

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL 3
FUNGSI**



Oleh:

GALIH TRISNA

2311102050

IF-11-02

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024**

I. DASAR TEORI

Fungsi dalam Golang adalah blok kode yang dapat dipanggil dari berbagai tempat di program. Fungsi membantu membagi kode menjadi bagian-bagian yang lebih kecil, terstruktur, dan mudah dibaca. Setiap fungsi bisa menerima input dalam bentuk parameter dan bisa mengembalikan nilai. Dalam Golang, fungsi `main()` adalah titik awal eksekusi program. Parameter dalam Golang adalah variabel yang digunakan untuk menerima input ketika sebuah fungsi dipanggil. Parameter dideklarasikan di dalam tanda kurung setelah nama fungsi. Dalam Golang, parameter juga bisa bertipe, yang menunjukkan tipe data yang diterima oleh fungsi.

Strukturnya

```
func [nama_fungsi]([parameter_list]) [tipe_data_pengembalian] {  
    return [nilai_pengembalian]  
}
```

Contoh s

```
package main  
import "fmt"  
  
func greet(name string) {  
    fmt.Println("Hello,", name)  
}  
  
func main() {  
    greet("GALIH")  
}
```

II. GUIDED

1. 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

```
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
\Algoritma Pemrogramman 2\2-081024\GUIDED\1\guided_1.go"
5 3
60
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>
```

Penjelasan

Program di atas adalah program perhitungan permutasi. Proses didalam program ini dimulai dari dengan pengguna menginputkan a dan b. Setelah itu program akan mengecek apakah a lebih besar dari pada b. Jika iya maka program akan menjalankan

perhitungan a permutasi b. Jika tidak maka akan dilakukan sebaliknya yaitu b permutasi a. Permutasi dijalankan dengan memanggil fungsi permutasi dengan parameter a dan b. Fungsi permutasi merupakan fungsi yang digunakan untuk menghitung hasil dari permutasi 2 bilangan yaitu n dan r. Fungsi ini mengembalikan perhitungan $n!/(n-r)!$. Kemudian ada fungsi faktorial dengan parameter n yang merupakan angka yang akan difaktorialkan. Kemudian hasil dari faktorial akan diproses di fungsi permutasi yaitu membagi kedua faktorial. Setelah dibagi maka hasilnya akan dikembalikan ke fungsi main dan lalu ditampilkan.

2. Source Code

```
package main

import "fmt"

// Fungsi buat menghitung faktorial
func factorial(n int) int {
    if n == 0 || n == 1 {
        return 1
    }
    result := 1
    for i := 2; i <= n; i++ {
        result *= i
    }
    return result
}

// Fungsi buat menghitung permutasi
func permutation(n, r int) int {
    return factorial(n) / factorial(n-r)
}

// Fungsi buat menghitung kombinasi
func combination(n, r int) int {
    return factorial(n) / (factorial(r) * factorial(n-r))
}
```

```

func main() {
    // Input 4 bilangan
    var a, b, c, d int
    fmt.Println("Masukkan bilangan a, b, c, d (dengan spasi): ")
    fmt.Scanf("%d %d %d %d", &a, &b, &c, &d)

    // Cek syarat a >= c dan b >= d
    if a >= c && b >= d {
        // Menghitung permutasi dan kombinasi a dan c
        permutasiAC := permutation(a, c)
        kombinasiAC := combination(a, c)

        // Menghitung permutasi dan kombinasi b dan d
        permutasiBD := permutation(b, d)
        kombinasiBD := combination(b, d)

        // Output hasil
        fmt.Println("Permutasi(a, c) dan Kombinasi(a, c):",
permutasiAC, kombinasiAC)
        fmt.Println("Permutasi(b, d) dan Kombinasi(b, d):",
permutasiBD, kombinasiBD)
    } else {
        fmt.Println("Syarat a >= c dan b >= d tidak terpenuhi.")
    }
}

```

Screenshot

```

PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> .\081024\GUIDED\2\guided_2.go
Masukkan bilangan a, b, c, d (dengan spasi):
5 10 3 10
Permutasi(a, c) dan Kombinasi(a, c): 60 10
Permutasi(b, d) dan Kombinasi(b, d): 3628800 1
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> .\2\guided_2.go
Masukkan bilangan a, b, c, d (dengan spasi):
1 2 3 4
Syarat a >= c dan b >= d tidak terpenuhi.
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2>

```

Penjelasan

Program diatas merupakan program dari hasil modifikasi dari program guided 1. Hanya saja ditambahkan proses tambahan yaitu melakukan perhitungan kombinasi. Program ini pada awal dijalankan akan meminta 4 bilangan (a,b,c,d). A lebih besar atau sama dengan c dan b lebih besar atau sama dengan d. Jika tidak terpenuhi maka ada pesan dan proses tidak dilanjutkan. Jika terpenuhi maka program akan menghitung a permutasi c, a kombinasi c, b permutasi d, dan b kombinasi d. Perhitungan tersebut dijalankan dengan memanggil fungsi.

III. UNGUIDED

1. Source Code

```
package main

import "fmt"
import "math"

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

func didalam(cx float64, cy float64, r float64, x float64, y float64) bool
{
    var jarak_titik = jarak(x, y, cx, cy)
    if jarak_titik < r {
        return true
    } else {
        return false
    }
}

func main() {
    var cx1, cx3, cy1, cy2, r1, r2, x, y float64
    var result1, result2 bool
    fmt.Scanln(&cx1, &cy1, &r1)
    fmt.Scanln(&cx3, &cy2, &r2)
    fmt.Scanln(&x, &y)
    result1 = didalam(cx1, cy1, r1, x, y)
    result2 = didalam(cx3, cy2, r2, x, y)
    if result1 && result2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if result1 {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if result2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}
```

Screenshot

```
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2\2-081024\UNGUIDED> go run "c:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2\2-081024\UNGUIDED\unguided_1.go"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> go run "c:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2\2-081024\UNGUIDED\unguided_1.go"
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> go run "c:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2\2-081024\UNGUIDED\unguided_1.go"
5 10 15
-15 4 20
0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> go run "c:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2\2-081024\UNGUIDED\unguided_1.go"
1 1 5
8 8 4
15 20
Titik di luar lingkaran 1 dan 2
PS C:\Users\galih\Documents\belajar\Algoritma Pemrogramman 2> 
```

Penjelasan

Program diatas merupakan program menghitung apakah sebuah titik berada di dalam satu atau dua lingkaran. Program dimulai dengan pengguna memasukkan data dua lingkaran yang terdiri dari koordinat pusat (cx_1 , cy_1) dan radius r_1 untuk lingkaran pertama, serta pusat (cx_2 , cy_2) dan radius r_2 untuk lingkaran kedua. Setelah itu, pengguna memasukkan koordinat titik (x , y) yang akan diperiksa. Program kemudian memeriksa apakah titik tersebut berada di dalam lingkaran pertama, kedua, atau keduanya, dan mencetak hasilnya.

- Jika titik berada di kedua lingkaran, pesan "Titik di dalam lingkaran 1 dan 2" dicetak.
- Jika hanya berada di lingkaran pertama, pesan "Titik di dalam lingkaran 1".
- Jika hanya berada di lingkaran kedua, pesan "Titik di dalam lingkaran 2".
- Jika tidak di dalam keduanya, pesan "Titik di luar lingkaran 1 dan 2" akan dicetak.

Didalam proses tersebut terdapat fungsi jarak, Fungsi ini menghitung jarak antara dua titik pada bidang kartesian menggunakan rumus jarak Euclidean, yaitu akar kuadrat dari penjumlahan kuadrat selisih koordinat x dan y dari dua titik. Kemudian ada fungsi didalam, Fungsi ini memeriksa apakah sebuah titik (x, y) berada di dalam lingkaran dengan pusat (cx, cy) dan radius r . Fungsi ini memanggil fungsi jarak untuk menghitung jarak titik ke pusat lingkaran, dan jika jaraknya lebih kecil dari radius, fungsi mengembalikan nilai `true`, yang berarti titik berada di dalam lingkaran