

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMOGRAMAN II
MODUL III
FUNCTION



Oleh:

NAMA : DWI HESTI ARIANI

NIM : 2311102094

KELAS : 11- IF -02

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

I. DASAR TEORI

a. Function

Fungsi merupakan salah satu sub program yang melakukan satu atau lebih proses yang mengembalikan sebuah nilai untuk dikirimkan ke program utama (return). Fungsi juga berperan untuk melakukan pemilahan kode program agar penulisan kode tidak ditulis menjadi satu pada program utama . Fungsi pada dasarnya sama dengan prosedur.

Jika prosedur didalamnya merupakan proses yang dijalankan, namun pada fungsi didalamnya serangkain proses yang berjalan dan menghasilkan keluaran. Fungsi dapat dipanggil oleh program utamanya ataupun oleh prosedur atau fungsi lain yang juga merupakan bagian dari program utamanya. Pada sebuah program dapat terdiri dari satu atau lebih fungsi dan satu program utamanya.

b. Penerapan Fungsi pada Bahasa Go

Penggunaan fungsi caranya cukup mudah dengan menuliskan keyword func kemudian diikuti nama fungsi, lalu kurung () (yang bisa diisi parameter), dan diakhiri dengan kurung kurawal untuk membungkus blok kode. Parameter merupakan variabel yang menempel di fungsi yang nilainya ditentukan saat pemanggilan fungsi tersebut.

Parameter sifatnya opsional, suatu fungsi bisa tidak memiliki parameter, atau bisa saja memiliki satu atau banyak parameter (tergantung kebutuhan). Data yang digunakan sebagai value parameter saat pemanggilan fungsi biasa disebut dengan argument parameter (atau argument). Contoh implementasi fungsi

```
package main

import "fmt"

import "strings"

func main() {

    var names = []string{"John", "Wick"}

    printMessage("halo", names)

}

func printMessage(message string, arr []string) {

    var nameString = strings.Join(arr, " ")

}
```

```
fmt.Println(message, nameString)
```

```
}
```

c. Pemanggilan Fungsi

Pemanggilan fungsi cukup dilakukan dengan penulisan nama fungsi beserta argument yang diminta oleh parameter dari fungsi . Perbedaannya dengan prosedur ialah fungsi bisa di assign ke suatu variable , menjadi bagian dari ekspresi , dan argument dari suatu subprogram

II. GUIDED

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.")
    }
}

```

Output Program

```

PS C:\Users\dwih> go run "c:\Users\dwih\OneDrive\Documents\Program Go\Permutasi dan Kombinasi.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\dwih>

```

Deskripsi Program

Program diatas ialah program Go sederhana penggunaan fungsi matematika untuk menghitung permutasi dan kombinasi dari dua pasang bilangan. Fungsi **factorial** digunakan untuk menghitung faktorial dari sebuah bilangan. Faktorial adalah hasil perkalian dari semua bilangan positif yang lebih kecil atau sama dengan bilangan tersebut.

Fungsi **combination** digunakan untuk menghitung kombinasi dari dua bilangan. Kombinasi adalah hasil perkalian dari semua bilangan positif yang lebih kecil atau sama dengan bilangan pertama, dibagi dengan hasil perkalian dari semua bilangan positif yang lebih kecil atau sama dengan bilangan kedua dan bilangan pertama minus bilangan kedua.

User diminta untuk memasukkan empat bilangan, yaitu **a**, **b**, **c**, dan **d**. Kemudian, program memeriksa apakah **a** lebih besar atau sama dengan **c** dan **b** lebih besar atau sama dengan **d**. Jika syarat ini terpenuhi, program menghitung permutasi dan kombinasi dari **a** dan **c**, serta **b** dan **d**

Guided 2

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 Program

```
10 9
3628800
```

Deskripsi Program

Program diatasprogram yang meminta pengguna untuk memasukkan dua bilangan, yaitu **a** dan **b**. Kemudian, program memeriksa apakah **a** lebih besar atau sama dengan **b**. Jika **a** lebih besar atau sama dengan **b**, program menghitung

permutasi dari **a** dan **b**. Jika tidak, program menghitung permutasi dari **b** dan **a**, digunakan juga untuk menghitung faktorial dari sebuah bilangan. Faktorial adalah hasil perkalian dari semua bilangan positif yang lebih kecil atau sama dengan bilangan tersebut.

III. UNGUIDED

Source Code

Unguided 1

```
package main

import (
    "fmt"
    "math"
)

// Fungsi untuk menghitung jarak antara dua titik (a, b) dan (c, d)
func jarak(a, b, c, d float64) float64 {
    return math.Sqrt(math.Pow(a-c, 2) + math.Pow(b-d, 2))
}

// Fungsi untuk mengecek apakah titik (x, y) berada di dalam lingkaran
func didalam(cx, cy, r, x, y float64) bool {
    return jarak(cx, cy, x, y) <= r
}

func main() {
    // Masukan untuk dua lingkaran
    var cx1, cy1, r1 float64
    var cx2, cy2, r2 float64
    var x, y float64

    // Input dari pengguna
    fmt.Println("Masukkan pusat dan radius lingkaran 1 (cx1, cy1, r1):")
    fmt.Scan(&cx1, &cy1, &r1)
    fmt.Println("Masukkan pusat dan radius lingkaran 2 (cx2, cy2, r2):")
    fmt.Scan(&cx2, &cy2, &r2)
    fmt.Println("Masukkan koordinat titik sembarang (x, y):")
    fmt.Scan(&x, &y)

    // Mengecek posisi titik (x, y) relatif terhadap lingkaran 1 dan 2
    diLingkaran1 := didalam(cx1, cy1, r1, x, y)
    diLingkaran2 := didalam(cx2, cy2, r2, x, y)
}
```



```

// Menentukan output berdasarkan posisi titik
if diLingkaran1 && diLingkaran2 {
    fmt.Println("Titik di dalam lingkaran 1 dan 2")
} else if diLingkaran1 {
    fmt.Println("Titik di dalam lingkaran 1")
} else if diLingkaran2 {
    fmt.Println("Titik di dalam lingkaran 2")
} else {
    fmt.Println("Titik di luar lingkaran 1 dan 2")
}
}

```

Output Program

```

PS C:\Users\dwih\OneDrive\Documents\SEMESTER 3\PRAKTIKUM ALPRO2\unguidedmodul3\unguided1.3.go"
Masukkan pusat dan radius lingkaran 1 (cx1, cy1, r1):
1 1 5
Masukkan 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\dwih\OneDrive\Documents\SEMESTER 3\PRAKTIKUM ALPRO2\unguidedmodul3\unguided1.3.go"
Masukkan pusat dan radius lingkaran 1 (cx1, cy1, r1):
1 2 3
Masukkan pusat dan radius lingkaran 2 (cx2, cy2, r2):
4 5 6

```

```

Masukkan 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\dwih\OneDrive\Documents\SEMESTER 3\PRAKTIKUM ALPRO2\unguidedmodul3\unguided1.3.go"
Masukkan pusat dan radius lingkaran 1 (cx1, cy1, r1):
1 1 5
Masukkan 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\dwih\OneDrive\Documents\SEMESTER 3\PRAKTIKUM ALPRO2\unguidedmodul3\unguided1.3.go"

```

Deskripsi Program

Program diatas ialah program untuk menentukan apakah sebuah titik berada di dalam, di luar, atau di dalam salah satu dari dua lingkaran yang didefinisikan oleh pengguna.

Fungsi **jarak** digunakan untuk menghitung jarak antara dua titik (**a**, **b**) dan (**c**, **d**). Fungsi ini menggunakan rumus jarak Euclidean, yaitu akar kuadrat dari jumlah kuadrat perbedaan koordinat x dan y.

Fungsi **didalam** digunakan untuk mengecek apakah titik (**x**, **y**) berada di dalam lingkaran dengan pusat (**cx**, **cy**) dan radius **r**. Fungsi ini menggunakan fungsi **jarak** untuk menghitung jarak antara titik (**x**, **y**) dan pusat lingkaran, lalu membandingkan hasilnya dengan radius lingkaran. Jika jaraknya kurang dari atau sama dengan radius, maka titik berada di dalam lingkaran.

Fungsi **main** adalah fungsi utama program ini. Fungsi ini meminta pengguna untuk memasukkan pusat dan radius dua lingkaran, serta koordinat titik sembarang. Kemudian, fungsi ini menggunakan fungsi **didalam** untuk mengecek apakah titik berada di dalam lingkaran 1, lingkaran 2, atau di luar kedua lingkaran. Berdasarkan hasilnya, program akan mencetak pesan yang sesuai.