

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL III
FUNGSI**



Disusun Oleh :

Muhammad Djoko Susilo / 2311102212

11-06

Dosen Pengampu :

ABEDNEGO DWI SEPTIADI

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Dasar Teori

Fungsi

Definisi fungsi adalah suatu program yang dirancang untuk melakukan proses-proses tertentu, Biasanya fungsi bisa mengembalikan/menghasilkan nilai. Suatu subprogram dikatakan fungsi apabila:

1. Bisa mengembalikan nilai
2. Bisa dipanggil berulang kali dengan argumen yang berbeda
3. Fokus pada perhitungan nilai

Fungsi digunakan ketika suatu nilai biasanya diperlukan, seperti:

1. Menetapkan nilai ke suatu variabel.
2. Menjadi bagian dari suatu ekspresi.
3. Menjadi argumen dari subprogram lain, dan sebagainya.

Karena itu selalu pilih nama fungsi yang bisa menggambarkan nilai yang ingin dikembalikan, seperti kata benda atau kata sifat. Contoh **luasPersegi**, **volumeBalok**, **isValid**, **isEmpty** dan sebagainya.

Deklarasi Fungsi

Deklarasi fungsi sama dengan prosedur, yaitu berada pada blok yang terpisah dengan program utama.

Notasi Algoritma

```
Function namaFungsi(parameter1, parameter2, ...) :  
tipe_data_hasil  
  
kamus:  
  
    variabel_lokal1, variabel_lokal2 : tipe_data  
  
algoritma:  
  
    // Langkah-langkah perhitungan atau proses  
  
    hasil <- perhitungan_atau_proses(parameter1,  
parameter2, ...)
```

```
    return hasil  
end function
```

Notasi fungsi dalam bahasa Go

```
func faktorial(n int) int {  
    hasil := 1  
    for i := 1; i <= n; i++ {  
        hasil *= i  
    }  
    return hasil  
}
```

II. GUIDED

III. Soal 1

Soal Studi Case

Buatlah sebuah program beserta fungsi yang digunakan untuk menghitung nilai faktorial dan permutasi.

Masukan terdiri dari dua buah bilangan positif a dan b

Keluaran berupa sebuah bilangan bulat yang menyatakan nilai a permutasi b apabila $a \geq b$

Sourcecode

```
package main

import "fmt"

// Fungsi untuk menghitung faktorial
func faktorial(n int) int {
    hasil := 1
    for i := 1; i <= n; i++ {
        hasil *= i
    }
    return hasil
}

// Fungsi untuk menghitung permutasi
func permutasi(n, r int) int {
    return faktorial(n) / faktorial(n-r)
}

func main() {
    var a, b int

    // Input nilai a dan b
    fmt.Print("Masukkan nilai a: ")
    fmt.Scan(&a)
    fmt.Print("Masukkan nilai b: ")
    fmt.Scan(&b)

    // Periksa kondisi dan hitung permutasi sesuai nilai a
    // dan b
    if a >= b {
        fmt.Printf("Permutasi(%d, %d) = %d\n", a, b,
            permutasi(a, b))
    } else {
        fmt.Printf("Permutasi(%d, %d) = %d\n", b, a,
            permutasi(b, a))
    }
}
```

```
}
```

Screenshoot Output

```
PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_Djoko_Modul-3\guided1.go"
Masukkan nilai a: 5
Masukkan nilai b: 3
Permutasi(5, 3) = 60
PS D:\vsCode> |
```

Deskripsi Program

Program diatas adalah sebuah code yang berfungsi untuk menghitung permutasian dengan rumus $(n! / (n - r)!)$, input yang dimasukan adalah tipe data integer.

Cara kerja dari program ini adalah mula mula user memasukan input dari nilai n dan nilai r, setelah itu nilai yang dimasukan oleh user akan dikelola oleh fungsi permutasi yang memanggil fungsi faktorial untuk menghitung nilai x(nilai yang diterima oleh fungsi faktorial) pangkat 2 dan akan mengembalikan nilai dalam bentuk integer yang akan diterima fungsi permutasi dan setelah itu akan dihitung menggunakan rumus $(n! / (n - r)!)$, fungsi permutasi akan mengembalikan nilai kedalam program dan akan ditampilkan ke layar.

Soal 2

Soal Studi Case

Buatlah fungsi untuk menghitung volume (Panjang x Lebar x Tinggi) dan luas permukaan balok $(2 \times ((\text{Panjang} \times \text{Lebar}) + (\text{Panjang} \times \text{Tinggi}) + (\text{Tinggi} \times \text{Lebar})))$ dengan input menggunakan tipe data real.

Sourcecode

```
package main

import "fmt"

// Fungsi untuk menghitung volume balok
func volumeBalok(p, l, t float64) float64 {
    return p * l * t
}

// Fungsi untuk menghitung luas permukaan balok
func luasBalok(p, l, t float64) float64 {
```

```

        return 2 * ((p * l) + (p * t) + (l * t))
    }

    func main() {
        var p, l, t float64

        // Input nilai panjang (p), lebar (l), dan tinggi (t)
        fmt.Print("Masukkan panjang balok: ")
        fmt.Scan(&p)
        fmt.Print("Masukkan lebar balok: ")
        fmt.Scan(&l)
        fmt.Print("Masukkan tinggi balok: ")
        fmt.Scan(&t)

        // Output hasil volume dan luas permukaan balok
        fmt.Printf("Volume dari balok tersebut adalah: %.2f\n", volumeBalok(p, l, t))
        fmt.Printf("Luas dari balok tersebut adalah: %.2f\n", luasBalok(p, l, t))
    }
}

```

Screenshoot Output

```

PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_Djoko_Modul-3\guided2.go"
Masukkan panjang balok: 10
Masukkan lebar balok: 10
Masukkan tinggi balok: 10
Volume dari balok tersebut adalah: 1000.00
Luas dari balok tersebut adalah: 600.00
PS D:\vsCode>

```

Deskripsi Program

Program adalah sebuah perhitungan volume dan luas permukaan balok. Volume balok memiliki rumus (Panjang x Lebar x Tinggi) dan Luas permukaan balok ($2 \times ((\text{Panjang} \times \text{Lebar}) + (\text{Panjang} \times \text{Tinggi}) + (\text{Tinggi} \times \text{Lebar}))$) dengan user dapat menginputkan tipe data real.

Cara kerja program ini adalah user diminta untuk memasukan nilai dari panjang, lebar dan tinggi balok yang bertipe data real, setelah itu data data yang sudah diambil akan dimasukan kedalam fungsi volume balok dan akan dihitung berapa nilai volume baloknya setelah dihitung akan kembali nilai hasil dari perhitungan tersebut dan cara tersebut berlaku juga untuk fungsi luas permukaan balok.

UNGUIDED

Soal 1

Soal Studi Case

Buatkan program untuk menghitung kombinasi dan permutasi.

Masukan terdiri dari empat buah bilangan asli a, b, c dan d yang dipisahkan oleh spasi dengan syarat $a \leq c$ dan $b \geq d$.

Keluaran terdiri dari dua baris, baris pertama adalah hasil permutasi dan kombinasi a terhadap c . sedangkan baris kedua adalah hasil permutasi dan kombinasi b terhadap d

Catatan permutasi (P) dan kombinasi (C) dari n terhadap r ($n \geq r$) dapat dihitung dengan menggunakan persamaan berikut

$$P(n, r) = \frac{n!}{(n-r)!}, \text{ sedangkan } C(n, r) = \frac{n!}{r!(n-r)!}$$

Sourcecode

```
package main

import (
    "fmt"
)

// Fungsi untuk menghitung faktorial
func faktorial(n int) int {
    hasil := 1
    for i := 1; i <= n; i++ {
        hasil *= i
    }
    return hasil
}

// Fungsi untuk menghitung permutasi
func permutasian(n, r int) int {
    return faktorial(n) / faktorial(n-r)
}

// Fungsi untuk menghitung kombinasi
func kombinasi(n, r int) int {
    return faktorial(n) / (faktorial(r) * faktorial(n-r))
}

func main() {
    var a1, b1, a2, b2 int

    // Input nilai a1, b1, a2, b2
```

```

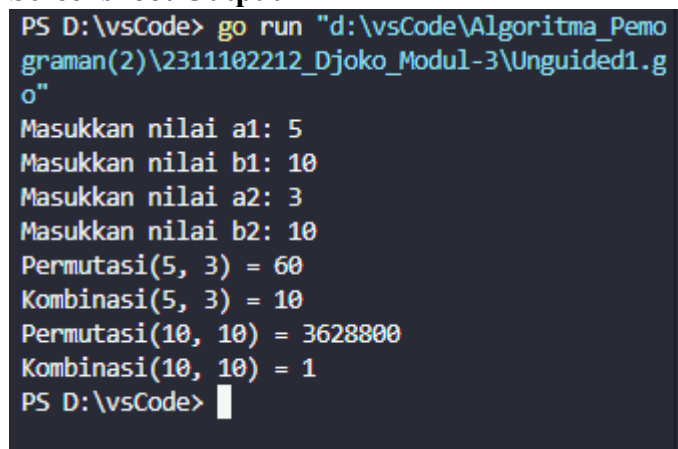
    fmt.Print("Masukkan nilai a1: ")
    fmt.Scan(&a1)
    fmt.Print("Masukkan nilai b1: ")
    fmt.Scan(&b1)
    fmt.Print("Masukkan nilai a2: ")
    fmt.Scan(&a2)
    fmt.Print("Masukkan nilai b2: ")
    fmt.Scan(&b2)

    // Menghitung permutasi dan kombinasi
    p1 := permutasian(a1, a2)
    p2 := permutasian(b1, b2)
    c1 := kombinasi(a1, a2)
    c2 := kombinasi(b1, b2)

    // Output hasil perhitungan
    fmt.Printf("Permutasi(%d, %d) = %d\n", a1, a2, p1)
    fmt.Printf("Kombinasi(%d, %d) = %d\n", a1, a2, c1)
    fmt.Printf("Permutasi(%d, %d) = %d\n", b1, b2, p2)
    fmt.Printf("Kombinasi(%d, %d) = %d\n", b1, b2, c2)
}

```

Screenshoot Output



```

PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_Djoko_Modul-3\Unguided1.go"
Masukkan nilai a1: 5
Masukkan nilai b1: 10
Masukkan nilai a2: 3
Masukkan nilai b2: 10
Permutasi(5, 3) = 60
Kombinasi(5, 3) = 10
Permutasi(10, 10) = 3628800
Kombinasi(10, 10) = 1
PS D:\vsCode>

```

Deskripsi Program

Program digunakan untuk menghitung kombinasi dan permutasian sesuai dengan rumus yang sesuai dengan soalnya.

Cara kerjanya adalah user memasukan nilai n dan nilai r yang berurutan mulai dari input pertama yaitu n dan seterusnya yaitu n, r dan r.. Nilai yang sudan dimasukan kedalam program akan dihitung sesuai dengan yang diperintahkan yaitu akan dihitung pertama yaitu

Fungsi permutasian : mengambil nilai yang akan dihitung dahulu faktorialnya lalu akan dihitung menggunakan rumus permutasian dan dikembalikan dengan nilai yang sudah jadinya.

Fungsi kombinasi : mengambil nilai yang akan dihitung dahulu faktorialnya lalu akan dihitung menggunakan rumus kombinasi dan dikembalikan dengan nilai yang sudah jadinya.

Soal 2

Soal Studi Case

Diberikan tiga buah fungsi matematika yaitu $f(x) = x^2$, $g(x) = x - 2$ dan $h(x) = x + 1$. Fungsi komposisi $(fogoh)(x)$ artinya adalah $f(g(h(x)))$ Tuliskan, $f(x)$, $g(x)$ dan $h(x)$ dalam bentuk function.

Masukan terdiri dari sebuah bilangan bulat a, b dan c yang dipisahkan oleh spasi.

Keluaran terdiri dari tiga baris. Baris pertama adalah $(fogoh)(a)$, baris kedua $(gohof)(b)$, dan baris ketiga adalah $(hofog)(c)$.

Sourcecode

```
package main

import (
    "fmt"
)

// Fungsi fogoh
func fogoh(a int) int {
    h := a + 1
    g := h - 2
    f := g * g
    return f
}

// Fungsi gohof
func gohof(a int) int {
    f := a * a
    h := f + 1
    g := h - 2
    return g
}

// Fungsi hofog
func hofog(a int) int {
    g := a - 2
    f := g * g
    h := f + 1
    return h
}
```

```

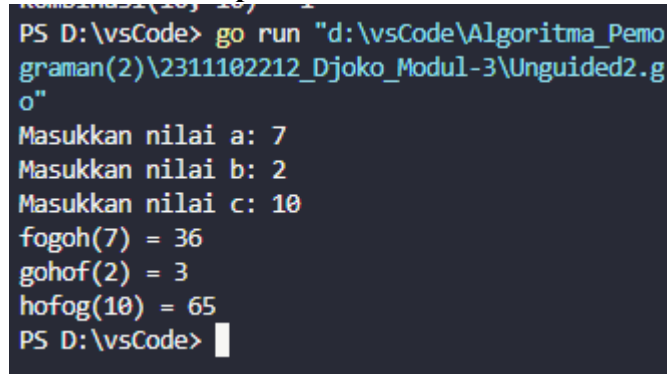
func main() {
    var a, b, c int

    // Input nilai a, b, c
    fmt.Print("Masukkan nilai a: ")
    fmt.Scan(&a)
    fmt.Print("Masukkan nilai b: ")
    fmt.Scan(&b)
    fmt.Print("Masukkan nilai c: ")
    fmt.Scan(&c)

    // Output hasil perhitungan fungsi
    fmt.Printf("fogoh(%d) = %d\n", a, fogoh(a))
    fmt.Printf("gohof(%d) = %d\n", b, gohof(b))
    fmt.Printf("hofog(%d) = %d\n", c, hofog(c))
}

```

Screenshoot Output



```

PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_Djoko_Modul-3\Unguided2.go"
Masukkan nilai a: 7
Masukkan nilai b: 2
Masukkan nilai c: 10
fogoh(7) = 36
gohof(2) = 3
hofog(10) = 65
PS D:\vsCode>

```

Deskripsi Program

Ini adalah program yang bisa menghitung fungsi matematika sesuai dengan soalnya.

Cara kerjanya adalah user memasukan nilai 3 kali untuk disimpan dan dihitung sesuai dengan perintahnya, seperti nilai a akan dihitung menggunakan rumus $(fogoh)(a)$

Soal 3

Soal Studi Case

Buatkan sebuah program yang bisa menentukan posisi sebuah titik sembarang (x, y) berdasarkan dua lingkaran yang memiliki koordinat titik pusaran (cx, cy) dengan radius r .

Masukan terdiri dari tiga masukan yang pertama koordinat titik pusat dan radius dari lingkaran 1 dan yang kedua lingkaran 2 nya, dan baris ketiga adlah koordinat titik pusat sembarang.

Keluaran berupa string yang menyatakan posisi titik ada didalam lingkaran 1 dan 2 atau didalam lingkaran 1 atau didalam lingkaran 2 atau diluar lingkaran keduanya.

Fungsi untuk menghitung jarak titik $(a,b)^2$ dan $(c,d)^2$ dimana rumus jarak adalah

$$Jarak = \sqrt{(a - c)^2 + (b - d)^2}$$

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

// Fungsi untuk menghitung kuadrat
func kuadrat(a float64) float64 {
    return a * a
}

// Fungsi untuk mencari posisi titik sembarang terhadap
lingkaran
func mencariTitikSembarang(cx, cy, r, x, y float64) string
{
    // Menghitung jarak Euclidean antara titik (x, y) dan
pusat lingkaran (cx, cy)
    jarak := math.Sqrt(kuadrat(x-cx) + kuadrat(y-cy))

    // Menentukan apakah titik di dalam, di luar, atau di
tepi lingkaran
    if jarak < r {
        return "dl" // dl: di dalam lingkaran
    } else if jarak == r {
        return "tl" // tl: di tepi lingkaran
    } else {
        return "ll" // ll: di luar lingkaran
    }
}

func main() {
    var cx1, cy1, r1, cx2, cy2, r2, x, y float64

    // Input pusat dan radius lingkaran pertama
    fmt.Print("Masukkan cx1, cy1, r1 untuk lingkaran 1: ")
    fmt.Scan(&cx1, &cy1, &r1)
```

```

// Input pusat dan radius lingkaran kedua
fmt.Print("Masukkan cx2, cy2, r2 untuk lingkaran 2: ")
fmt.Scan(&cx2, &cy2, &r2)

// Input titik sembarang (x, y)
fmt.Print("Masukkan titik sembarang (x, y): ")
fmt.Scan(&x, &y)

// Mencari posisi titik sembarang terhadap kedua
lingkaran
d1 := mencariTitikSembarang(cx1, cy1, r1, x, y)
d2 := mencariTitikSembarang(cx2, cy2, r2, x, y)

// Menentukan posisi titik relatif terhadap lingkaran
1 dan 2
if d1 == "d1" && d2 == "d1" {
    fmt.Println("Titik di dalam lingkaran 1 dan 2")
} else if d1 == "l1" && d2 == "l1" {
    fmt.Println("Titik di luar lingkaran 1 dan 2")
} else if d1 == "d1" && d2 == "l1" {
    fmt.Println("Titik di dalam lingkaran 1")
} else if d1 == "l1" && d2 == "d1" {
    fmt.Println("Titik di dalam lingkaran 2")
} else {
    fmt.Println("Titik berada di tepi lingkaran")
}
}

```

Screenshoot Output

```
PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_D
joko_Modul-3\Unguided3.go"
Masukkan cx1, cy1, r1 untuk lingkaran 1: 1 1 5
Masukkan cx2, cy2, r2 untuk lingkaran 2: 8 8 4
Masukkan titik sembarang (x, y): 2 2
Titik di dalam lingkaran 1
PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_D
joko_Modul-3\Unguided3.go"
Masukkan cx1, cy1, r1 untuk lingkaran 1: 1 2 3
Masukkan cx2, cy2, r2 untuk lingkaran 2: 4 5 6
Masukkan titik sembarang (x, y): 7 8
Titik di dalam lingkaran 2
PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_D
joko_Modul-3\Unguided3.go"
Masukkan cx1, cy1, r1 untuk lingkaran 1: 5 10 15
Masukkan cx2, cy2, r2 untuk lingkaran 2: -15 4 20
Masukkan titik sembarang (x, y): 0 0
Titik di dalam lingkaran 1 dan 2
PS D:\vsCode> go run "d:\vsCode\Algoritma_Pemograman(2)\2311102212_D
joko_Modul-3\Unguided3.go"
Masukkan cx1, cy1, r1 untuk lingkaran 1: 1 1 5
Masukkan cx2, cy2, r2 untuk lingkaran 2: 8 8 4
Masukkan titik sembarang (x, y): 15 20
Titik di luar lingkaran 1 dan 2
PS D:\vsCode> █
```

Deskripsi Program

Program digunakan untuk menentukan letak dari titik sembarang antara dua lingkaran.

Cara kerjanya adalah user memasukan titik pusat dari lingkaran beserta radiusnya dan titik sembarangnya, lalu program akan menghitung nilai yang sudah dimasukan menggunakan fungsi mencari TitikSembarang yang mengambil nilai titik pusat, radius dan titik sembarangnya.

Fungsi mencari TitikSembarang : menghitung jarak menggunakan library math yang memanggil fungsi sqrt atau akar untuk menghitung hasil akhirnya. Saat memanggil sqrt fungsi ini juga memanggil fungsi kuadrat untuk mengetahui nilai kuadrat dair sebuah nilai. Fungsi ini aka mengembalikan nilai dalam bentuk string berupa "dr" apabila jarak $< r$, "tl" apabila jarak $= r$, dan terakhir "ll" apabila jarak $> r$.

Setelah mendapatkan hasil dari kedua lingkaran akan ditentukan apakah posisi titik ada didalam lingkaran 1 dan 2 atau didalam lingkaran 1 atau didalam lingkaran 2 atau diluar lingkaran keduanya menggunakan percabangan.