

**LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK**

MODUL III

FUNGSI



Oleh:

NAMA : ARVAN MURBIYANTO

NIM : 2311102074

KELAS : IF 11 02

**S1 TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

2024

I. DASAR TEORI

Fungsi adalah serangkaian instruksi yang bekerja untuk menghasilkan sebuah nilai tertentu. Fungsi berperan dalam memetakan input (argumen) menjadi suatu nilai hasil. Dengan kata lain, fungsi menerima data sebagai masukan, memprosesnya, dan mengembalikan hasil olahan tersebut. Fungsi sangat berguna dalam pemrograman karena memungkinkan pemecahan kode menjadi bagian-bagian kecil (modularisasi), yang membuat kode lebih terstruktur dan memudahkan penggunaan ulang. Sebuah subprogram dianggap sebagai fungsi jika memenuhi dua syarat berikut:

1. Mengandung deklarasi tipe data dari nilai yang akan dikembalikan.
2. Terdapat kata kunci *return* dalam tubuh subprogram untuk mengembalikan hasil.

- 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

Sama halnya dengan prosedur, pemanggilan fungsi cukup dilakukan dengan penulisan nama fungsi beserta argumen yang diminta oleh parameter dari fungsi. Perbedaannya dengan prosedur adalah fungsi bisa di-assign ke suatu variabel, menjadi bagian dari ekspresi, dan argumen dari suatu subprogram.

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

Pada contoh pemanggilan fungsi di atas terlihat tidak ada perbedaan pada saat pemanggilan fungsi pada pseudocode ataupun Golang. Di sini terlihat fungsi bisa di-assign ke suatu variabel pada saat pemanggilan, bisa dioperasikan sesuai dengan tipe data yang dikembalikan, dan juga bisa langsung ditampilkan dengan perintah output ataupun print.

II. GUIDED

Source Code + Screenshot hasil program beserta penjelasan

SOAL

NO 1.

Source code:

```
package main

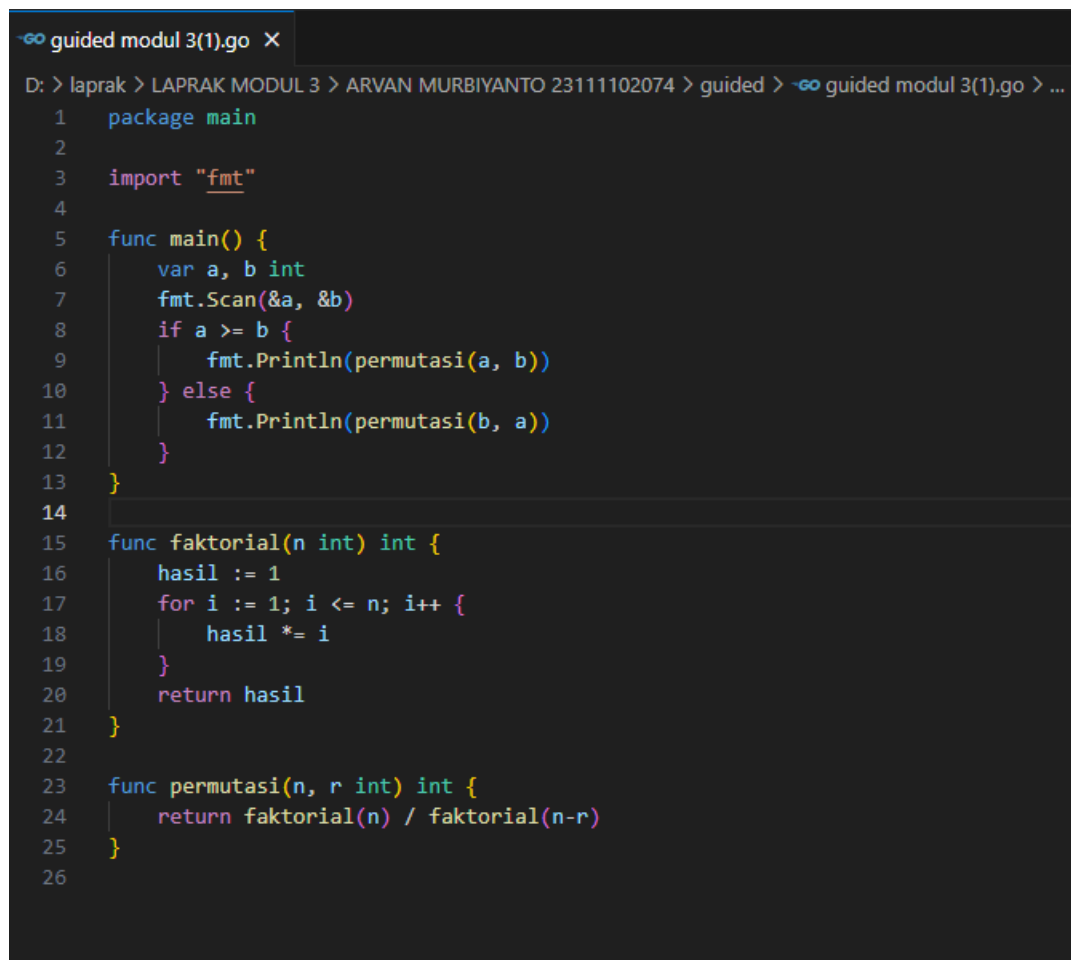
import "fmt"

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
}

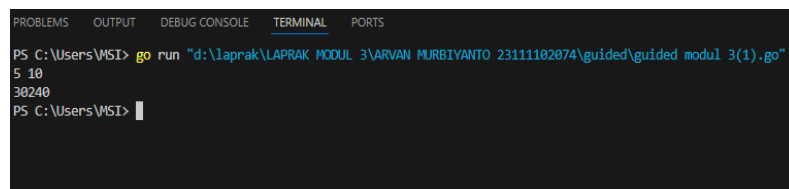
func permutasi(n, r int) int {
    return faktorial(n) / faktorial((n - r))
}
```

Screenshot :



```
guided modul 3(1).go X
D: > laprak > LAPRAK MODUL 3 > ARVAN MURBIYANTO 23111102074 > guided > guided modul 3(1).go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      var a, b int
7      fmt.Scan(&a, &b)
8      if a >= b {
9          fmt.Println(permutasi(a, b))
10     } else {
11         fmt.Println(permutasi(b, a))
12     }
13 }
14
15 func faktorial(n int) int {
16     hasil := 1
17     for i := 1; i <= n; i++ {
18         hasil *= i
19     }
20     return hasil
21 }
22
23 func permutasi(n, r int) int {
24     return faktorial(n) / faktorial(n-r)
25 }
26
```

Output :



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\WSI> go run "d:\laprak\LAPRAK MODUL 3\ARVAN MURBIYANTO 23111102074\guided\guided modul 3(1).go"
5 10
30240
PS C:\Users\WSI>
```

Deklarasi :

Program ini dirancang untuk menghitung permutasi dari dua bilangan bulat yang diinput oleh user dengan memeriksa apakah a lebih besar atau sama dengan b. Jika benar, program akan menghitung permutasi dengan a sebagai n dan b sebagai r. Jika tidak, program akan menghitung permutasi dengan b sebagai n dan a sebagai r.

NO 2.

Source code:

```
package main

import "fmt"

func factorial(n int) int {
    if n == 0 || n == 1 {
        return 1
    }
    result := 1
    for i := 2; i <= n; i++ {
        result *= i
    }
    return result
}

func permutation(n, r int) int {
    return factorial(n) / factorial(n-r)
}

func combination(n, r int) int {
    return factorial(n) / (factorial(r) * factorial(n-r))
}

func main() {
    var a, b, c, d int
    fmt.Println("Masukkan bilangan a, b, c, d (dengan spasi):")
    fmt.Scanf("%d %d %d %d", &a, &b, &c, &d)

    if a >= c && b >= d {

        permutasiAC := permutation(a, c)
        kombinasiAC := combination(a, c)

        permutasiBD := permutation(b, d)
        kombinasiBD := combination(b, d)

        fmt.Println("Permutasi(a, c) dan Kombinasi(a, c):", permutasiAC, kombinasiAC)
        fmt.Println("Permutasi(b, d) dan Kombinasi(b, d):", permutasiBD, kombinasiBD)
    } else {
        fmt.Println("Syarat a >= c dan b >= d tidak terpenuhi.")
    }
}
```

Screenshot :

```
guided modul 3(1).go X
D: > laprak > LAPRAK MODUL 3 > ARVAN MURBIYANTO 23111102074 > guided > guided modul 3(1).go > main
1  package main
2
3  import "fmt"
4
5  func factorial(n int) int {
6      if n == 0 || n == 1 {
7          return 1
8      }
9      result := 1
10     for i := 2; i <= n; i++ {
11         result *= i
12     }
13     return result
14 }
15
16 func permutation(n, r int) int {
17     return factorial(n) / factorial(n-r)
18 }
19
20 func combination(n, r int) int {
21     return factorial(n) / (factorial(r) * factorial(n-r))
22 }
23
24 func main() {
25     var a, b, c, d int
26     fmt.Println("Masukkan bilangan a, b, c, d (dengan spasi): ")
27     fmt.Scanf("%d %d %d %d", &a, &b, &c, &d)
28
29     if a >= c && b >= d {
30         permutasiAC := permutation(a, c)
31         kombinasiAC := combination(a, c)
32
33         permutasiBD := permutation(b, d)
34         kombinasiBD := combination(b, d)
35
36         fmt.Println("Permutasi(a, c) dan Kombinasi(a, c):", permutasiAC, kombinasiAC)
37         fmt.Println("Permutasi(b, d) dan Kombinasi(b, d):", permutasiBD, kombinasiBD)
38     } else {
39         fmt.Println("Syarat a >= c dan b >= d tidak terpenuhi.")
40     }
41 }
42
```

Output :

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\MSI> go run "C:\Users\MSI\AppData\Local\Temp\tempCodeRunnerFile.go"
5 10
30240
PS C:\Users\MSI> go run "d:\laprak\LAPRAK MODUL 3\ARVAN MURBIYANTO 23111102074\guided\guided modul 3(1).go"
Masukkan bilangan a, b, c, d (dengan spasi):
5 4 2 3
Permutasi(a, c) dan Kombinasi(a, c): 20 10
Permutasi(b, d) dan Kombinasi(b, d): 24 4
PS C:\Users\MSI>
```

Deklarasi :

Program bertujuan untuk menghitung permutasi dan kombinasi dari dua pasang bilangan yang diinput oleh user. Program memeriksa apakah $a \geq c$ dan $b \geq d$. Jika kondisi ini terpenuhi, maka program akan melanjutkan untuk menghitung permutasi dan kombinasi. Jika kondisi tidak terpenuhi, program akan mencetak pesan bahwa syarat tidak terpenuhi.

III. UNGUIDED

Source Code + Screenshot hasil program beserta penjelasan

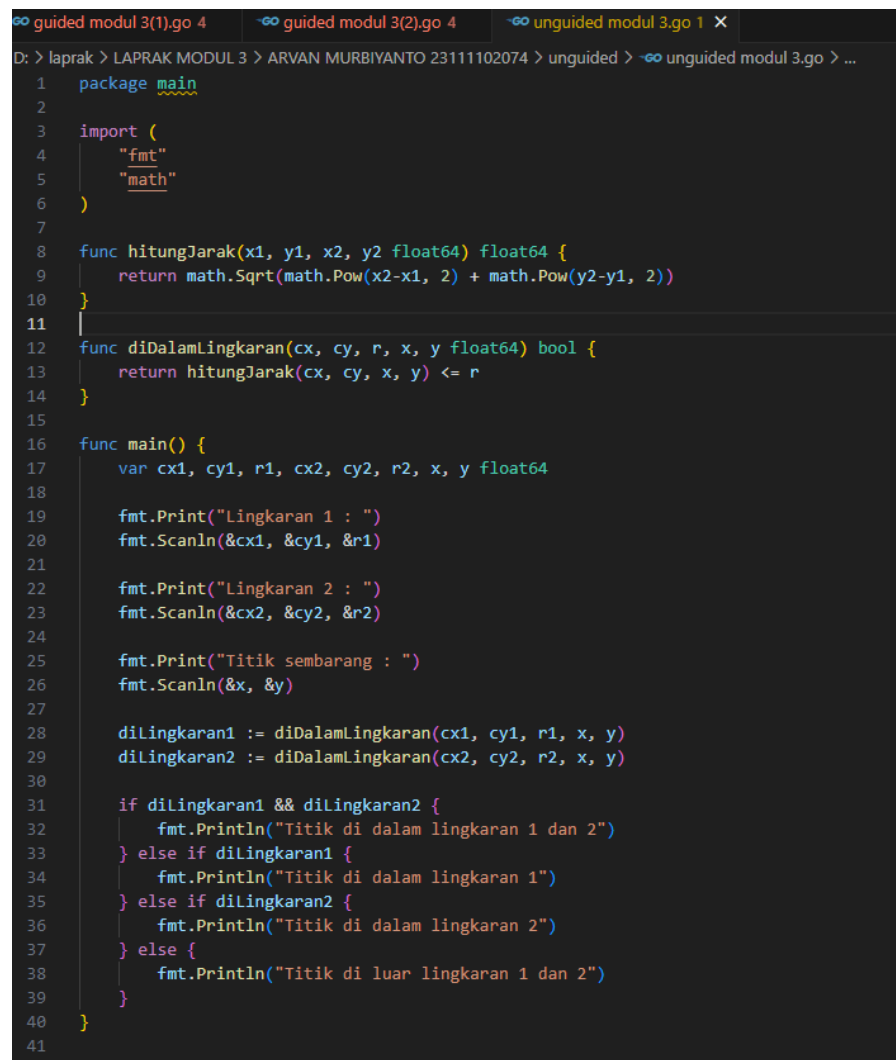
SOAL 3

[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.

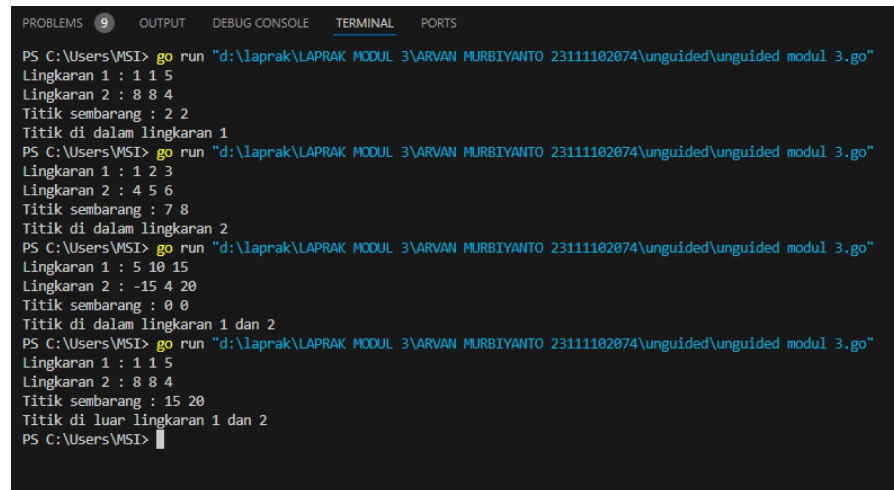
Keluaran berupa string yang menyatakan posisi titik "Titik di dalam lingkaran 1 dan 2", "Titik di dalam lingkaran 1", "Titik di dalam lingkaran 2", atau "Titik di luar lingkaran 1 dan 2".

Source code :



```
guided modul 3(1).go 4    guided modul 3(2).go 4    unguided modul 3.go 1 X
D: > laprak > LAPRAK MODUL 3 > ARVAN MURBIYANTO 231111102074 > unguided > unguided modul 3.go > ...
1  package main
2
3  import (
4      "fmt"
5      "math"
6  )
7
8  func hitungJarak(x1, y1, x2, y2 float64) float64 {
9      return math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))
10 }
11
12 func diDalamLingkaran(cx, cy, r, x, y float64) bool {
13     return hitungJarak(cx, cy, x, y) <= r
14 }
15
16 func main() {
17     var cx1, cy1, r1, cx2, cy2, r2, x, y float64
18
19     fmt.Print("Lingkaran 1 : ")
20     fmt.Scanln(&cx1, &cy1, &r1)
21
22     fmt.Print("Lingkaran 2 : ")
23     fmt.Scanln(&cx2, &cy2, &r2)
24
25     fmt.Print("Titik sembarang : ")
26     fmt.Scanln(&x, &y)
27
28     dilingkaran1 := diDalamLingkaran(cx1, cy1, r1, x, y)
29     dilingkaran2 := diDalamLingkaran(cx2, cy2, r2, x, y)
30
31     if dilingkaran1 && dilingkaran2 {
32         fmt.Println("Titik di dalam lingkaran 1 dan 2")
33     } else if dilingkaran1 {
34         fmt.Println("Titik di dalam lingkaran 1")
35     } else if dilingkaran2 {
36         fmt.Println("Titik di dalam lingkaran 2")
37     } else {
38         fmt.Println("Titik di luar lingkaran 1 dan 2")
39     }
40 }
41
```


Output :



```
PROBLEMS 9 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\WSI> go run "d:\laprak\LAPRAK MODUL 3\ARVAN MURBIYANTO 23111102074\unguided\unguided modul 3.go"
Lingkaran 1 : 1 1 5
Lingkaran 2 : 8 8 4
Titik sembarang : 2 2
Titik di dalam lingkaran 1
PS C:\Users\WSI> go run "d:\laprak\LAPRAK MODUL 3\ARVAN MURBIYANTO 23111102074\unguided\unguided modul 3.go"
Lingkaran 1 : 1 2 3
Lingkaran 2 : 4 5 6
Titik sembarang : 7 8
Titik di dalam lingkaran 2
PS C:\Users\WSI> go run "d:\laprak\LAPRAK MODUL 3\ARVAN MURBIYANTO 23111102074\unguided\unguided modul 3.go"
Lingkaran 1 : 5 10 15
Lingkaran 2 : -15 4 20
Titik sembarang : 0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\WSI> go run "d:\laprak\LAPRAK MODUL 3\ARVAN MURBIYANTO 23111102074\unguided\unguided modul 3.go"
Lingkaran 1 : 1 1 5
Lingkaran 2 : 8 8 4
Titik sembarang : 15 20
Titik di luar lingkaran 1 dan 2
PS C:\Users\WSI>
```

Deskripsi :

Program ini membantu menentukan posisi sebuah titik relatif terhadap dua lingkaran dengan memeriksa apakah titik tersebut berada di dalam salah satu atau kedua lingkaran.

