

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL 3
FUNGSI



Oleh:

JONATHAN KANTOHE

2311102281

S1IF-11-02

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

I. DASAR TEORI

1. Definisi Fungsi

Fungsi adalah satu kesatuan rangkaian instruksi yang memberikan atau menghasilkan suatu nilai, biasanya memetakan input ke suatu nilai yang lain. Dalam konteks pemrograman, fungsi berfungsi untuk mengorganisir kode, memudahkan pemeliharaan, dan meningkatkan keterbacaan.

Fungsi dapat digunakan dalam berbagai cara, seperti:

- Menugaskan nilai ke suatu variabel
- Menjadi bagian dari ekspresi
- Menjadi argumen dari suatu subprogram.

2. Deklarasi fungsi

Deklarasi fungsi mirip dengan prosedur, dan biasanya terletak pada blok yang terpisah dari program utama. Notasi algoritma untuk mendeklarasikan fungsi adalah sebagai berikut: Dalam bahasa Go, notasi tersebut ditulis sebagai:

```
func <nama function> (<params>) <type> {  
    // deklarasi variabel lokal dari fungsi  
    // badan algoritma fungsi  
    return <value/variabel>  
}
```

3. Cara memanggil fungsi

Pemanggilan fungsi dilakukan dengan menuliskan nama fungsi beserta argumen yang diminta oleh parameter dari fungsi. Fungsi dapat di-assign ke suatu variabel, menjadi bagian dari ekspresi, dan juga dapat langsung ditampilkan dengan perintah output atau print. Contoh pemanggilan fungsi dalam pseudocode dan GoLang adalah sebagai berikut:

```
v1 = volumeTabung(r, t) // cara pemanggilan #1  
v2 = volumeTabung(r, t) + volumeTabung(15, t) // cara pemanggilan #2  
fmt.Println(volumeTabung(14, 18)) // cara pemanggilan #3
```

4. Contoh Program dengan Fungsi

Salah satu contoh penggunaan fungsi adalah untuk menghitung nilai faktorial dan permutasi. Berikut adalah contoh penulisan fungsi dalam program:

```
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)  
}
```

Dalam contoh di atas, fungsi faktorial dipanggil secara tidak langsung melalui fungsi permutasi, menunjukkan bagaimana fungsi dapat saling berinteraksi.

II. GUIDED

1. 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))
}
```

output :

```
PS C:\Users\lenovo\Downloads\JojoModul3> go
7
8
40320
PS C:\Users\lenovo\Downloads\JojoModul3> █
```

Penjelasan :

Program ini menghitung perulangan dari dua bilangan bulat yang di input, program ini memulai dengan memasukan ke dua bilangan bulat (a dan b), program akan menghitung faktorial dari bilangan yang lebih besar dan membaginya dengan faktorial dari selisih kedua bilangan tersebut.

2. Guided 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))
}
```

Output :

```
PS C:\Users\lenovo\Downloads\JojoModul3> go run
Masukkan bilangan a, b, c, d (dengan spasi):
10 8 6 4
Permutasi(a, c) dan Kombinasi(a, c): 151200 210
Permutasi(b, d) dan Kombinasi(b, d): 1680 70
PS C:\Users\lenovo\Downloads\JojoModul3> █
```

Pejelasan :

Program ini menghitung faktorial, permutasian dan kombinasi dari dua bilangan bulat, fungsi menghitung faktorial dari sebuah bilangan dengan menggunakan perulangan for. Fungsi permutasian menghitung permutasian dari dua bilangan dengan memanfaatkan hasil dari fungsi faktorial, menghitung kombinasi dari dua bilangan. Juga dengan memanfaatkan fungsi factorial.

III. UNGUIDED

1. Unguided 1

Source code :

```
package main

import (
    "fmt"
    "math"
)

// Fungsi jarak menghitung jarak antara dua titik (a,b) dan (c,d) dalam
// koordinat kartesian
// menggunakan rumus jarak Euclidean:  $\sqrt{(a-c)^2 + (b-d)^2}$ 
func jarak(a, b, c, d float64) float64 {
    return math.Sqrt(math.Pow(a-c, 2) + math.Pow(b-d, 2))
}

// Fungsi didalam memeriksa apakah suatu titik (x,y) berada di dalam
// lingkaran
// dengan pusat (cx,cy) dan radius r
func didalam(cx, cy, r, x, y float64) bool {
    return jarak(cx, cy, x, y) <= r
}

func main() {
    // Deklarasi variabel untuk koordinat dan radius dua lingkaran, serta
    // titik yang akan diperiksa
    var cx1, cy1, r1, cx2, cy2, r2, x, y float64

    // Meminta input dari pengguna untuk lingkaran 1
    fmt.Println("Masukkan koordinat titik pusat lingkaran 1 (cx, cy) dan
    radius (r):")
    fmt.Scanln(&cx1, &cy1, &r1)

    // Meminta input dari pengguna untuk lingkaran 2
    fmt.Println("Masukkan koordinat titik pusat lingkaran 2 (cx, cy) dan
    radius (r):")
    fmt.Scanln(&cx2, &cy2, &r2)

    // Meminta input koordinat titik yang akan diperiksa
    fmt.Println("Masukkan koordinat titik (x, y):")
    fmt.Scanln(&x, &y)

    // Memeriksa posisi titik terhadap kedua lingkaran dan mencetak
    // hasilnya
    if didalam(cx1, cy1, r1, x, y) && didalam(cx2, cy2, r2, x, y) {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if didalam(cx1, cy1, r1, x, y) {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if didalam(cx2, cy2, r2, x, y) {
        fmt.Println("Titik di dalam lingkaran 2")
    }
```

```
    } else {  
        fmt.Println("Titik di luar lingkaran ")  
    }  
}
```

Output :

```
PS C:\Users\lenovo\Downloads\JojoModul3> go run "c:\Users\lenovo\Down  
Masukkan koordinat titik pusat lingkaran 1 (cx, cy) dan radius (r):  
1 1 5  
Masukkan koordinat titik pusat lingkaran 2 (cx, cy) dan radius (r):  
8 8 4  
Masukkan koordinat titik (x, y):  
2 2  
Titik di dalam lingkaran 1  
PS C:\Users\lenovo\Downloads\JojoModul3> |
```

Pejelasan :

Program ini menentukan posisi suatu titik terhadap dua lingkungan yang diberikan. Program ini berjalan dengan cara memasukan koordinat pusat dan jari-jari dari dua lingkaran, serta koordianat titik yang ingin di periksa. Kemudian proragm akan menghiutng jarak anatar titik tersebut dengan pusat masing-masing lingkaran menggunakan rumus jarak euclidean. Jika jarak antara titik dan pusat lingkaran lebih kecil atau sama dengan jari-jari lingkaran, maka titik tersebut berada di dalam lingkaran. Program akan memeriksa posisi titik terhadap kedua lingkaran dan mencetak hasil pemeriksaan ke layar, yaitu apakah titik berada di dalam lingkaran pertama, kedua, atau di luar kedua lingkaran. Singkatnya, kode ini menyelesaikan masalah geometri sederhana yaitu menentukan posisi suatu titik relatif terhadap dua lingkaran.