader untuk input dari user
    //reader := bufio.NewReader(os.Stdin)

    nama := "Hello World"

    // Menampilkan prompt kepada pengguna
    fmt.Println(nama)

    // Membaca input dari pengguna hingga baris baru (newline)
    //nama, _ := reader.ReadString('\n')

    // Menampilkan input yang diterima
    //fmt.Println("Halo,", nama)

    fmt.Println()
    balok()
}

func balok() {
```

```

var panjang, lebar, tinggi float64

fmt.Print("Panjang = ")
fmt.Scanln(&panjang)

fmt.Print("Lebar = ")
fmt.Scanln(&lebar)

fmt.Print("tinggi = ")
fmt.Scanln(&tinggi)

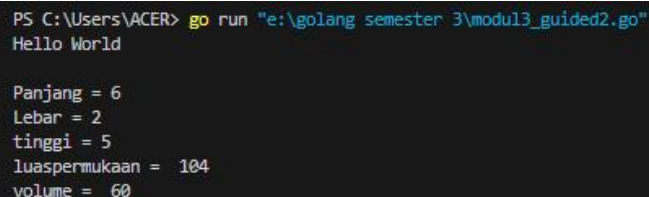
luaspermukaan := 2 * (panjang*lebar + panjang*tinggi +
lebar*tinggi)
volume := panjang * lebar * tinggi

fmt.Println("luaspermukaan = ", luaspermukaan)
fmt.Println("volume = ", volume)

}

```

Screenshoot Output



```

PS C:\Users\ACER> go run "e:\golang semester 3\modul3_guided2.go"
Hello World

Panjang = 6
Lebar = 2
tinggi = 5
luaspermukaan = 104
volume = 60

```

Deskripsi Program

Program ini ditulis dalam bahasa Go dan dirancang untuk menghitung luas permukaan dan volume dari sebuah balok berdasarkan dimensi yang dimasukkan oleh pengguna. Saat program dijalankan, ia terlebih dahulu menampilkan pesan "Hello World" sebagai sambutan. Setelah itu, program meminta pengguna untuk memasukkan tiga nilai: panjang, lebar, dan tinggi balok.

Pengguna akan diminta untuk mengisi panjang, lebar, dan tinggi secara berurutan. Setelah semua input diterima, program menghitung luas permukaan dan volume balok berdasarkan nilai yang dimasukkan. Akhirnya, program menampilkan hasil perhitungan tersebut. Jika pengguna memasukkan panjang, lebar, dan tinggi, program akan

menampilkan luas permukaan dan volume balok berdasarkan dimensi tersebut.

3.

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))

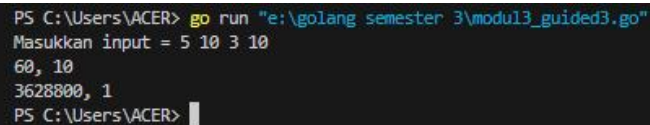
        // Baris kedua: Permutasi dan Kombinasi b terhadap d
        fmt.Printf("%d, %d\n", permutasi(b, d), kombinasi(b, d))
    }
}
```

```

    } else {
        fmt.Println("Syarat tidak terpenuhi: a harus >= c dan b
        harus >= d")
    }
}

```

Screenshoot Output



```

PS C:\Users\ACER> go run "e:\golang semester 3\modul3_guided3.go"
Masukkan input = 5 10 3 10
60, 10
3628800, 1
PS C:\Users\ACER>

```

Deskripsi Program

Program ini berfungsi untuk menghitung permutasi dan kombinasi dari dua pasang bilangan yang dimasukkan oleh pengguna. Cara kerjanya dimulai dengan meminta pengguna memasukkan empat angka: `a`, `b`, `c`, dan `d`. Program akan memeriksa apakah nilai `a` lebih besar atau sama dengan `c`, dan `b` lebih besar atau sama dengan `d`. Jika syarat ini tidak terpenuhi, program akan menampilkan pesan bahwa syarat tidak terpenuhi. Jika syarat tersebut terpenuhi, program akan melakukan perhitungan dan menampilkan dua baris hasil. Baris pertama menunjukkan hasil permutasi dan kombinasi dari `a` terhadap `c`, sedangkan baris kedua menunjukkan hasil permutasi dan kombinasi dari `b` terhadap `d`. Program menggunakan fungsi untuk menghitung faktorial, yang digunakan dalam proses perhitungan permutasi dan kombinasi. Hasil akhirnya adalah dua baris angka yang menampilkan hasil perhitungan berdasarkan input yang diberikan.

III. UNGUIDED

1.

Sourcecode

```

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
    }

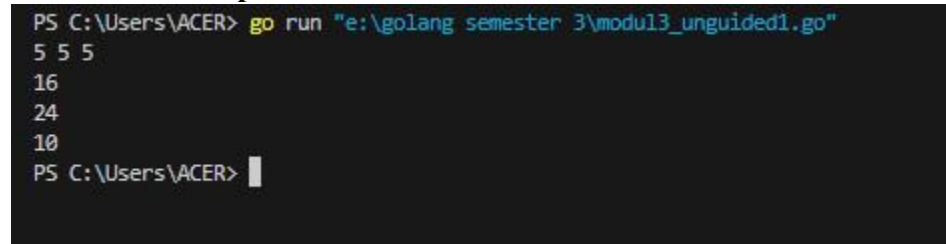
    func main() {
        var a, b, c int
        fmt.Scan(&a, &b, &c)

        fogoh := f(g(h(a)))
        gohof := g(h(f(b)))
        hofog := h(f(g(c)))

        fmt.Println(fogoh)
        fmt.Println(gohof)
        fmt.Println(hofog)
    }

```

Screenshoot Output



```

PS C:\Users\ACER> go run "e:\golang semester 3\modul3_unguided1.go"
5 5 5
16
24
10
PS C:\Users\ACER>

```

Deskripsi Program

Program ini dirancang untuk menghitung hasil dari tiga operasi matematika yang berbeda. Pertama, pengguna diminta untuk memasukkan tiga angka ('a', 'b', dan 'c'). Setelah itu, program memproses setiap angka dengan cara yang unik. Untuk angka 'a', program menambahkan 1, lalu mengurangkannya dengan 2, dan mengalikannya hasilnya. Untuk angka 'b', program mengalikannya dulu, kemudian menambahkan 1, dan akhirnya mengurangkannya dengan 2. Sedangkan untuk angka 'c', program mengurangkan 2 terlebih dahulu, lalu mengalikannya, dan terakhir menambahkan 1. Setelah semua perhitungan selesai, hasil dari ketiga operasi tersebut akan ditampilkan. Program ini membantu pengguna memahami bagaimana setiap angka diproses melalui serangkaian langkah yang berbeda untuk mendapatkan hasil akhir.

2.

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

type Circle struct {
    cx, cy, r float64
}

func isPointInside(c Circle, x, y float64) bool {
    distance := math.Sqrt(math.Pow(x-c.cx, 2) + math.Pow(y-c.cy, 2))
    return distance <= c.r
}

func determinePosition(c1, c2 Circle, x, y float64) string {
    in1 := isPointInside(c1, x, y)
    in2 := isPointInside(c2, x, y)

    if in1 && in2 {
        return "Titik di dalam lingkaran 1 dan 2"
    } else if in1 {
        return "Titik di dalam lingkaran 1"
    } else if in2 {
        return "Titik di dalam lingkaran 2"
    }
    return "Titik di luar lingkaran 1 dan 2"
}

func main() {
    var c1, c2 Circle
    var x, y float64

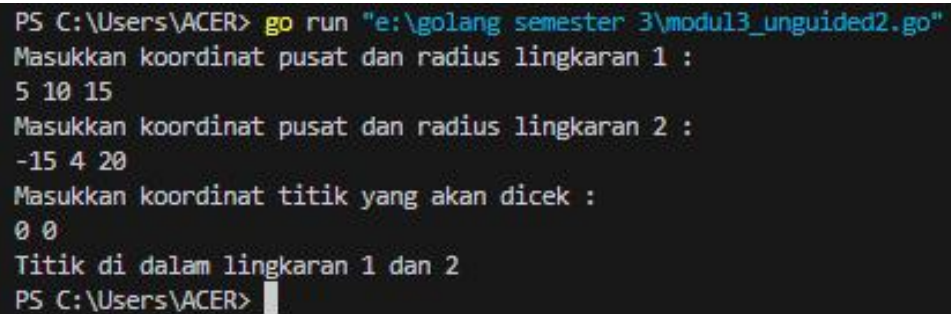
    fmt.Println("Masukkan koordinat pusat dan radius lingkaran 1 :")
    fmt.Scan(&c1.cx, &c1.cy, &c1.r)

    fmt.Println("Masukkan koordinat pusat dan radius lingkaran 2 :")
    fmt.Scan(&c2.cx, &c2.cy, &c2.r)
```

```
fmt.Println("Masukkan koordinat titik yang akan dicek :")
fmt.Scan(&x, &y)

result := determinePosition(c1, c2, x, y)
fmt.Println(result)
}
```

Screenshoot Output



```
PS C:\Users\ACER> go run "e:\golang semester 3\modul3_unguided2.go"
Masukkan koordinat pusat dan radius lingkaran 1 :
5 10 15
Masukkan koordinat pusat dan radius lingkaran 2 :
-15 4 20
Masukkan koordinat titik yang akan dicek :
0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\ACER> |
```

Deskripsi Program

Program ini bertujuan untuk menentukan apakah suatu titik berada di dalam atau di luar dua lingkaran yang diberikan. Program dimulai dengan meminta pengguna untuk memasukkan koordinat pusat (cx, cy) dan jari-jari (r) dari dua lingkaran. Kemudian, pengguna diminta untuk memasukkan koordinat titik yang ingin diperiksa (x, y).

Algoritma yang digunakan adalah menghitung jarak antara titik yang ingin dicek dengan pusat lingkaran menggunakan rumus jarak Euclidean. Jika jarak tersebut lebih kecil atau sama dengan jari-jari lingkaran, maka titik tersebut berada di dalam lingkaran; jika lebih besar, titik berada di luar lingkaran. Program melakukan pengecekan untuk kedua lingkaran dan mengembalikan hasil sebagai salah satu dari empat kemungkinan output: titik berada di dalam kedua lingkaran, hanya di dalam lingkaran pertama, hanya di dalam lingkaran kedua, atau di luar kedua lingkaran.

Hasil akhir dari program ditampilkan kepada pengguna sebagai keterangan posisi titik berdasarkan hasil perhitungan.