

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL III
FUNGSI**



Disusun Oleh :

Afra Lintang Maharani (2311102258)

IF 11 05

Dosen Pengampu :

Arif Amrulloh, S.Kom., M.Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Fungsi adalah sekumpulan blok kode yang dibungkus dengan nama tertentu. Penerapan fungsi yang tepat akan menjadikan kode lebih modular dan juga dry (singkatan dari don't repeat yourself) yang artinya kita tidak perlu menuliskan banyak kode untuk kegunaan yang sama berulang kali. Cukup deklarasikan sekali saja blok kode sebagai suatu fungsi, lalu panggil sesuai kebutuhan.

A. Penerapan Fungsi

Fungsi *main()* sendiri merupakan fungsi utama pada program Go, yang akan dieksekusi ketika program dijalankan. Selain fungsi *main()*, kita juga bisa membuat fungsi lainnya. Dan caranya cukup mudah, yaitu 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).

Dalam Go fungsi yang sederhana itu bentuknya seperti ini`:

```
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)

}
```

Output :

```
novalagung:belajar-golang $ go run bab17.go
halo John Wick
novalagung:belajar-golang $
```

Pada kode di atas, sebuah fungsi baru dibuat dengan nama *printMessage()* memiliki 2 buah parameter yaitu string **message** dan slice string **arr**

Fungsi tersebut dipanggil dalam *main()*, dalam pemanggilannya disisipkan dua buah argument parameter.

1. Argument parameter pertama adalah string "halo" yang ditampung parameter message
2. Argument parameter ke-2 adalah slice string names yang nilainya ditampung oleh parameter arr

Di dalam *printMessage()*, nilai **arr** yang merupakan slice string digabungkan menjadi sebuah string dengan pembatas adalah karakter spasi. Penggabungan slice dapat dilakukan dengan memanfaatkan fungsi *strings.Join()* (berada di dalam package strings)

B. Fungsi Dengan Return Value / Nilai Balik

Sealin parameter, fungsi bisa memiliki attribute return value atau nilai balik. Fungsi yang memiliki return value, saat deklarasinya harus ditentukan terlebih dahulu tipe data dari nilai baliknya. Fungsi yang tidak mengembalikan nilai apapun (contohnya seperti fungsi *main()* dan *printMessage()*) biasa disebut dengan *void function*. Berikut adalah contoh fungsi dengan pengembalian nilai.

```
package main

import (
    "fmt"
    "math/rand"
    "time"
)

var randomizer =
rand.New(rand.NewSource(time.Now().Unix()))

func main() {
    var randomValue int
```

```

    randomValue = randomWithRange(2, 10)
    fmt.Println("random number:", randomValue)

    randomValue = randomWithRange(2, 10)
    fmt.Println("random number:", randomValue)

    randomValue = randomWithRange(2, 10)
    fmt.Println("random number:", randomValue)
}

func randomWithRange(min, max int) int {
    var value = randomizer.Int()%(max-min+1) + min
    return value
}

```

Output kode ini akan mencetak output berikut:

```

[novalagung:belajar-golang $ go run bab17.go]
random number: 9
random number: 6
random number: 2
novalagung:belajar-golang $

```

Fungsi *randomWithRange()* didesain untuk generate angka acak sesuai dengan *range* yang ditentukan lewat parameter, yang kemudian angka tersebut dijadikan nilai balik fungsi. Cara menentukan tipe data nilai balik fungsi adalah dengan menuliskan tipe data yang diinginkan setelah kurung parameter. Bisa dilihat pada kode di atas, bahwa *int* merupakan tipe data nilai balik fungsi *randomWithRange()*.

Sedangkan cara untuk mengembalikan nilai itu sendiri adalah dengan menggunakan keyword *return* diikuti data yang dikembalikan. Pada contoh di atas, *return value* artinya nilai variabel *value* dijadikan nilai kembalian fungsi. Eksekusi keyword *return* akan menjadikan proses dalam blok fungsi berhenti pada saat itu juga. Semua statement setelah keyword tersebut tidak akan dieksekusi.

II. GUIDED

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 $a \geq b$ atau b permutasi a untuk kemungkinan yang lain.

Sourcecode

```
package main

import "fmt"

func main() {
    var a, b int
    fmt.Scan(&a, &b)
    if a >= b {
        fmt.Print(permutasi(a, b))
    } else {
        fmt.Print(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 D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided\Guided1.go"
10 20
670442572800
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided\Guided1.go"
3 4
24
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided>
```

Deskripsi

Program ini digunakan untuk menghitung inputan dari user lalu menghitung nilai permutasi. Program ini memiliki beberapa fungsi

- `faktorial(n int)` Fungsi ini menghitung faktorial dari angka n, yaitu $n!$
- `permutasi(n, r int)` Fungsi ini menghitung nilai permutasi menggunakan rumus $P(n, r) = \frac{n!}{(n-r)!}$ dengan memanggil fungsi faktorial.
- Fungsi utama memeriksa apakah $a \geq b$ dan menghitung permutasi berdasarkan perbandingan tersebut.

Memiliki Algoritma :

- Input a, b
- Jika $a \geq b$, hitung permutasi $P(a, b)$, jika tidak maka $P(b, a)$
- Kemudian tampilkan output dari P

2. Soal Studi Case

Sourcecode

```
package main

import "fmt"

func main() {
    var s int
    fmt.Print("Input Sisi: ")
    fmt.Scan(&s)

    luas := luasPersegi(s)
    kel := kelilingPersegi(s)

    fmt.Println("Hasil Luas: ", luas)
    fmt.Print("Hasil keliling: ", kel)

}

func luasPersegi(s int) int {
    return s * s
}

func kelilingPersegi(s int) int {
    return 4 * s
}
```

Screenshoot Output

```
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided\Guided2.go"
Input Sisi: 5
Hasil Luas: 25
Hasil keliling: 20
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Guided> █
```

Deskripsi Program

Program ini digunakan untuk mengukur luas dan keliling sebuah persegi. Program ini memiliki beberapa fungsi :

- `func luasPersegi(s int)` digunakan untuk mengukur luas persegi
- `func kelilingPersegi(s int)` digunakan untuk mengukur keliling persegi

Memiliki algoritma :

- Input (s)
- Definisikan variabel **luas** dengan fungsi *luasPersegi(s)*
- Definisikan variabel **kel** dengan fungsi *kelilingPersegi(s)*
- Kemudian panggil variabel **luas dan kel**
- Output ditampilkan

III. UNGUIDED

1. Soal Studi Case

Minggu Ini, mahasiswa Fakultas Informatika mendapatkan tugas dari mata kuliah matematika diskrit untuk mempelajari kombinasi dan permutasi. Jonas salah seorang mahasiswa, iseng untuk mengimplementasikannya ke dalam suatu program. Oleh karena itu bersediakah kalian membantu Jonas?

(tidak tentunya ya :p)

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"

func main() {
    var a, b, c, d int
    fmt.Println("Input nilai a, b, c, d : ")
    fmt.Scan(&a, &b, &c, &d)

    if a >= c && b >= d {
        fmt.Println("Hasil Permutasi : ", permutasi(a, c))
        fmt.Println("Hasil Kombinasi : ", kombinasi(a, c))
        fmt.Println("Hasil Permutasi : ", permutasi(b, d))
        fmt.Println("Hasil Kombinasi : ", kombinasi(b, d))
    } else {
        fmt.Print("Kondisi 'a >= c && b >= d' tidak terpenuhi ", a, b, c, d)
    }
}

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)
}

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

```

Screenshoot Output

```

PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided1> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided1\main.go"
Input nilai a, b, c, d : 5 10 3 10
Hasil Permutasi : 60
Hasil Kombinasi : 10
Hasil Permutasi : 3628800
Hasil Kombinasi : 1
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided1> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided1\main.go"
Input nilai a, b, c, d : 8 0 2 0
Hasil Permutasi : 56
Hasil Kombinasi : 28
Hasil Permutasi : 1
Hasil Kombinasi : 1
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided1> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided1\main.go"
Input nilai a, b, c, d : 5 2 10 32
Kondisi 'a >= c && b >= d' tidak terpenuhi 5 2 10 32

```

Deskripsi Program

Program ini merupakan implementasi lebih lanjut dari guided 1 yang dimana program ini digunakan untuk menghitung permutasi dan kombinasi. Program ini memiliki fungsi yang sama dengan guided1 tetapi memiliki fungsi lain juga berikut fungsi dari program diatas :

- `faktorial(n int)` Fungsi ini menghitung faktorial dari angka n, yaitu $n!$
- `permutasi(n, r int)` Fungsi ini menghitung nilai permutasi menggunakan rumus $P(n,r) = n!/(n-r)!$ dengan memanggil fungsi faktorial.
- `Kombinasi(n, r int)` fungsi ini menghitung nilai kombinasi dengan rumus $C(n,r) = n! / r! * (n-r)!$
- Fungsi utama memeriksa apakah $a \geq b$ dan menghitung permutasi berdasarkan perbandingan tersebut.

Algoritmanya adalah :

- Input (a, b, c, d) type data int

- Cek kondisi apakah $a \geq c \ \&\& \ b \geq d$
- Jika true maka, lakukan permutasi dan kombinasi dari $P(a,c)$, $C(a,c)$, $P(b,d)$, $C(b,d)$ dan lakukan print output hasil dari permutasi dan kombinasi
- Jika false maka output akan "Kondisi ' $a \geq c \ \&\& \ b \geq d$ ' tidak terpenuhi "

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 $(f \circ g \circ h)(x)$ 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 $(f \circ g \circ h)(a)$, baris kedua $(g \circ h \circ f)(b)$, dan baris ketiga adalah $(h \circ f \circ g)(c)$!

Sourcecode

```
package main

import "fmt"

func main() {
    var a, b, c int
    fmt.Print("Input a, b, c : ")
    fmt.Scanln(&a, &b, &c)

    hasil1 := fogoh(a)
    hasil2 := gohof(b)
    hasil3 := hofog(c)
    fmt.Printf("(Fogoh) (%d) = %d\n", a, hasil1)
    fmt.Printf("(Gohof) (%d) = %d\n", b, hasil2)
    fmt.Printf("(Hofog) (%d) = %d\n", c, hasil3)

}

//catatan fungsi utama
// f(x) := x * x
// g(x) := x - 2
// h(x) := x + 1

func fogoh(x int) int {
    //f(g(hx))
    hx := x + 1
```

```

    ghx := hx - 2
    Hasil := ghx * ghx

    return Hasil
}

func gohof(x int) int {
    //g(h(fx))
    fx := x * x
    hfx := fx + 1
    Hasil := hfx - 2

    return Hasil
}

func hofog(x int) int {
    //h(f(gx))
    gx := x - 2
    fgx := gx * gx
    Hasil := fgx + 1

    return Hasil
}

```

Screenshoot Output

```

PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided2> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided2\main.go"
Input a, b, c : 7 2 10
(fogoh)(7) = 36
(gohof)(2) = 3
(hofog)(10) = 65
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided2> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided2\main.go"
Input a, b, c : 5 5 5
(fogoh)(5) = 16
(gohof)(5) = 24
(hofog)(5) = 10
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided2> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided2\main.go"
Input a, b, c : 3 8 4
(fogoh)(3) = 4
(gohof)(8) = 63
(hofog)(4) = 5
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided2>

```

Deskripsi Program

Program diatas digunakan untuk menghitung sebuah fungsi dari $f(g(h(x)))$, $g(h(f(x)))$, dan $h(f(g(x)))$, dengan inputan *user*. Memiliki beberapa fungsi yang dimana memiliki dasaran dari fungsi yang telah ada yaitu :

- $f(x) = x * x$
- $g(x) = x - 2$
- $h(x) = x + 1$
- `func fogoh(x int)` Fungsi ini merupakan hasil perkalian dari beberapa fungsi yang telah ada seperti $f(g(h(x)))$
- `func hofog(x int)` merupakan fungsi dari $h(f(g(x)))$
- `func gohof(x int)` merupakan fungsi dari $g(h(f(x)))$

Jadi fungsi yang telah ada diatas merupakan perkalian dari ketiga fungsi yang telah ada yang membedahkan hanya posisi ketiganya yang berubah ubah. Memiliki algoritma :

- Input (a,b,c)
- hasil1 := fogoh(a)
- hasil2 := gohof(b)
- hasil3 := hofog(c)
- Program akan menjalankan fungsi yang telah ada, dari inputan user, seperti variabel hasil1, hasil2, dan hasil3 diatas
- Kemudian program akan menampilkan hasil1, hasil2, dan hasil3

3. Soal Studi Case

[Lingkaran] Suatu lingkaran didefinisikan dengan koordinat titik pusat (ex, cy) dengan radiüs 7. Apabila diberikan dua buah lingkaran, maka tentukan posisi sebuah titik sembarang (x, y) berdasarkan dua lingkaran tersebut.

Masukan terdiri dari beberapa tiga baris. Baris pertama dan kedua adalah koordinat titik pusat dan radius dari lingkaran 1 dan lingkaran 2, sedangkan baris ketiga adalah koordinat titik sembarang. Asumsi sumbu x dan y dari semua titik dan juga radius direpresentasikan dengan bilangan bulat.

Keluaran berupa string yang menyatakan posisi titik "Titik di dalam lingkaran 1 dan 2", "Titik di dalam lingkaran 1", "Titik di dalam lingkaran 2", atau "Titik di luar ling ran 1 dan 2".

Fungsi untuk menghitung Jarak titik (a, b) dan (c, d) dimana rumus Jarak adalah:

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

dan juga fungsi untuk menentukan posisi sebuah titik sembarang berada di dalam suatu lingkaran atau tidak.

```
function jarak(a,b,c,d real) -> real
{Mengembalikan jarak antara titik (a,b) dan titik (c,d)}

function didalam (cx,cy, r, x, y real) -> boolean
{Mengembalikan true apabila titik (x,y) berada di dalam
lingkaran yang memiliki titik pusat (cx,cy) dan radius r}
```

Catatan: Lihat paket math dalam lampiran untuk menggunakan fungsi `math.Sqrt()` untuk menghitung akar kuadrat.

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var cx, cxx, cy, cyy, r, rr, x, y float64

    fmt.Println("Lingkaran 1")
    fmt.Print("(cx) (cy) (r) : ")
    fmt.Scanln(&cx, &cy, &r)

    fmt.Println("Lingkaran 2")
    fmt.Print("(cx) (cy) (r) : ")
    fmt.Scanln(&cxx, &cyy, &rr)

    fmt.Print("Input koordinat sembarang (x) (y) : ")
    fmt.Scanln(&x, &y)

    //mari kita cek posisi Lingkaran1 dan Lingkaran2
    posisi1 := inercircle(cx, cy, x, y, r)
    posisi2 := inercircle(cxx, cyy, x, y, rr)

    if posisi1 && posisi2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if posisi1 {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if posisi2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}

func distance(a, b, c, d float64) float64 {
    //pengecekan Jarak
    kiri := (a - c) * (a - c)
    kanan := (b - d) * (b - d)
    return math.Sqrt(kiri + kanan)
    //math.Sqrt digunakan untuk pemanggilan akar
}

func inercircle(cx, cy, x, y, r float64) bool {
```

```

    //pengecekan apakah x, y berada dalam lingkaran
    return distance(cx, cy, x, y) <= r
}

```

Screenshoot Output

```

PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3\main.go"
Lingkaran 1
(cx)(cy)(r) : 1 1 5
Lingkaran 2
(cx)(cy)(r) : 8 8 4
Input koordinat sembarang (x)(y) : 2 2
Titik di dalam lingkaran 1
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3\main.go"
Lingkaran 1
(cx)(cy)(r) : 1 2 3
Lingkaran 2
(cx)(cy)(r) : 4 5 6
Input koordinat sembarang (x)(y) : 7 8
Titik di dalam lingkaran 2
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3\main.go"
Lingkaran 1
(cx)(cy)(r) : 5 10 15
Lingkaran 2
(cx)(cy)(r) : -15 4 20
Input koordinat sembarang (x)(y) : 0 0
Titik di dalam lingkaran 1 dan 2
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3> go run "d:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3\main.go"
Lingkaran 1
(cx)(cy)(r) : 1 1 5
Lingkaran 2
(cx)(cy)(r) : 8 8 4
Input koordinat sembarang (x)(y) : 15 20
Titik di luar lingkaran 1 dan 2
PS D:\AfraLintang\Pratikum Alpro 2\Laprak 3\Unguided3>

```

Deskripsi Program

Program ini digunakan untuk melihat apakah titik koordinat sembarang yang diinputkan berada pada suatu lingkaran yang dibuat oleh user, Program ini memiliki beberapa fungsi yang cukup kompleks dengan cara menambahkan import "math", math digunakan untuk perhitungan matematis pada bahasa golang. Program ini memiliki beberapa fungsi yaitu :

- func distance(a, b, c, d float64) Digunakan untuk menghitung jarak antara inputan (cx, cy, x, y) dengan kiri := (a - c) * (a - c); kanan := (b - d) * (b - d); return math.Sqrt(kiri + kanan) "math.Sqrt" digunakan untuk menghitung akar
- func inercircle(cx, cy, x, y, r float64) Digunakan untuk mengecek apakah func distance(a, b, c, d float64) True atau false dengan cara membandingkan dengan variable r

Algoritma dari program ini adalah :

- //lingkarang pertama
- Input (cx, cy, r)
- Input (x,y)
- //lingkarang kedua
- Input (cxx, cyy, rr)
- Input (x,y)
- posisi1 := inercircle(cx, cy, x, y, r)
- posisi2 := inercircle(cxx, cyy, x, y, rr)
- posisi1, dan posisi2 digunakan untuk mengecek apakah kedua lingkaran *true* atau *false* dan menghitung jarak antara (cx, cy, x, y)

- Kemudian program akan mengklasifikasi apakah
 - o if posisi1 && posisi2 maka "Titik di dalam lingkaran 1 dan 2"
 - o if posisi1 maka "Titik di dalam lingkaran 1"
 - o if posisi2 maka "Titik di dalam lingkaran 2"
 - o jika tidak ketiganya maka "Titik di luar lingkaran 1 dan 2"

IV. DAFTAR PUSTAKA

N. Agung, "Golang Fungsi - Dasar Pemrograman Golang," *Dasar Pemrograman Golang*, 2024.
<https://dasarpemrogramangolang.novalagung.com/A-fungsi.html>. diakses pada: 11 Oktober 2024.