

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 3  
FUNGSI**



**Disusun Oleh :**

**Aby Hakim Al Yasiry Faozi/2311102208**

**IF-11-06**

**Dosen Pengampu :**

**Abednego Dwi Septiadi**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## **I. DASAR TEORI**

### Dasar Teori

fungsi dalam pemrograman adalah konsep yang penting dan umum digunakan untuk meningkatkan keterbacaan, pengelolaan, dan pemeliharaan kode.

#### A. Pengertian Fungsi

1. **Definisi:** Fungsi adalah blok kode yang dirancang untuk melakukan tugas tertentu. Fungsi dapat menerima input dalam bentuk argumen dan dapat mengembalikan output.
2. **Tujuan:** Fungsi digunakan untuk menghindari pengulangan kode, meningkatkan modularitas, dan mempermudah debugging serta pengujian.

#### B. Komponen Fungsi

1. **Nama Fungsi:** Identifikasi unik yang digunakan untuk memanggil fungsi.
2. **Parameter/Argumen:** Input yang diterima oleh fungsi. Parameter didefinisikan saat fungsi dideklarasikan, sedangkan argumen adalah nilai yang diberikan saat fungsi dipanggil.
3. **Badan Fungsi:** Blok kode yang berisi instruksi yang akan dieksekusi ketika fungsi dipanggil.
4. **Nilai Kembali (Return Value):** Nilai yang dikembalikan oleh fungsi setelah eksekusi selesai. Tidak semua fungsi harus mengembalikan nilai.

#### C. Jenis Fungsi

1. **Fungsi Prosedural:** Fungsi yang melakukan tugas tetapi tidak mengembalikan nilai.
2. **Fungsi yang Mengembalikan Nilai:** Fungsi yang melakukan tugas dan mengembalikan hasil.

#### D. Keuntungan Penggunaan Fungsi

1. **Reusabilitas:** Fungsi dapat dipanggil berkali-kali dari berbagai bagian program, mengurangi duplikasi kode.
2. **Modularitas:** Membagi program menjadi bagian-bagian yang lebih kecil dan terpisah, membuatnya lebih mudah untuk dikelola dan dipahami.
3. **Abstraksi:** Menyembunyikan detail implementasi dan hanya mengekspose antarmuka yang diperlukan, sehingga memudahkan

pengguna untuk menggunakan fungsi tanpa mengetahui detail internalnya.

4. Pemeliharaan: Memudahkan pemeliharaan dan pembaruan kode, karena perubahan dalam fungsi tidak mempengaruhi bagian lain dari program selama antarmuka fungsi tetap sama.

#### E. Implementasi Fungsi

1. Deklarasi: Menentukan nama fungsi, parameter, dan tipe nilai kembali (jika ada).
2. Definisi: Menyediakan implementasi dari fungsi, termasuk logika yang akan dijalankan.
3. Pemanggilan: Menjalankan fungsi dengan memberikan argumen yang diperlukan.

## II. GUIDED

### 1. Soal Studi Case

#### Sourcecode

```
package main

import "fmt"

var a, b, c, d int
// Fungsi untuk menghitung faktorial dari bilangan n
func faktorial(n int) int {
    hasil := 1
    // Loop untuk menghitung faktorial, dimulai dari 1
    hingga n
    for i := 1; i <= n; i++ {
        hasil = hasil * i
    }
    return hasil
}

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

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

func main() {

    fmt.Print("Masukkan input = ")
    fmt.Scan(&a, &b, &c, &d)

    if a >= c && b >= d {
        // Baris pertama: Permutasi dan Kombinasi a
        terhadap c
        fmt.Printf("%d, %d\n", permutasi(a, c),
            kombinasi(a, c))
        // Baris kedua: Permutasi dan Kombinasi b
        terhadap d
        fmt.Printf("%d, %d\n", permutasi(b, d),
            kombinasi(b, d))
    } else {
        fmt.Println("Syarat tidak terpenuhi: a harus >=
            c dan b harus >= d")
    }
}
```

## Screenshoot Output

```
\Semester 3\prak alpro 2\modul3> go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3\guided1\guided1.go"
Masukkan input = 9 7 1 2
9, 9
42, 21
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3>
```

## Deskripsi Program

### Algoritma

1. Inisialisasi Variabel:
  - Deklarasikan empat variabel integer a, b, c, dan d untuk menyimpan input pengguna.
2. Fungsi Faktorial:
  - Buat fungsi faktorial(n) yang menghitung faktorial dari n.
  - Inisialisasi variabel hasil dengan nilai 1.
  - Gunakan loop untuk mengalikan hasil dengan setiap bilangan dari 1 hingga n.
  - Kembalikan nilai hasil sebagai hasil faktorial.
3. Fungsi Permutasi:
  - Buat fungsi permutasi(n, r) untuk menghitung permutasi (  $P(n, r)$  ).
  - Gunakan rumus (  $P(n, r) = \frac{n!}{(n-r)!}$  ).
  - Panggil fungsi faktorial untuk menghitung n! dan (n-r)!.
  - Kembalikan hasil pembagian dari kedua faktorial tersebut.
4. Fungsi Kombinasi:
  - Buat fungsi kombinasi(n, r) untuk menghitung kombinasi (  $C(n, r)$  ).
  - Gunakan rumus (  $C(n, r) = \frac{n!}{r! \times (n-r)!}$  ).
  - Panggil fungsi faktorial untuk menghitung n!, r!, dan (n-r)!.
  - Kembalikan hasil pembagian dari n! dengan hasil perkalian r! dan (n-r)!.
5. Proses Utama (Main Function):
  - Minta pengguna untuk memasukkan empat nilai integer yang akan disimpan dalam a, b, c, dan d.
  - Periksa apakah  $a \geq c$  dan  $b \geq d$  untuk memastikan bahwa perhitungan permutasi dan kombinasi valid.
  - Jika syarat terpenuhi:
    - Hitung dan cetak permutasi dan kombinasi untuk pasangan (a, c) menggunakan fungsi permutasi dan kombinasi.
    - Hitung dan cetak permutasi dan kombinasi untuk pasangan (b, d).

- Jika syarat tidak terpenuhi, cetak pesan bahwa syarat tidak terpenuhi.

#### 6. Output:

- Jika syarat terpenuhi, keluarkan hasil perhitungan permutasi dan kombinasi untuk kedua pasangan.
- Jika syarat tidak terpenuhi, beri tahu pengguna bahwa input tidak valid sesuai syarat yang ditetapkan.

## 2. Soal Studi Case

### 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

```
\Semester 3\prak alpro 2\modul3> go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3\guided2\guided2.go"
Masukkan panjang balok: 12
Masukkan lebar balok: 7
Masukkan tinggi balok: 3
Volume dari balok tersebut adalah: 252.00
Luas dari balok tersebut adalah: 282.00
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3>
```

## Deskripsi Program

### Algoritma:

1. Inisialisasi:
  - Import package fmt untuk menangani input dan output.
2. Definisi Fungsi:
  - Fungsi volumeBalok(p, l, t):
    - Input: p (panjang), l (lebar), t (tinggi) sebagai float64.
    - Proses: Hitung volume menggunakan rumus ( $\text{volume} = p \times l \times t$ ).
    - Output: Kembalikan nilai volume sebagai float64.
  - Fungsi luasBalok(p, l, t):
    - Input: p (panjang), l (lebar), t (tinggi) sebagai float64.
    - Proses: Hitung luas permukaan menggunakan rumus ( $\text{luas} = 2 \times ((p \times l) + (p \times t) + (l \times t))$ ).
    - Output: Kembalikan nilai luas permukaan sebagai float64.
3. Fungsi main():
  - Deklarasi Variabel:
    - Deklarasikan variabel p, l, dan t sebagai float64 untuk menyimpan panjang, lebar, dan tinggi balok.
  - Input Pengguna:
    - Tampilkan pesan untuk meminta pengguna memasukkan nilai panjang balok.
    - Gunakan fmt.Scan untuk membaca input pengguna dan simpan dalam variabel p.
    - Tampilkan pesan untuk meminta pengguna memasukkan nilai lebar balok.
    - Gunakan fmt.Scan untuk membaca input pengguna dan simpan dalam variabel l.
    - Tampilkan pesan untuk meminta pengguna memasukkan nilai tinggi balok.
    - Gunakan fmt.Scan untuk membaca input pengguna dan simpan dalam variabel t.

- Hitung dan Tampilkan Hasil:
  - Panggil fungsi `volumeBalok(p, l, t)` untuk menghitung volume balok dan simpan hasilnya.
  - Panggil fungsi `luasBalok(p, l, t)` untuk menghitung luas permukaan balok dan simpan hasilnya.
  - Gunakan `fmt.Printf` untuk mencetak hasil volume dengan format dua angka desimal.
  - Gunakan `fmt.Printf` untuk mencetak hasil luas permukaan dengan format dua angka desimal.
- 4. Akhir Program:
  - Program selesai setelah menampilkan volume dan luas permukaan balok berdasarkan input pengguna.



### III. UNGUIDED

#### 1. Soal Studi Case

##### Sourcecode

```
package main

import (
    "fmt"
    "math/big"
)

func factorial(n int64) *big.Int {
    result := big.NewInt(1)
    for i := int64(2); i <= n; i++ {
        result.Mul(result, big.NewInt(i))
    }
    return result
}

func permutation(n, r int64) *big.Int {
    return new(big.Int).Div(factorial(n), factorial(n-r))
}

func combination(n, r int64) *big.Int {
    return new(big.Int).Div(factorial(n),
        new(big.Int).Mul(factorial(r), factorial(n-r)))
}

func main() {
    var a, b, c, d int64
    fmt.Print("Masukkan empat bilangan (a, b, c, d)
    dipisahkan oleh spasi: ")
    _, err := fmt.Scan(&a, &b, &c, &d)

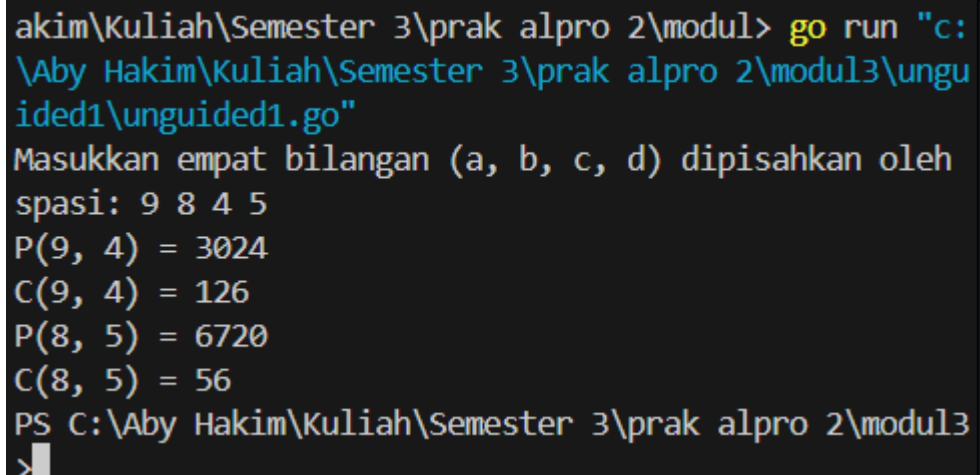
    if err != nil {
        fmt.Println("Input tidak valid. Silakan masukkan
        empat bilangan.")
        return
    }

    if a < c || b < d {
        fmt.Println("Input tidak valid. Pastikan a >= c
        dan b >= d.")
        return
    }

    fmt.Printf("P(%d, %d) = %v\n", a, c, permutation(a,
    c))
    fmt.Printf("C(%d, %d) = %v\n", a, c, combination(a,
    c))
}
```

```
    fmt.Printf("P(%d, %d) = %v\n", b, d, permutation(b,
d))
    fmt.Printf("C(%d, %d) = %v\n", b, d, combination(b,
d))
}
```

### Screenshoot Output



```
akim\Kuliah\Semester 3\prak alpro 2\modul> go run "c:
\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3\ungu
ided1\unguided1.go"
Masukkan empat bilangan (a, b, c, d) dipisahkan oleh
spasi: 9 8 4 5
P(9, 4) = 3024
C(9, 4) = 126
P(8, 5) = 6720
C(8, 5) = 56
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3
>
```

### Deskripsi Program

Algoritma:

#### 1. Library yang Digunakan

- `fmt`: Digunakan untuk input dan output.
- `math/big`: Digunakan untuk menangani perhitungan bilangan besar menggunakan tipe data `big.Int`, karena hasil dari faktorial, permutasi, atau kombinasi bisa sangat besar.

#### 2. Fungsi factorial

- Fungsi ini menerima satu parameter `n` (`int64`) dan menghitung faktorial dari `n`.
- Mengembalikan hasil faktorial dalam bentuk `*big.Int`.
- Menggunakan perulangan `for` untuk mengalikan setiap bilangan dari 2 hingga `n`.

#### 3. Fungsi permutation

- Fungsi ini menerima dua parameter `n` dan `r`, lalu menghitung permutasi  $P(n, r)$ .
- Fungsi factorial digunakan untuk menghitung  $n!$  dan  $(n-r)!$ .

#### 4. Fungsi combination

- Fungsi ini juga menerima dua parameter  $n$  dan  $r$ , lalu menghitung kombinasi  $C(n, r)$ .
- Fungsi ini menggunakan factorial untuk menghitung  $n!$ ,  $r!$ , dan  $(n-r)!$ .

#### 5. Fungsi main

- Fungsi utama program yang menjalankan keseluruhan logika:
- Mengambil empat bilangan bulat  $a$ ,  $b$ ,  $c$ , dan  $d$  dari pengguna.
- Memeriksa apakah input valid dengan memastikan bahwa  $a \geq c$  dan  $b \geq d$ .
- Jika input tidak valid, program akan mencetak pesan kesalahan dan keluar.
- Jika input valid, program akan menghitung dan mencetak hasil:
- Permutasi dan kombinasi dari  $(a, c)$ .
- Permutasi dan kombinasi dari  $(b, d)$ .

## 2. Soal Studi Case

### Sourcecode

```
package main

import (
    "fmt"
)

func f(x int) int {
    return x * x
}

func g(x int) int {
    return x - 2
}

func h(x int) int {
    return x + 1
}

func fogoh(x int) int {
    return f(g(h(x)))
}

func gohof(x int) int {
    return g(h(f(x)))
}

func hofog(x int) int {
    return h(f(g(x)))
}

func main() {
    var a, b, c int
    fmt.Print("Masukkan tiga bilangan (pisahkan dengan spasi): ")
    fmt.Scan(&a, &b, &c)

    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

```
\Semester 3\prak alpro 2\modul3> go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3\unguided2\unguided2.go"
Masukkan tiga bilangan (pisahkan dengan spasi): 7 2 10

fogoh(7) = 36
gohof(2) = 3
hofog(10) = 65
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3>
```

## Deskripsi Program

Algoritma:

### 1. Fungsi Dasar:

- $f(x)$ : Menghitung kuadrat dari  $x$  (mengembalikan  $x * x$ ).
- $g(x)$ : Mengurangi  $x$  dengan 2 (mengembalikan  $x - 2$ ).
- $h(x)$ : Menambah  $x$  dengan 1 (mengembalikan  $x + 1$ ).

### 2. Fungsi Komposisi:

- $\text{fogoh}(x)$ : Menggabungkan  $f$ ,  $g$ , dan  $h$  dalam urutan  $f(g(h(x)))$ . Artinya,  $h(x)$  dieksekusi terlebih dahulu, lalu hasilnya dimasukkan ke  $g$ , dan akhirnya ke  $f$ .
- $\text{gohof}(x)$ : Menggabungkan  $g$ ,  $h$ , dan  $f$  dalam urutan  $g(h(f(x)))$ .
- $\text{hofog}(x)$ : Menggabungkan  $h$ ,  $f$ , dan  $g$  dalam urutan  $h(f(g(x)))$ .

### 3. Fungsi main:

- Program meminta pengguna untuk memasukkan tiga angka ( $a$ ,  $b$ ,  $c$ ).
- Mencetak hasil dari  $\text{fogoh}(a)$ ,  $\text{gohof}(b)$ , dan  $\text{hofog}(c)$ .

### 3. Soal Studi Case

#### Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func distance(x1, y1, x2, y2 float64) float64 {
    return math.Sqrt((x1-x2)*(x1-x2) + (y1-y2)*(y1-y2))
}

func isInsideCircle(cx, cy, r, x, y float64) bool {
    return distance(cx, cy, x, y) <= r
}

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

    fmt.Println("Masukkan koordinat pusat dan radius lingkaran 1:")
    fmt.Scan(&cx1, &cy1, &r1)

    fmt.Println("Masukkan koordinat pusat dan radius lingkaran 2:")
    fmt.Scan(&cx2, &cy2, &r2)

    fmt.Println("Masukkan koordinat titik sembarang:")
    fmt.Scan(&x, &y)

    insideCircle1 := isInsideCircle(cx1, cy1, r1, x, y)
    insideCircle2 := isInsideCircle(cx2, cy2, r2, x, y)

    if insideCircle1 && insideCircle2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if insideCircle1 {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if insideCircle2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}
```

## Screenshoot Output

```
\Semester 3\prak alpro 2\modul3> go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3\unguided3\unguided3.go"
Masukkan koordinat pusat dan radius lingkaran 1:
5 10 15
Masukkan koordinat pusat dan radius lingkaran 2:
-15 4 20
Masukkan koordinat titik sembarang:
0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul3> |
```

## Deskripsi Program

Algoritma:

1. Fungsi distance:

- Menghitung jarak antara dua titik (x1, y1) dan (x2, y2)

2. Fungsi isInsideCircle:

- Mengecek apakah titik (x, y) berada di dalam lingkaran dengan pusat (cx, cy) dan radius r.
- Menggunakan fungsi distance untuk menghitung jarak antara pusat lingkaran dan titik (x, y), lalu mengecek apakah jarak tersebut kurang dari atau sama dengan radius r.

3. Fungsi main:

- Meminta pengguna untuk memasukkan koordinat pusat dan radius dari dua lingkaran, serta koordinat dari titik yang akan dicek.
- Menggunakan isInsideCircle untuk menentukan apakah titik tersebut berada di dalam lingkaran 1, lingkaran 2, keduanya, atau di luar keduanya.
- Menampilkan hasil sesuai kondisi yang ditemukan:
  - "Titik di dalam lingkaran 1 dan 2"
  - "Titik di dalam lingkaran 1"
  - "Titik di dalam lingkaran 2"
  - "Titik di luar lingkaran 1 dan 2"