

LAPORAN
PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL 3
FUNCTION



Oleh:

NAMA: MOHAMMAD REYHAN ARETHA FATIN

NIM: 103112400078

KELAS: IF-12-01

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

I. DASAR TEORI

1. Fungsi Dalam Pemograman

Fungsi adalah sebuah subprogram yang digunakan untuk menyelesaikan tugas tertentu dan menghasilkan nilai tertentu. Dalam konteks pemrograman, fungsi memungkinkan kode yang lebih modular, terstruktur, dan mudah dipelihara. Fungsi selalu mengembalikan nilai, yang digunakan dalam berbagai keperluan, seperti penugasan nilai variabel, ekspresi, atau argumen dalam subprogram lain.

2. Definisi Function

Fungsi merupakan satu kesatuan rangkaian instruksi yang memberikan atau menghasilkan suatu nilai dan biasanya memetakan input ke suatu nilai yang lain. Oleh karena itu, fungsi selalu menghasilkan/mengembalikan nilai. Suatu subprogram dikatakan fungsi apabila:

1. Ada deklarasi tipe nilai yang dikembalikan, dan
2. Terdapat kata kunci return dalam badan subprogram.

Maka fungsi digunakan jika suatu nilai biasanya diperlukan, seperti:

- Assignment nilai ke suatu variabel
- Bagian dari ekspresi
- Bagian dari argumen suatu subprogram, dsb.

Karena itu selalu pilih nama fungsi yang menggambarkan nilai, seperti kata benda dan kata sifat. Contoh nama-nama fungsi: median, rerata, nilaiTerbesar, ketemu, selesai, ...

3. 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.

II. GUIDED

GUIDED 1

SOURCE CODE:

```
guided1 > guided1.go > main
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 func permutasi(n, r int) int {
23     if r > n {
24         return 0
25     }
26     return faktorial(n) / faktorial(n-r)
27 }
28
```

OUTPUT:

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRA
5 6
720
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> 
```

DEKSRIPSI:

Program yang menghitung permutasi dua angka yang diberikan oleh pengguna. Program dimulai dengan menerima dua input integer, yaitu a dan b. Jika a lebih besar atau sama dengan b, maka program akan menghitung dan menampilkan hasil permutasi dari a dan b dengan memanggil fungsi permutasi(a, b). Jika tidak, fungsi permutasi(b, a) yang dipanggil. Fungsi permutasi sendiri menghitung permutasi dengan menggunakan rumus faktorial, yang dihitung melalui fungsi faktorial. Fungsi faktorial mengalikan angka dari 1 hingga n untuk menghasilkan hasil faktorial, yang kemudian digunakan dalam rumus permutasi.

GUIDED 2

SURCE CODE

```
guided2 > go guided2.go > main
1 package main
2
3 import "fmt"
4
5 func celciusToFahrenheit(celcius float64) float64 {
6     return (celcius * 9.0 / 5.0) + 32
7 }
8
9 func main() {
10     var N int
11     fmt.Print("Masukan jumlah data: ")
12     _, err := fmt.Scan(&N)
13     if err != nil || N <= 0 {
14         fmt.Println("Input tidak valid, masukan angka positif.")
15         return
16     }
17     temperatures := make([]float64, N)
18
19     fmt.Println("Masukan suhu dalam celcius: ")
20     for i := 0; i < N; i++ {
21         _, err = fmt.Scan(&temperatures[i])
22         if err != nil {
23             fmt.Println("Input tidak valid, pastikan masukan angka.")
24         }
25     }
26
27     fmt.Println("Suhu dalam Fahrenheit: ")
28     for _, temp := range temperatures {
29         fmt.Printf("%.2f\n", celciusToFahrenheit(temp))
30     }
31 }
32
```

OUTPUT

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumSem2\
Masukan jumlah data: 3
Masukan suhu dalam celcius:
18
22
24
Suhu dalam Fahrenheit:
64.40
71.60
75.20
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3>
```

DEKSRIPSI

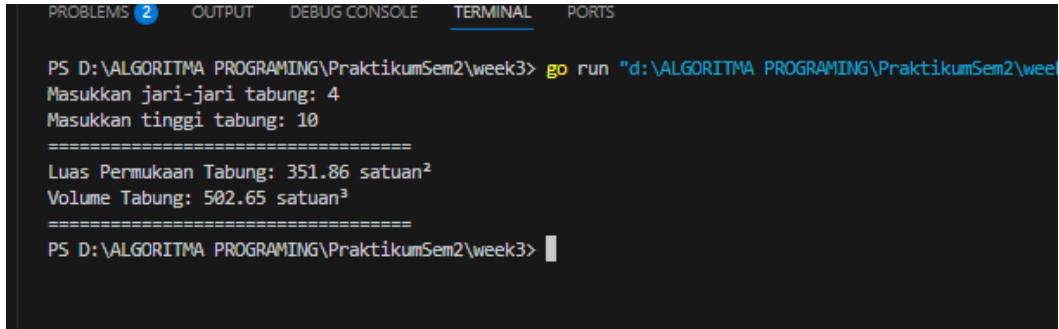
Program ini menerima input suhu dalam derajat Celsius dari pengguna dan mengkonversinya ke Fahrenheit. Pertama, program meminta jumlah data suhu yang ingin dimasukkan. Jika input tidak valid atau jumlah data yang dimasukkan kurang dari atau sama dengan nol, program akan menampilkan pesan kesalahan. Kemudian, program meminta pengguna untuk memasukkan suhu dalam Celsius sebanyak jumlah data yang sudah ditentukan. Setiap suhu yang dimasukkan disimpan dalam array temperatures. Setelah semua suhu dimasukkan, program mengonversi setiap suhu Celsius ke Fahrenheit menggunakan fungsi celciusToFahrenheit dengan rumus: $(\text{celcius} * 9.0 / 5.0) + 32$ Program kemudian menampilkan hasil konversi suhu dalam Fahrenheit.

GUIDED 3

SOURCE CODE

```
guided3 > ~ guided3.go > ...
2
3 import
4     "fmt"
5     "math"
6
7
8 // Fungsi untuk menghitung luas permukaan tabung
9 func luasPermukaanTabung(r, t float64) float64 {
10     return 2 * math.Pi * r * (r + t)
11 }
12
13 // Fungsi untuk menghitung volume tabung
14 func volumeTabung(r, t float64) float64 {
15     return math.Pi * math.Pow(r, 2) * t
16 }
17
18 func main() {
19     var r, t float64
20
21     // Input jari-jari dan tinggi tabung dengan validasi
22     fmt.Print("Masukkan jari-jari tabung: ")
23     _, errR := fmt.Scan(&r)
24     fmt.Print("Masukkan tinggi tabung: ")
25     _, errT := fmt.Scan(&t)
26
27     // Memeriksa apakah input valid
28     if errR != nil || errT != nil {
29         fmt.Println("Input tidak valid! Harap masukkan angka yang benar.")
30         return
31     }
32
33     // Memeriksa apakah jari-jari dan tinggi bernilai positif
34     if r <= 0 || t <= 0 {
35         fmt.Println("Jari-jari dan tinggi tabung harus lebih dari nol.")
36         return
37     }
38
39     // Menghitung luas permukaan dan volume
40     luas := luasPermukaanTabung(r, t)
41     volume := volumeTabung(r, t)
42
43     // Menampilkan hasil
44     fmt.Println("=====")
45     fmt.Printf("Luas Permukaan Tabung: %.2f satuan²\n", luas)
46     fmt.Printf("Volume Tabung: %.2f satuan³\n", volume)
47     fmt.Println("=====")
48 }
49
```

OUTPUT



```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumSem2\wee
Masukkan jari-jari tabung: 4
Masukkan tinggi tabung: 10
=====
Luas Permukaan Tabung: 351.86 satuan2
Volume Tabung: 502.65 satuan3
=====
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> |
```

DEKSRISI

Program ini menghitung luas permukaan dan volume tabung berdasarkan input jari-jari dan tinggi tabung dari pengguna. Program dimulai dengan meminta input jari-jari dan tinggi tabung, memastikan input tersebut valid dan positif. Jika input tidak valid atau kurang dari atau sama dengan nol, program akan menampilkan pesan kesalahan dan berhenti. Jika input valid, program melanjutkan untuk menghitung luas permukaan dan volume tabung menggunakan rumus:
Luas permukaan tabung
dan Volume tabung
Setelah menghitung, program menampilkan hasil luas permukaan dan volume tabung dalam satuan yang sesuai dengan format yang ditentukan.

III. UNGUIDED

UNGUIDED 1

SOURCE CODE

```
unguided1 > 1.go > ...
1  package main
2  // 103112400078
3  // Mohammad Reyhan Aretha Fatin
4  import "fmt"
5
6  func faktorial(n int) int {
7      if n <= 1 {
8          return 1
9      }
10     fktr := 1
11     for i := 2; i <= n; i++ {
12         fktr *= i
13     }
14     return fktr
15 }
16
17 func hitungPermutasi(n, r int) int {
18     return faktorial(n) / faktorial(n-r)
19 }
20
21 func hitungKombinasi(n, r int) int {
22     return faktorial(n) / (faktorial(r) * faktorial(n-r))
23 }
24
25 func main() {
26     var a, b, c, d int
27     fmt.Scan(&a, &b, &c, &d)
28
29     if a >= c && b >= d {
30         fmt.Println(hitungPermutasi(a, c), hitungKombinasi(a, c))
31         fmt.Println(hitungPermutasi(b, d), hitungKombinasi(b, d))
32     } else {
33         fmt.Println("Input tidak sesuai")
34     }
35 }
36
```

OUTPUT



```
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumSem
5 20 3 20
60 10
2432902008176640000 1
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumSem
8 0 2 0
56 28
1 1
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> |
```

DEKSRIPSI


Program ini menghitung permutasi dan kombinasi berdasarkan input yang diberikan. Program dimulai dengan mendefinisikan fungsi faktorial, yang digunakan untuk menghitung faktorial dari suatu angka. Fungsi hitungPermutasi dan hitungKombinasi kemudian menggunakan fungsi faktorial untuk menghitung permutasi dan kombinasi sesuai dengan rumus matematis. Pada bagian main, program menerima empat input: dua angka untuk menghitung permutasi dan kombinasi pertama, dan dua angka lagi untuk menghitung permutasi dan kombinasi kedua. Setelah itu, program memeriksa apakah kondisi pertama ($a \geq b$ dan $c \geq d$) terpenuhi. Jika iya, hasil perhitungan permutasi dan kombinasi akan ditampilkan. Jika tidak, program akan menampilkan pesan "Input tidak sesuai" untuk menunjukkan bahwa input yang diberikan tidak valid atau tidak memenuhi syarat.

UNGUIDED 2

SOURCE CODE

```
unguided2 > 2.go > ...
1  package main
2  √ //1031121400078
3  //Mohammad Reyhan Aretha Fatin
4  import "fmt"
5
6  √ func kuadrat(n int) int {
7      |     return n * n
8      | }
9  √ func kurang(n int) int {
10     |     return n - 2
11     | }
12 √ func tambah(n int) int {
13     |     return n + 1
14     | }
15 √ func fogoh(n int) int {
16     |     return kuadrat(kurang(tambah(n)))
17     | }
18 √ func gohof(n int) int {
19     |     return kurang(tambah(kuadrat(n)))
20     | }
21 √ func hofog(n int) int {
22     |     return tambah(kuadrat(kurang(n)))
23     | }
24 √ func main() {
25     |     var a, b, c int
26     |     fmt.Scan(&a, &b, &c)
27     |     fmt.Println(fogoh(a))
28     |     fmt.Println(gohof(b))
29     |     fmt.Println(hofog(c))
30     | }
31
```

OUTPUT




```
PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumS
7 2 10
36
3
65
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumS
5 5 5
16
24
10
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> 
```

DEKSRIPSI

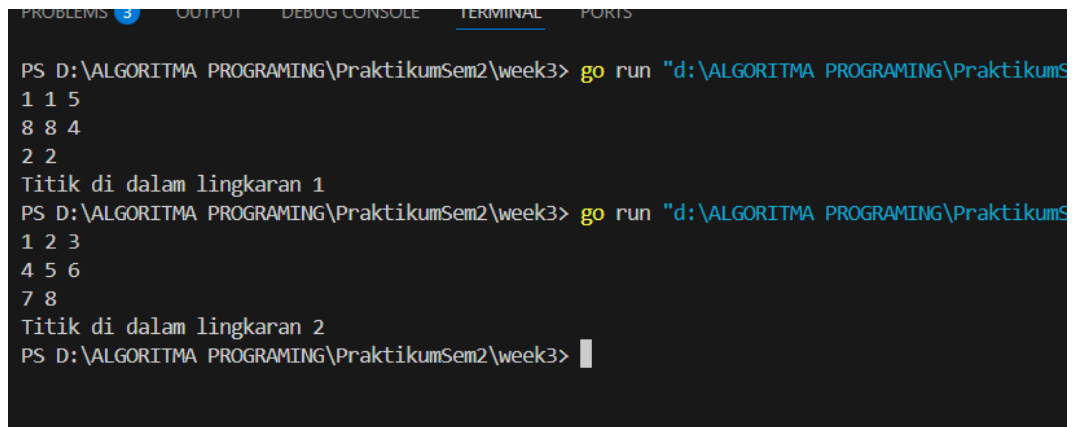
Program ini berfungsi untuk menghitung nilai dari tiga fungsi komposisi yang berbeda dengan menggunakan beberapa operasi matematika. Fungsi-fungsi yang ada adalah kuadrat, kurang, dan tambah, yang masing-masing melakukan operasi pengkuadratan, pengurangan 2, dan penambahan 1 terhadap nilai input. Tiga fungsi komposisi utama yang didefinisikan adalah fogoh, gohof, dan hofog, yang mengkombinasikan ketiga fungsi ini dalam urutan yang berbeda. Dalam fungsi main, program menerima tiga nilai input yaitu a, b, dan c, kemudian menghitung hasil komposisi dari masing-masing fungsi untuk fogoh(a), gohof(b), dan hofog(c) dan menampilkannya sebagai output.

UNGUIDED 3

SOURCE CODE

```
unguided3 >  3.go > ...
1  package main
2  // 103112400078
3  // Mohammad Reyhan Aretha Fatin
4  import "fmt"
5
6  func jarak(a, b, c, d float64) float64 {
7      return ((a - c) * (a - c)) + ((b - d) * (b - d))
8  }
9  func didalam(cx, cy, r, x, y float64) bool {
10     return jarak(cx, cy, x, y) <= (r * r)
11 }
12 func main() {
13     var cx1, cy1, r1 float64
14     var cx2, cy2, r2 float64
15     var x, y float64
16     fmt.Scan(&cx1, &cy1, &r1)
17     fmt.Scan(&cx2, &cy2, &r2)
18     fmt.Scan(&x, &y)
19     dalamLingkaran1 := didalam(cx1, cy1, r1, x, y)
20     dalamLingkaran2 := didalam(cx2, cy2, r2, x, y)
21     if dalamLingkaran1 && dalamLingkaran2 {
22         fmt.Println("Titik di dalam lingkaran 1 dan 2")
23     } else if dalamLingkaran1 {
24         fmt.Println("Titik di dalam lingkaran 1")
25     } else if dalamLingkaran2 {
26         fmt.Println("Titik di dalam lingkaran 2")
27     } else {
28         fmt.Println("Titik di luar lingkaran 1 dan 2")
29     }
30 }
```

OUTPUT

A screenshot of a Go IDE's terminal window. The terminal shows the execution of a Go program. The first run takes three inputs (1 1 5, 8 8 4, 2 2) and outputs 'Titik di dalam lingkaran 1'. The second run takes three inputs (1 2 3, 4 5 6, 7 8) and outputs 'Titik di dalam lingkaran 2'. The terminal tabs at the top are 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL', and 'PORTS'.

```
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumS
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> go run "d:\ALGORITMA PROGRAMING\PraktikumS
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS D:\ALGORITMA PROGRAMING\PraktikumSem2\week3> |
```

DEKSRIPSI

Program ini bertujuan untuk memeriksa apakah sebuah titik (dengan koordinat x dan y) berada di dalam dua lingkaran yang berbeda. Fungsi jarak digunakan untuk menghitung jarak antara dua titik menggunakan rumus jarak Euclidean, dan fungsi didalam memeriksa apakah titik tersebut berada di dalam lingkaran dengan membandingkan jarak titik ke pusat lingkaran dengan jari-jari lingkaran. Dalam fungsi main, program meminta input untuk pusat dan jari-jari kedua lingkaran, serta koordinat titik yang akan diuji. Program kemudian memeriksa apakah titik tersebut berada di dalam kedua lingkaran, hanya di lingkaran pertama, hanya di lingkaran kedua, atau di luar kedua lingkaran, dan menampilkan hasilnya sesuai dengan kondisi tersebut.

IV. KESIMPULAN

Kesimpulan dari laporan praktikum ini adalah bahwa fungsi dalam pemrograman sangat penting untuk menyelesaikan tugas-tugas tertentu secara modular dan efisien. Dalam setiap program yang diuji, berbagai jenis fungsi digunakan untuk menghitung hasil dari berbagai masalah matematis dan fisika, seperti permutasi, konversi suhu, dan perhitungan volume dan luas permukaan tabung. Keberhasilan program tergantung pada pemahaman dan implementasi yang benar terhadap konsep-konsep fungsi, parameter, dan pengembalian nilai. Keseluruhan praktikum ini memberikan pemahaman yang lebih baik tentang cara kerja fungsi dalam pemrograman dan memperlihatkan pentingnya keterampilan dalam mendefinisikan serta memanggil fungsi dengan tepat dalam memecahkan masalah komputasi.

REFERENSI

MODUL 3-FUNGSI-ALGORITMA PEMOGRAMAN 2