

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL III

FUNGSI



**Disusun Oleh :
RORY REFALDO SINULINGGA(2311102323)**

Kelas: IF-06

Dosen Pengampu :

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO**

2024

I. DASAR TEORI

Dasar Teori

Dalam pemrograman, fungsi adalah blok kode yang ditanamkan untuk tujuan tertentu. Fungsi digunakan oleh pemrogram untuk menyusun kode menjadi potongan-potongan kecil yang dapat dikelola. Fungsi memaafkan membuat kode menjadi modular sehingga setiap bagian kode bertanggung jawab untuk hal tertentu saja. Hal ini tidak hanya meningkatkan kemampuan pengguna untuk memahami kode, namun juga memudahkan pemeliharaan dan debugging.

Suatu fungsi biasanya didefinisikan dengan nama yang berbeda dan dapat diakses di mana saja dalam program. Ketika suatu fungsi dipanggil, program mengeksekusi pernyataan yang ada dalam fungsi tersebut. Suatu fungsi dapat mengambil masukan berupa parameter dan keluaran juga dapat diamati. Parameter memungkinkan suatu fungsi mengambil beberapa nilai dari luar, sedangkan nilai kembalian memungkinkan fungsi meneruskan hasilnya kembali ke bagian program yang memintanya. Dengan demikian, hal ini memungkinkan fungsi menjadi sangat mudah beradaptasi dan dilakukan di lokasi berbeda dalam program.

Selain itu, penggunaan fungsi dapat mengoptimalkan kode dan menghilangkan duplikasi kode. Dengan menulis fungsi yang dapat digunakan kembali, pengembang tidak perlu menulis ulang kode yang sama tanpa henti. Hal ini tidak hanya membantu menghemat waktu tetapi juga membantu mengurangi kesalahan. Fungsi juga memungkinkan pengembang untuk menyembunyikan rincian bagaimana sesuatu diimplementasikan, memungkinkan mereka berkonsentrasi pada logika tingkat tinggi dari program mereka. Oleh karena itu, dapat dikatakan bahwa fungsi adalah elemen penting dalam pemrograman, karena fungsi membantu menghasilkan kode yang lebih rapi, terorganisir, dan efisien.

.

II. GUIDED

1. Berisi source code dan output dari kegiatan praktikum yang telah dilaksanakan. Source Code diberi penjelasan maka akan menjadi nilai ++

Soal Studi Case

Sourcecode

```
package main

import "fmt"

func main() {
    var a, b int
    fmt.Scan(&a, &b)

    if a == b {
        fmt.Println("a and b are equal")
    } else {
        fmt.Println("a and b are not equal")
    }

    fmt.Println(permutasi(a, b))
    fmt.Println(permutasi(b, a))
}

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

func permutasi(n, r int) int {
    return faktorial(n) / faktorial(n-r)
}
```

Screenshoot Output

```
PS D:\src> go run "d:\src\modul2\hipotenusa\hipo.go"

1
2
a and b are not equal
1
2
PS D:\src> 
```

Deskripsi Program

Program dimulai dengan menanyakan dua masukan bilangan bulat a dan b kepada pengguna menggunakan pernyataan `fmt.Scan(&a, &b)`.

Kemudian program memeriksa apakah kedua nilai tersebut sama.

Misalnya, jika a sama dengan b maka program akan menuliskan “a dan b sama”.

Sedangkan jika a tidak sama dengan b maka program akan menyatakan “a dan b tidak sama”.

Setelah itu, program menjalankan fungsi `permutasi(a,b)`, yang menghitung a menjadi b, dan b menjadi a.

Kedua hasil permutasi ditampilkan di layar.

Fungsi ini menghitung faktorial suatu bilangan tertentu n dengan melakukan perulangan dari 1 ke n, mengalikan setiap nilai iterasi secara berurutan dan menyimpannya dalam variabel hasil kali.

Di akhir perulangan, nilai faktorial dikembalikan.

Fungsi ini menggunakan rumus substitusi: $P_{n,r} = \frac{n!}{(n-r)!}$ faktorial atas n dikurangi r faktorial.

Iklan harus menyertakan gambar.

Fungsi `fastial(n)` dipanggil untuk menghitung n faktorial, dan hasilnya adalah n r faktorial dibagi n faktorial.

III. UNGUIDED

1.

Sourcecode

```
package main

import "fmt"

func f(x int) int {
    return x * x * 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.Scan(&a, &b, &c)

    fmt.Println(fogoh(a))
    fmt.Println(gohof(b))
    fmt.Println(hofog(c))
}
```

Screenshoot Output

```
PS D:\src> go run "d:\src\modul2\hipotenusa\hipo.go"
2 3 4
1
30
17
PS D:\src> 
```

Deskripsi Program

Program ini mendefinisikan beberapa fungsi matematika sederhana (f, g, h) dan beberapa komposisi fungsi (fogoh, gohof, hofog). Program menerima tiga input integer dari pengguna (a, b, dan c) dan menampilkan hasil dari komposisi fungsi-fungsi tersebut.

3.

```
package main

import (
    "fmt"
    "math"
)

func jarak_aldo_323(x1_aldo_323, y1_aldo_323, x2_aldo_323,
y2_aldo_323 float64) float64 {
    return math.Sqrt(math.Pow(x2_aldo_323-x1_aldo_323, 2) +
math.Pow(y2_aldo_323-y1_aldo_323, 2))
}

func diDalamLingkaran_aldo_323(cx_aldo_323, cy_aldo_323,
r_aldo_323, x_aldo_323, y_aldo_323 float64) bool {
    return jarak_aldo_323(cx_aldo_323, cy_aldo_323, x_aldo_323,
y_aldo_323) <= r_aldo_323
}

func main() {
    var cx1_aldo_323, cy1_aldo_323, r1_aldo_323, cx2_aldo_323,
cy2_aldo_323, r2_aldo_323, x_aldo_323, y_aldo_323 float64
    fmt.Scan(&cx1_aldo_323, &cy1_aldo_323, &r1_aldo_323,
&cx2_aldo_323, &cy2_aldo_323, &r2_aldo_323, &x_aldo_323,
&y_aldo_323)

    dalamLingkaran1_aldo_323 :=
diDalamLingkaran_aldo_323(cx1_aldo_323, cy1_aldo_323,
r1_aldo_323, x_aldo_323, y_aldo_323)
    dalamLingkaran2_aldo_323 :=
diDalamLingkaran_aldo_323(cx2_aldo_323, cy2_aldo_323,
r2_aldo_323, x_aldo_323, y_aldo_323)

    if dalamLingkaran1_aldo_323 && dalamLingkaran2_aldo_323 {
```

Screenshot output

```
PS D:\src> go run "d:\src\modul2\hipotenusa\hipo.go"
0 0 5 10 10 5 3 3
Titik di dalam lingkaran 1
PS D:\src> |
```

Deskripsi Program

Program ini mendefinisikan beberapa fungsi matematika sederhana (f, g, h) dan beberapa komposisi fungsi (fogoh, gohof, hofog). Program menerima tiga input integer dari pengguna (a, b, dan c) dan menampilkan hasil dari komposisi fungsi-fungsi tersebut.