

**LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK**

MODUL III

FUNGSI



Oleh:

NAMA : AJI TRI PRASETYO

NIM : 2311102064

KELAS : IF 11 02

**S1 TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

2024

I. DASAR TEORI

Fungsi adalah serangkaian instruksi yang menghasilkan nilai dengan memetakan input ke output. Karena itu, fungsi selalu memberikan hasil. Suatu subprogram disebut fungsi jika memenuhi dua syarat: mendeklarasikan tipe data yang dikembalikan dan menggunakan kata kunci `return` dalam tubuh subprogram. Fungsi digunakan ketika diperlukan sebuah nilai, seperti untuk menetapkan nilai ke variabel, menjadi bagian dari ekspresi, atau sebagai argumen subprogram. Oleh sebab itu, nama fungsi sebaiknya mencerminkan hasil yang dikembalikan, seperti dengan menggunakan kata benda atau kata sifat, contohnya: median, rerata, nilaiTerbesar, ditemukan, selesai, dan sebagainya.

- Deklarasi fungsi

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

Notasi Algoritma
<pre>function <nama kamus {deklarasi function>(<params>)-><type> ... algoritma {badan algoritma fungsi} ... return <value/variabel> endfunction</pre>
Notasi dalam bahasa Go
<pre>func <nama function>(<params>) <type> { /* deklarasi variabel lokal dari fungsi */ ... /* badan algoritma fungsi */ ... return <value/variabel> }</pre>

Cara Pemanggilan Function

Cara memanggil fungsi mirip dengan memanggil prosedur, yaitu dengan menuliskan nama fungsi dan argumen yang sesuai dengan parameternya. Namun, perbedaannya adalah fungsi dapat ditetapkan ke sebuah variabel, digunakan dalam ekspresi, atau menjadi argumen dalam subprogram lain.

	Notasi Algoritma
1	<pre>program kamus <small>Canal-P-prosedur</small> r,t : integer v1,v2 : real algoritma r <- 5; t <- 10 v1 <- volumeTabung(r,t) v2 <- volumeTabung(r,t) volumeTabung(15,t) {cara #1} output(volumeTabung(14,100)) {cara #2} endprogram {cara #3}</pre>
	Notasi dalam bahasa Go
	<pre>func main() { var r,t int var v1,v2 float64 5 10 v1 = volumeTabung(r,t) // cara #1 v2 = volumeTabung(r,t) + volumeTabung(15,t) fmt.Println(volumeTabung(14,100)) // cara #2 }</pre>

Pada contoh pemanggilan fungsi di atas, tidak ada perbedaan antara pemanggilan fungsi di pseudocode dan Golang. Fungsi dapat diberikan ke variabel saat pemanggilan, dioperasikan sesuai tipe data yang dikembalikan, dan juga dapat langsung ditampilkan menggunakan perintah output atau print.

II. GUIDED

Source Code + Screenshot hasil program beserta penjelasan

SOAL

NO 1.

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 :

```
PS C:\Users\MyBook Z Series> go run "C:\Users\MYBOOK~1\AppData\Local\Temp\1\go-build1011101110\bin\permutasi.go"
5 10
30240
PS C:\Users\MyBook Z Series> █
```

Penjelasan :

Program tersebut menghitung permutasi dari dua bilangan yang diberikan oleh pengguna. Pertama, program menerima dua nilai integer, yaitu `a` dan `b`. Jika `a` lebih besar atau sama dengan `b`, program memanggil fungsi `permutasi(a, b)`; jika tidak, maka program memanggil `permutasi(b, a)`. Fungsi `permutasi` memanfaatkan fungsi `faktorial` untuk menghitung nilai permutasi menggunakan rumus berikut:

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

di mana faktorial dihitung dalam fungsi `faktorial`.

GUIDED

NO 2.

```
1 package main
2 import "fmt"
3
4 func factorial(n int) int {
5     if n == 0 || n == 1 {
6         return 1
7     }
8     result := 1
9     for i := 2; i <= n; i++ {
10        result *= i
11    }
12    return result
13 }
14
15 func permutation(n, r int) int {
16     return factorial(n) / factorial(n-r)
17 }
18 func combination(n, r int) int {
19     return factorial(n) / (factorial(r) * factorial(n-r))
20 }
21 func main() {
22     var a, b, c, d int
23     fmt.Println("Masukkan bilangan a, b, c, d (dengan spasi): ")
24     fmt.Scanf("%d %d %d %d", &a, &b, &c, &d)
25
26     if a >= c && b >= d {
27         permutasiAC := permutation(a, c)
28         kombinasiAC := combination(a, c)
29
30         permutasiBD := permutation(b, d)
31         kombinasiBD := combination(b, d)
32
33         fmt.Println("Permutasi(a, c) dan Kombinasi(a, c):", permutasiAC, kombinasiAC)
34         fmt.Println("Permutasi(b, d) dan Kombinasi(b, d):", permutasiBD, kombinasiBD)
35     } else {
36         fmt.Println("Syarat a >= c dan b >= d tidak terpenuhi.")
37     }
38 }
```

Output :

```
PS C:\Users\MyBook Z Series> go run "C:\Users\MYBOOK~1\AppData\Local\Temp\tempCodeRunnerFile.go"
Masukkan bilangan a, b, c, d (dengan spasi):
4 3 2 1
Permutasi(a, c) dan Kombinasi(a, c): 12 6
Permutasi(b, d) dan Kombinasi(b, d): 3 3
PS C:\Users\MyBook Z Series> go run "C:\Users\MYBOOK~1\AppData\Local\Temp\tempCodeRunnerFile.go"
Masukkan bilangan a, b, c, d (dengan spasi):
1 2 3
Syarat a >= c dan b >= d tidak terpenuhi.
PS C:\Users\MyBook Z Series> 
```

Penjelasan :

Program tersebut menghitung permutasi dan kombinasi dari dua pasang angka yang dimasukkan oleh pengguna. Fungsi `factorial` digunakan untuk menghitung faktorial, sementara fungsi `permutation` dan `combination` memanfaatkan faktorial tersebut untuk melakukan perhitungan permutasi dan kombinasi. Di dalam fungsi `main`, pengguna diminta memasukkan empat bilangan (a, b, c, d). Jika syarat `a >= c` dan `b >= d` terpenuhi, program akan menghitung dan menampilkan hasil permutasi dan kombinasi dari pasangan (a, c) dan (b, d). Jika syarat tidak terpenuhi, program akan memberikan pesan kesalahan.

III. UNGUIDED

Source Code + Screenshot hasil program beserta penjelasan

SOAL 3

[Lingkaran] Suatu lingkaran didefinisikan dengan koordinat titik pusat (cx, cy) dengan radius r. 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 lingkaran 1 dan 2".

Source code :

```
1 package main
2 import (
3     "fmt"
4     "math"
5 )
6
7 func main() {
8     var (
9         cx1, cy1, r1 int
10        cx2, cy2, r2 int
11        x, y          int
12    )
13
14    fmt.Scan(&cx1, &cy1, &r1, &cx2, &cy2, &r2, &x, &y)
15
16    switch {
17    case cekDalamLingkaran(x, y, cx1, cy1, r1) && cekDalamLingkaran(x, y, cx2, cy2, r2):
18        fmt.Println("Titik di dalam lingkaran 1 dan 2")
19    case cekDalamLingkaran(x, y, cx1, cy1, r1):
20        fmt.Println("Titik di dalam lingkaran 1")
21    case cekDalamLingkaran(x, y, cx2, cy2, r2):
22        fmt.Println("Titik di dalam lingkaran 2")
23    default:
24        fmt.Println("Titik di luar lingkaran 1 dan 2")
25    }
26 }
27
28 func cekDalamLingkaran(x, y, cx, cy, r int) bool {
29     jarak := math.Hypot(float64(x-cx), float64(y-cy))
30     return jarak <= float64(r)
31 }
32
```

Output :

```
PS C:\Users\MyBook Z Series> go run "C:\Users\MYBOOK~1\AppData\Local\Temp\tempCodeRunnerFile.go"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS C:\Users\MyBook Z Series> go run "C:\Users\MYBOOK~1\AppData\Local\Temp\tempCodeRunnerFile.go"
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS C:\Users\MyBook Z Series> go run "C:\Users\MYBOOK~1\AppData\Local\Temp\tempCodeRunnerFile.go"
5 10 15
-15 4 20
0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\MyBook Z Series> go run "C:\Users\MYBOOK~1\AppData\Local\Temp\tempCodeRunnerFile.go"
1 1 5
8 8 4
15 20
Titik di luar lingkaran 1 dan 2
PS C:\Users\MyBook Z Series> 
```

Penjelasan :

Program di atas berfungsi untuk menentukan apakah suatu titik berada di dalam salah satu, kedua, atau di luar dua lingkaran yang diberikan. Program menerima input koordinat pusat dan jari-jari dari dua lingkaran serta koordinat titik yang akan dicek. Fungsi `cekDalamLingkaran` menghitung jarak antara titik dan pusat lingkaran menggunakan rumus Euclidean (dengan `math.Hypot`), lalu membandingkan jarak tersebut dengan jari-jari lingkaran. Berdasarkan hasil pengecekan, program akan mencetak apakah titik tersebut berada di dalam lingkaran pertama, kedua, keduanya, atau di luar keduanya.

