

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMOGRAMAN 2**

MODUL 3

FUNGSI



Oleh:

HAIKAL SATRIATAMA

2311102066

IF-11-02

S1 TEKNIK INFORMATIKA

INSTITUT TEKNOLOGI TELKOM PURWOKERTO

2024

I. DASAR TEORI

Struktur dan Deklarasi Fungsi

Untuk mendeklarasikan sebuah fungsi di Golang, digunakan kata kunci `func`, diikuti dengan nama fungsi, parameter (jika ada), dan tipe data nilai balik (jika fungsi tersebut mengembalikan nilai). Berikut adalah contoh struktur dasar dari fungsi:

```
func namaFungsi(parameter1 tipeData1, parameter2
tipeData2) tipeDataKembali {
    // blok kode
    return nilaiKembali
}
```

Contoh implementasi pada code

```
package main

import "fmt"

func tambah(a int, b int) int {
    return a + b
}

func main() {
    hasil := tambah(5, 3)
    fmt.Println(hasil) // Output: 8
}
```

Jenis-Jenis Fungsi

1. Fungsi Tanpa Parameter: Fungsi yang tidak memerlukan input dari pengguna.
2. Fungsi dengan Parameter: Fungsi yang memerlukan satu atau lebih input untuk menjalankan logika tertentu.
3. Fungsi dengan Nilai Balik: Fungsi yang mengembalikan nilai setelah dieksekusi.

Keunggulan Menggunakan Fungsi

1. Modularitas: Membagi program menjadi bagian-bagian kecil yang lebih mudah dikelola.
2. Reusabilitas: Fungsi dapat digunakan kembali tanpa perlu menulis ulang kode.
3. Memudahkan Pemeliharaan: Perubahan pada satu fungsi tidak mempengaruhi bagian lain dari program.

II. GUIDED

1. Guided 1

```
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 F:\Semester 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\PraktikumModul3Fungsi> cd latihan1
PS F:\Semester 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\PraktikumModul3Fungsi\latihan1> go run lat1.go
12 12
479001600
PS F:\Semester 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\PraktikumModul3Fungsi\latihan1> |
```

Penjelasan

Program ini menghitung dan menampilkan hasil permutasi dari dua bilangan bulat yang diinput oleh pengguna. Permutasi adalah pengaturan elemen-elemen dalam urutan tertentu. Dalam program ini, pengguna diminta untuk memasukkan dua bilangan bulat, `a` dan `b`. Program kemudian memeriksa apakah `a` lebih besar atau sama dengan `b`. Jika ya, maka fungsi `permutasi` dipanggil dengan argumen `a` dan `b`. Jika tidak, maka fungsi `permutasi` dipanggil dengan argumen `b` dan `a`.

Fungsi `permutasi` menghitung permutasi menggunakan rumus faktorial:

$$\text{permutasi}(n, r) = n! / (n - r)!$$

di mana `n` adalah jumlah elemen total dan `r` adalah jumlah elemen yang akan diurutkan. Fungsi `faktorial` digunakan untuk menghitung faktorial dari suatu bilangan bulat. Hasil permutasi kemudian ditampilkan pada layar.

2. Guided 2

```
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)

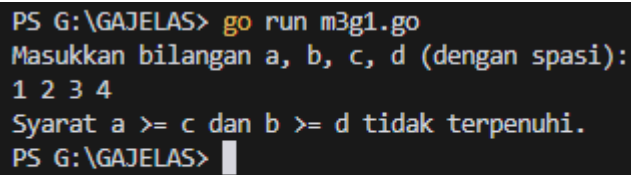
    // Cek syarat a >= c dan b >= d
    if a >= c && b >= d {
        // Menghitung permutasi dan kombinasi a dan c
        permutasiAC := permutation(a, c)
        kombinasiAC := combination(a, c)

        // 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.")  
    }  
}
```

Screenshoot Output



```
PS G:\GAJELAS> go run m3g1.go  
Masukkan bilangan a, b, c, d (dengan spasi):  
1 2 3 4  
Syarat a >= c dan b >= d tidak terpenuhi.  
PS G:\GAJELAS> █
```

Penjelasan

Menghitung faktorial, permutasi, dan kombinasi. Faktorial adalah hasil kali semua bilangan bulat positif kurang dari atau sama dengan suatu bilangan. Permutasi adalah banyaknya cara untuk menyusun sejumlah objek dalam urutan tertentu. Kombinasi adalah banyaknya cara untuk memilih sejumlah objek tanpa memperhatikan urutan.

III. UNGUIDED

1. Unguided 1

```
package main

import (
    "fmt"
)

func checkPointPosition(cx1, cy1, r1, cx2, cy2, r2, x, y
int) string {

    d1 := (x - cx1) * (x - cx1) + (y - cy1) * (y - cy1)
    d2 := (x - cx2) * (x - cx2) + (y - cy2) * (y - cy2)

    inCircle1 := d1 <= r1 * r1
    inCircle2 := d2 <= r2 * r2

    if inCircle1 && inCircle2 {
        return "Titik di dalam lingkaran 1 dan 2"
    } else if inCircle1 {
        return "Titik di dalam lingkaran 1"
    } else if inCircle2 {
        return "Titik di dalam lingkaran 2"
    } else {
        return "Titik di luar lingkaran 1 dan 2"
    }
}

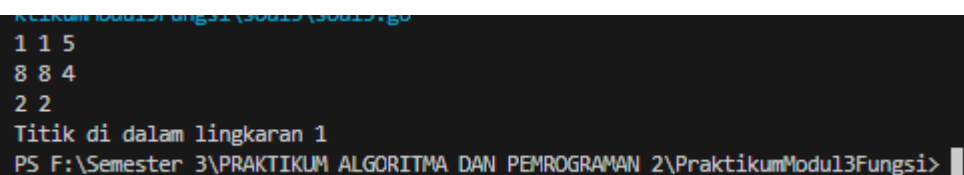
func main() {
    var cx1, cy1, r1 int
    var cx2, cy2, r2 int
    var x, y int

    fmt.Scan(&cx1, &cy1, &r1)
    fmt.Scan(&cx2, &cy2, &r2)

    fmt.Scan(&x, &y)

    result := checkPointPosition(cx1, cy1, r1, cx2, cy2,
r2, x, y)
    fmt.Println(result)
}
```

Screenshoot Output



```
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS F:\Semester 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\PraktikumModul3Fungsi>
```

Penjelasan :

Program ini menentukan posisi suatu titik terhadap dua lingkaran. Lingkaran pertama memiliki pusat (cx1, cy1) dan jari-jari r1, sedangkan lingkaran kedua memiliki pusat (cx2, cy2) dan jari-jari r2.

Jika jarak titik dari pusat suatu lingkaran lebih kecil atau sama dengan jari-jari lingkaran tersebut, maka titik tersebut berada di dalam lingkaran. Jika tidak, maka titik tersebut berada di luar lingkaran.

Fungsi ``checkPointPosition`` mengembalikan string yang menunjukkan posisi titik terhadap kedua lingkaran. Jika titik berada di dalam kedua lingkaran, maka fungsi mengembalikan "Titik di dalam lingkaran 1 dan 2". Jika titik berada di dalam hanya satu lingkaran, maka fungsi mengembalikan "Titik di dalam lingkaran 1" atau "Titik di dalam lingkaran 2". Jika titik berada di luar kedua lingkaran, maka fungsi mengembalikan "Titik di luar lingkaran 1 dan 2".