

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL 3

FUNGSI



Disusun Oleh :

HANIF REYHAN ZHAFRAN ADRYTONA / 2311102266

11 IF 06

Dosen Pengampu :

Abednego Dwi Septiadi, S.Kom., M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Fungsi dalam Go adalah blok kode yang dirancang untuk menjalankan tugas tertentu, menerima input, memprosesnya, dan mengembalikan output. Fungsi ini sangat penting karena memungkinkan pengembang menulis kode yang modular dan terorganisir, yang memudahkan pemeliharaan serta pengembangan aplikasi berskala besar.. Fungsi membantu memecah program besar menjadi komponen-komponen yang lebih kecil, sehingga lebih mudah dikelola, dipahami, dan di-debug. Dalam setiap program Go, setidaknya ada satu fungsi utama, yaitu ``main()``, yang menjadi titik awal eksekusi program. Fungsi di Go juga bersifat **first-class**, yang berarti fungsi dapat diperlakukan seperti nilai lainnya. Artinya, fungsi bisa disimpan dalam variabel, dilewatkan sebagai argumen ke fungsi lain, atau dikembalikan sebagai nilai dari fungsi lain, memberi fleksibilitas besar dalam desain program.

Selain itu, Golang mendukung berbagai fitur fungsi seperti parameter dan nilai kembalian **return value**. Fungsi dapat menerima input berupa parameter, yang bisa dari tipe data sederhana seperti integer dan string, hingga tipe data yang lebih kompleks seperti struktur **struct** dan slice. Keunikan Golang adalah kemampuan untuk mengembalikan lebih dari satu nilai dari fungsi. Hal ini sangat berguna, misalnya, dalam penanganan kesalahan di mana suatu fungsi dapat mengembalikan hasil operasi serta informasi kesalahan (error) secara bersamaan. Dengan demikian, penanganan kesalahan menjadi lebih jelas dan terstruktur tanpa memerlukan mekanisme pengecualian seperti di bahasa pemrograman lain.

Golang juga mendukung fitur **anonymous functions** atau fungsi anonim, yang merupakan fungsi tanpa nama. Fungsi anonim biasanya digunakan sebagai fungsi sementara, terutama dalam situasi **inline** atau sebagai argumen fungsi lain. Karena fungsi di Go adalah objek first-class, fungsi anonim sering dipakai dalam teknik-teknik seperti concurrency atau penanganan event, di mana fungsi anonim dapat dilewatkan sebagai parameter untuk digunakan di berbagai konteks.

Selain itu, Golang memungkinkan penggunaan **closures**, yakni fungsi yang dapat mengakses dan menyimpan variabel dari lingkup **scope** di mana fungsi tersebut didefinisikan, bukan hanya dari lingkup eksekusi saat itu. **Closures** sering digunakan dalam fungsi bersarang atau ketika membuat fungsi yang dipersonalisasi sesuai konteks spesifik.

Dalam hal concurrency, fungsi di Golang memainkan peran krusial melalui goroutines. Goroutines adalah fungsi yang berjalan secara bersamaan dengan fungsi lainnya, memungkinkan program untuk menangani beberapa pekerjaan secara paralel. Fitur goroutines ini memberikan kemampuan untuk menjalankan tugas secara asinkron dan mempercepat pemrosesan, menjadikan program lebih efisien. Kombinasi antara goroutines dan channel dalam Go memungkinkan komunikasi yang aman antara fungsi yang berjalan secara paralel, yang sangat berguna dalam membangun aplikasi berskala besar yang memerlukan pemrosesan cepat dan responsif.

Dengan fitur-fitur fungsionalitas yang kuat dan dukungan bawaan terhadap concurrency, fungsi di Golang menjadi fondasi utama dalam pengembangan aplikasi yang efisien, skalabel, dan responsif. Golang menyederhanakan banyak aspek pemrograman modern, menjadikannya pilihan ideal untuk membangun aplikasi yang membutuhkan kinerja tinggi dan skalabilitas yang baik.

II. GUIDED

Guided 1.0

Sourcecode

```
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)
}
```

Screenshoot Output



```
PROBLEMS  OUTPUT  TERMINAL  PORTS  COMMENTS  DEBUG CONSOLE

● PS C:\Users\M S I> go run "d:\Hanif\Golang\test.go\test.go"
go run "d:\Hanif\Golang\test.go\test.go"
1
○ PS C:\Users\M S I> 
```

Deskripsi Program

Program di atas adalah program go untuk menghitung permutasi $P(n, r)$ dengan dua bilangan input dari pengguna. Setelah memasukkan nilai, program akan membandingkan bilangan tersebut dan menentukan yang lebih besar sebagai nilai n dan yang lebih kecil sebagai r , lalu menghitung permutasi menggunakan fungsi `permutasi` yang memanggil fungsi `faktorial`. Namun, ada kesalahan dalam fungsi `faktorial`, di mana perhitungan hanya mengalikan hasil dengan 1, bukan dengan variabel `i` yang seharusnya, sehingga perlu diperbaiki agar perhitungan faktorial berjalan dengan benar dan hasil permutasi sesuai.

Guided 1.2

Sourcecode

```
package main

import (
    "fmt"
)

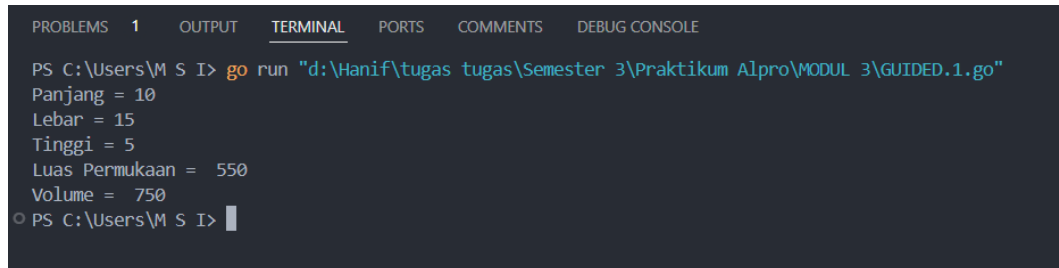
func main() {
    balok()
}

func balok() {
    var panjang, lebar, tinggi float64
    fmt.Print("Panjang = ")
    fmt.Scanln(&panjang)
    fmt.Print("Lebar = ")
    fmt.Scanln(&lebar)
    fmt.Print("Tinggi = ")
    fmt.Scanln(&tinggi)

    luasPermukaan := 2 * (panjang*lebar + panjang*tinggi +
lebar*tinggi)
    volume := panjang * lebar * tinggi

    fmt.Println("Luas Permukaan = ", luasPermukaan)
    fmt.Println("Volume = ", volume)
}
```

Screenshoot Output

A screenshot of a terminal window showing the execution of a Go program. The terminal has tabs for PROBLEMS, OUTPUT, TERMINAL, PORTS, COMMENTS, and DEBUG CONSOLE. The output shows the program calculating the surface area and volume of a rectangular prism with dimensions 10, 15, and 5. The calculations are: Panjang = 10, Lebar = 15, Tinggi = 5, Luas Permukaan = 550, and Volume = 750.

```
PS C:\Users\M S I> go run "d:\Hanif\tugas tugas\Semester 3\Praktikum Alpro\MODUL 3\GUIDED.1.go"
Panjang = 10
Lebar = 15
Tinggi = 5
Luas Permukaan = 550
Volume = 750
PS C:\Users\M S I>
```

Deskripsi Program

Program di atas adalah program Go yang menghitung luas permukaan dan volume sebuah balok. Program ini meminta pengguna memasukkan nilai panjang, lebar, dan tinggi balok, kemudian menggunakan fungsi `balok()` untuk menghitung luas permukaan dan volume berdasarkan rumus-rumus dasar. Rumus luas permukaan balok adalah $2 \times (\text{panjang} \times \text{lebar} + \text{panjang} \times \text{tinggi} + \text{lebar} \times \text{tinggi})$, sedangkan rumus volume adalah $\text{panjang} \times \text{lebar} \times \text{tinggi}$. Setelah perhitungan selesai, hasilnya ditampilkan ke layar.

Guided 1.3

Sourcecode

The source code is a Go program in the `main` package. It imports the `fmt` package and declares four variables: `a`, `b`, `c`, and `d` as integers. It defines three functions: `faktorial` which calculates the factorial of a number `n` using a loop; `permutasi` which calculates the number of permutations of `n` items taken `r` at a time by dividing the factorial of `n` by the factorial of `n-r`; and `kombinasi` which calculates the number of combinations of `n` items taken `r` at a time.

```
package main

import "fmt"

var a, b, c, d int

func faktorial(n int) int {
    hasil := 1

    for i := 1; i <= n; i++ {
        hasil = hasil * i
    }
    return hasil
}

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

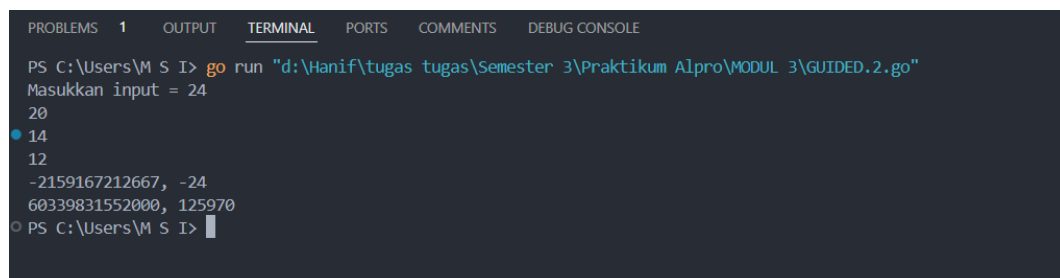
    if a >= c && b >= d {

        fmt.Printf("%d, %d\n", permutasi(a, c), kombinasi(a, c))

        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



```

PROBLEMS 1 OUTPUT TERMINAL PORTS COMMENTS DEBUG CONSOLE
PS C:\Users\M S I> go run "d:\Hanif\tugas tugas\Semester 3\Praktikum Alpro\MODUL 3\GUIDED.2.go"
Masukkan input = 24
14
12
-2159167212667, -24
60339831552000, 125970
PS C:\Users\M S I>

```

Deskripsi Program

Program di atas adalah sebuah aplikasi sederhana dalam bahasa Go yang menghitung permutasi dan kombinasi dari dua set angka berdasarkan input pengguna. Program ini menggunakan fungsi `faktorial` untuk menghitung faktorial dari suatu angka, lalu fungsi `permutasi` dan `kombinasi` untuk menghitung hasil permutasi dan kombinasi dengan rumus matematika yang sesuai. Pengguna diminta untuk memasukkan empat angka, yaitu `a`, `b`, `c`, dan `d`. Program akan mengevaluasi apakah `a` lebih besar atau sama dengan `c` dan `b` lebih besar atau sama dengan `d`. Jika syarat tersebut terpenuhi, program akan mencetak hasil permutasi dan kombinasi dari `a` terhadap `c` dan `b` terhadap `d`. Jika tidak, program akan menampilkan pesan bahwa syarat tersebut tidak terpenuhi.

III. UNGUIDED

Soal Studi Case

Minggu ini, mahasiswa Fakultas Informatika mendapatkan tugas dari mata kuliah matematika dishrit untuk mempelajari kombinasi dan permutasi. Jonas salah seorang mahasiswa, iseng untuk mengimplementasikannya he dalam suatu program. Oleh karena itu bersediakah kalian membantu Jonas? (tidak tentunya ya) Masukan terdiri dari empat buah bilangan asli a, b, c, dan d yang dipisahkan Oleh spasi, dengan Syarat a > c dan b > d. Keluaran terdiri dari dua baris. Baris pertama adalah hasil permutasi dan kombinasi a terhadap c, sedangkan baris kedua adalah hasil permutasi dan kombinasi b terhadap d. Catatan: permutasi (P) dan kombinasi (C) dari n terhadap r ($n \geq r$) dapat dihitung dengan menggunakan persamaan berikut!

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

Sourcecode

```
package main

import "fmt"

func main() {
    var a, b, c, d int
    fmt.Print("LEBOK KE NILE: ")
    fmt.Scan(&a, &b, &c, &d)
    if a >= c && b >= d {
        fmt.Println("Permutasi:", permutasi(a, c))
        fmt.Println("Kombinasi:", kombinasi(a, c))
        fmt.Println("Permutasi:", permutasi(b, d))
        fmt.Println("Kombinasi:", kombinasi(b, d))
    } else {
        fmt.Println("salah cok")
    }
}

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

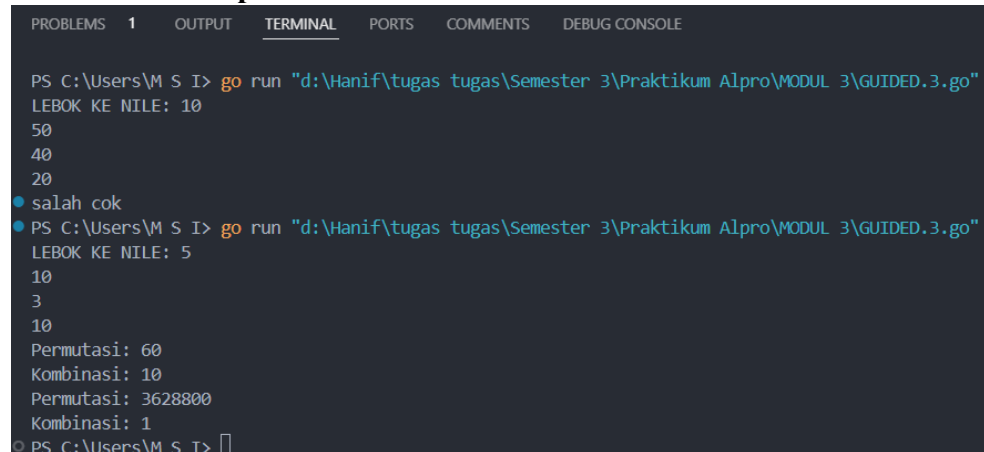
func faktorial(n int) int {
    hasil_2311102266 := 1
    for i := 1; i <= n; i++ {
        hasil_2311102266 *= i
    }
    return hasil_2311102266
}
```



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

// 2311102266_Hanif reyhan
```

Screenshoot Output



```
PROBLEMS 1 OUTPUT TERMINAL PORTS COMMENTS DEBUG CONSOLE

PS C:\Users\M S I> go run "d:\Hanif\tugas tugas\Semester 3\Praktikum Alpro\MODUL 3\GUIDED.3.go"
LEBOK KE NILE: 10
50
40
20
• salah cok
• PS C:\Users\M S I> go run "d:\Hanif\tugas tugas\Semester 3\Praktikum Alpro\MODUL 3\GUIDED.3.go"
LEBOK KE NILE: 5
10
3
10
Permutasi: 60
Kombinasi: 10
Permutasi: 3628800
Kombinasi: 1
PS C:\Users\M S I>
```

Deskripsi Program

Program di atas adalah program bahasa Go yang digunakan untuk menghitung permutasi dan kombinasi dari dua set angka yang diberikan oleh pengguna. Pertama, program meminta input empat angka ('a', 'b', 'c', dan 'd') dari pengguna. Setelah menerima input, program memeriksa apakah nilai 'a' lebih besar atau sama dengan 'c' dan 'b' lebih besar atau sama dengan 'd'. Jika kondisi ini terpenuhi, program akan menampilkan hasil perhitungan permutasi dan kombinasi dari 'a' terhadap 'c', serta 'b' terhadap 'd'. Program menggunakan fungsi 'faktorial' untuk menghitung faktorial dari suatu angka, yang kemudian digunakan dalam fungsi 'permutasi' dan 'kombinasi' untuk menghitung hasil yang diinginkan. Jika kondisi tidak terpenuhi, program akan mencetak pesan kesalahan "salah cok". Program ini menambahkan elemen dekoratif, seperti teks "LEBOK KE NILE" untuk meminta input dan variabel 'hasil_2311102266', yang disesuaikan dengan nomor identitas.

Unguided 2

SOAL :

Diberikan tiga buah fungsi matematika yaitu $f(x) = x^2$, $g(x) = x - 2$ dan $h(x) = x + 1$. Fungsi komposisi $(f \circ g \circ h)(x)$ artinya adalah Tuliskan $f(x)$, $g(x)$ dan $h(x)$ dalam bentuk function. Masukan terdiri dari sebuah bilangan bulat a , b dan c yang dipisahkan Oleh spasi. Keluaran terdiri dari tiga baris. Baris pertama adalah $(f \circ g \circ h)(a)$, baris kedua $(g \circ h \circ f)(b)$, dan baris ketiga adalah $(h \circ f \circ g)(c)$

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var a_2311102266, b_2311102266, c_2311102266 int
    fmt.Print("Masukkan Nilai a, b, c: ")
    fmt.Scanf("%d %d %d", &a_2311102266, &b_2311102266,
    &c_2311102266)
    fogohResult := fogoh(a_2311102266)
    gohofResult := gohof(b_2311102266)
    hofogResult := hofog(c_2311102266)
    fmt.Printf("%d\n", fogohResult)
    fmt.Printf("%d\n", gohofResult)
    fmt.Printf("%d\n\n", hofogResult)
}

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

// 2311102266_Hanif reyhan

```

Screenshoot Output

```

PROBLEMS 1 OUTPUT TERMINAL PORTS COMMENTS DEBUG CONSOLE
PS C:\Users\M S I> go run "d:\Hanif\tugas tugas\Semester 3\Praktikum Alpro\MODUL 3\UNGUIDED.1.go"
Masukkan Nilai a, b, c: 10 50 100
81
2499
9605

PS C:\Users\M S I> go run "d:\Hanif\tugas tugas\Semester 3\Praktikum Alpro\MODUL 3\UNGUIDED.1.go"
Masukkan Nilai a, b, c: 3 4 5
4
15
10

PS C:\Users\M S I>

```

Deskripsi Program

Program ini adalah program bahasa Go yang melakukan komposisi fungsi matematika dengan tiga input: `a`, `b`, dan `c`. Program meminta pengguna memasukkan tiga nilai integer, lalu menerapkan fungsi komposisi `fogoh`, `gohof`, dan `hofog` pada nilai-nilai tersebut. Setiap fungsi komposisi mengkombinasikan fungsi-fungsi dasar $f(x) = x * x$, $g(x) = x - 2$, dan $h(x) = x + 1$ dalam urutan tertentu: `fogoh` mengaplikasikan $f(g(h(x)))$, `gohof` melakukan $g(h(f(x)))$, dan `hofog` menghitung $h(f(g(x)))$. Hasil dari setiap

komposisi fungsi untuk nilai input `a`, `b`, dan `c` kemudian ditampilkan. Meskipun program juga memiliki fungsi `factorial`, fungsi ini tidak dipakai dalam proses komputasi utama program.

UNGUIDED 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.

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func jarak(a, b, c, d int) float64 {
    return math.Sqrt(math.Pow(float64(a-c), 2) + math.Pow(float64(b-d), 2))
}

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

    fmt.Println("Masukkan data lingkaran 1 (format: x y radius):")
    fmt.Scan(&cx1, &cy1, &r1)

    fmt.Println("Masukkan data lingkaran 2 (format: x y radius):")
    fmt.Scan(&cx2, &cy2, &r2)

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

    d1 := jarak(x, y, cx1, cy1)
```

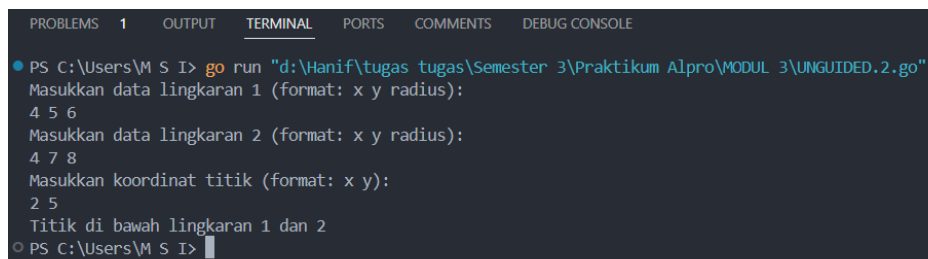
```

d2 := jarak(x, y, cx2, cy2)

if d1 <= float64(r1) && d2 <= float64(r2) {
    fmt.Println("Titik di bawah lingkaran 1 dan 2")
} else if d1 <= float64(r1) {
    fmt.Println("Titik di bawah lingkaran 1")
} else if d2 <= float64(r2) {
    fmt.Println("Titik di bawah lingkaran 2")
} else {
    fmt.Println("Titik di luar lingkaran 1 dan 2")
}
}
// 2311102266_Hanif reyhan

```

Screenshot Output



```

PROBLEMS 1 OUTPUT TERMINAL PORTS COMMENTS DEBUG CONSOLE
● PS C:\Users\M S I> go run "d:\Hanif\tugas tugas\Semester 3\Praktikum Alpro\MODUL 3\UNGUIDED.2.go"
Masukkan data lingkaran 1 (format: x y radius):
4 5 6
Masukkan data lingkaran 2 (format: x y radius):
4 7 8
Masukkan koordinat titik (format: x y):
2 5
Titik di bawah lingkaran 1 dan 2
○ PS C:\Users\M S I>

```

Deskripsi Program

Program di atas berfungsi untuk menentukan posisi suatu titik relatif terhadap dua lingkaran yang didefinisikan oleh koordinat pusat dan jari-jarinya. Pengguna diminta untuk memasukkan data lingkaran pertama (koordinat x, y dan radius) serta lingkaran kedua, diikuti dengan koordinat titik yang ingin dianalisis. Fungsi `jarak` digunakan untuk menghitung jarak antara titik tersebut dan pusat masing-masing lingkaran. Setelah itu, program membandingkan jarak tersebut dengan jari-jari lingkaran untuk menentukan apakah titik berada di dalam kedua lingkaran, salah satu lingkaran, atau di luar kedua lingkaran, dan kemudian mencetak hasilnya.