

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II
REVIEW STRUKTUR KONTROL**



Disusun Oleh :

Tegar Aji Pangestu/ 2311102021

IF-11-6

Dosen Pengampu :

ABEDNEGO DWI SEPTIADI

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Struktur kontrol dalam Golang adalah mekanisme yang mengatur alur eksekusi program. Mereka memungkinkan pengembang untuk membuat keputusan, mengulang blok kode, dan mengontrol jalannya program.

Program pertama golang adalah sebuah program dasar yang memunculkan output atau keluaran berupa text “Hello, World”. Untuk mencoba menjalankan program pertama golang silahkan tekan tombol “COBA” yang berada dibawah contoh pada aplikasi, kemudian tekan tombol “RUN” maka akan muncul jendela output dari program yang dijalankan

Pada program diatas, pada bagian baris pertama terdapat potongan kode “package main” atau bisa disebut dengan deklarasi dari paket yang digunakan. Paket adalah sebuah cara untuk mengatur dan menggunakan kode program yang dibutuhkan. Program Golang (Go) terbagi menjadi dua jenis yaitu executable dan pustaka (libraries). Program Golang executable adalah program golang yang dapat dijalankan langsung melalui terminal yang tersedia. Sedangkan program Golang pustaka (libraries) adalah program golang yang merupakan kumpulan atau koleksi dari program Golang yang dijadikan paket sehingga dapat dijalankan pada program Golang lain

I. GUIDED

Soal Studi Case

Sourcecode

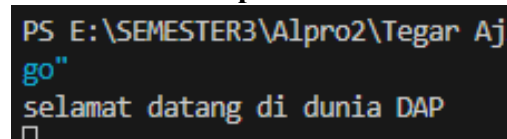
```
package main

import "fmt"

func main() {
    var greetings = "selamat datang di dunia DAP"
    var a, b int

    fmt.Println(greetings)
    fmt.Scanln(&a, &b)
    fmt.Println("%v + %v = %v\n", a, b, a+b)
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aj...
go
selamat datang di dunia DAP
█
```

Deskripsi Program

Program memulai dengan mendeklarasikan sebuah variabel bernama greetings yang berisi teks "selamat datang di dunia DAP". Kemudian, nilai variabel greetings ini dicetak ke layar menggunakan fungsi fmt.Println(), sehingga pengguna akan melihat pesan selamat datang tersebut.

II. GUIDED

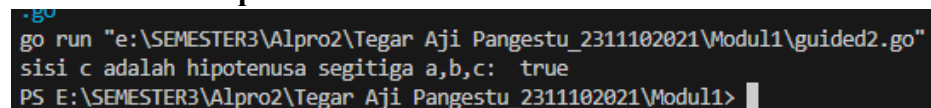
Soal Studi Case

Sourcecode

```
package main
import "fmt"
func main() {
    var a,b,c float64
    var hipotenusa bool

    fmt.Scanln(&a, &b, &c)
    hipotenusa = (c*c) == (a*a + b*b)
    fmt.Println("sisi c adalah hipotenusa segitiga
a,b,c: ", hipotenusa)
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> go run "e:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1\guided2.go"
sisi c adalah hipotenusa segitiga a,b,c: true
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1>
```

Deskripsi Program

Meminta pengguna untuk memasukkan tiga bilangan desimal (float64) yang mewakili panjang sisi-sisi suatu segitiga. Bilangan-bilangan ini akan disimpan dalam variabel a, b, dan c. Menggunakan konsep Teorema Pythagoras, program memeriksa apakah sisi c merupakan hipotenusa dari segitiga dengan sisi a dan b. Teorema Pythagoras menyatakan bahwa dalam segitiga siku-siku, kuadrat panjang sisi miring (hipotenusa) sama dengan jumlah kuadrat panjang kedua sisi lainnya. Persamaan yang digunakan adalah $c^2 == a^2 + b^2$. Jika persamaan ini benar, maka c adalah hipotenusa. Program akan mencetak hasil pemeriksaan ke layar. Jika c adalah hipotenusa, maka akan dicetak "sisi c adalah hipotenusa segitiga a,b,c: true". Jika tidak, maka akan dicetak "sisi c adalah hipotenusa segitiga a,b,c: false". a, b, c: Variabel bertipe float64 untuk menyimpan panjang sisi-sisi segitiga. hipotenusa: Variabel bertipe bool (boolean) untuk menyimpan hasil pemeriksaan. Nilai true jika c adalah hipotenusa, false jika bukan.

III. GUIDED

Soal Studi Case

Sourcecode

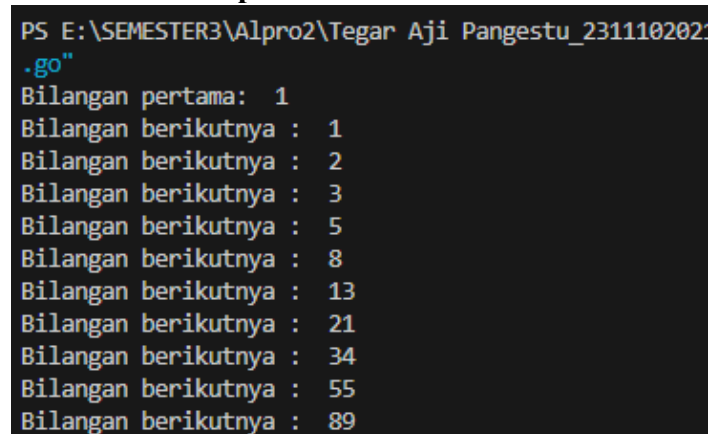
```
package main

import (
    "fmt"
)

func main() {
    maxF := 100
    f0 := 0
    f1 := 1
    f2 := 1

    fmt.Println("Bilangan pertama: ", f1)
    for f2 <= maxF {
        f0 = f1
        f1 = f2
        f2 = f1 + f0
        fmt.Println("Bilangan berikutnya : ", f1)
    }
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021>.go"
Bilangan pertama: 1
Bilangan berikutnya : 1
Bilangan berikutnya : 2
Bilangan berikutnya : 3
Bilangan berikutnya : 5
Bilangan berikutnya : 8
Bilangan berikutnya : 13
Bilangan berikutnya : 21
Bilangan berikutnya : 34
Bilangan berikutnya : 55
Bilangan berikutnya : 89
```

Deskripsi Program

Deret Fibonacci adalah urutan bilangan di mana setiap bilangan adalah jumlah dari dua bilangan sebelumnya. Program akan menghentikan perhitungan ketika nilai dalam deret Fibonacci melebihi nilai maksimum yang telah ditentukan (dalam kasus ini, 100).

IV. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
)

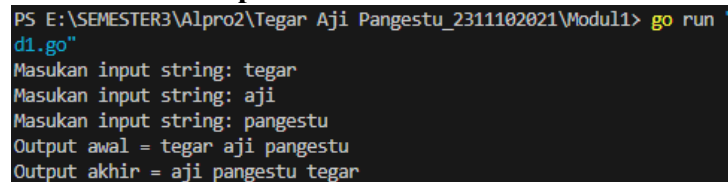
func main() {
    var celsius float64

    fmt.Print("Masukkan suhu dalam Celsius: ")
    fmt.Scan(&celsius)

    fahrenheit := (celsius * 9 / 5) + 32
    kelvin := celsius + 273.15

    fmt.Printf("%.2f°C sama dengan:\n", celsius)
    fmt.Printf("%.2f°F\n", fahrenheit)
    fmt.Printf("%.2f K\n", kelvin)
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> go run d1.go
Masukan input string: tegar
Masukan input string: aji
Masukan input string: pangestu
Output awal = tegar aji pangestu
Output akhir = aji pangestu tegar
```

Deskripsi Program

Meminta pengguna untuk memasukkan tiga buah string. String-string ini akan disimpan dalam variabel satu_2311102021, dua, dan tiga. Mencetak ketiga string yang telah dimasukkan oleh pengguna dalam urutan aslinya. Mencetak ketiga string setelah dilakukan pertukaran nilai.

V. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import "fmt"

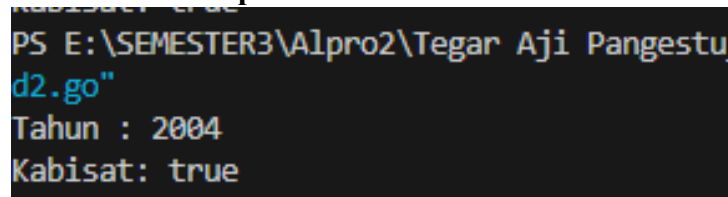
func main() {
    var tahun_2311102021 int
    var kabisat bool

    fmt.Print("Tahun : ")
    fmt.Scanln(&tahun_2311102021)

    if tahun_2311102021%400 == 0 {
        kabisat = true
    } else if tahun_2311102021%100 == 0 {
        kabisat = false
    } else if tahun_2311102021%4 == 0 {
        kabisat = true
    } else {
        kabisat = false
    }

    fmt.Println("Kabisat:", kabisat)
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu
d2.go"
Tahun : 2004
Kabisat: true
```

Deskripsi Program

Meminta pengguna untuk memasukkan sebuah tahun. Tahun ini akan disimpan dalam variabel tahun Menggunakan logika percabangan if-else if-else, program akan memeriksa apakah tahun yang dimasukkan merupakan tahun kabisat atau bukan berdasarkan aturan tahun kabisat. Jika tahun habis dibagi 400, maka tahun tersebut adalah tahun kabisat. Jika tahun habis dibagi 100 tetapi tidak habis dibagi 400, maka tahun tersebut bukan tahun kabisat. Jika tahun habis dibagi 4 tetapi tidak habis dibagi 100, maka tahun tersebut adalah tahun kabisat. Jika tidak memenuhi ketiga kondisi di atas, maka tahun tersebut bukan tahun kabisat.

VI. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var jejari_2311102021 float64
    const pi = 3.1415926535

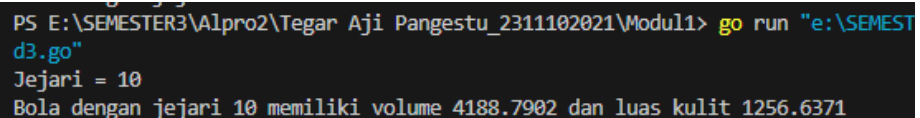
    fmt.Print("Jejari = ")
    fmt.Scanln(&jejari_2311102021)

    volume := (4.0 / 3.0) * pi *
    math.Pow(jejari_2311102021, 3)

    luas := 4 * pi * math.Pow(jejari_2311102021, 2)

    fmt.Printf("Bola dengan jejari %.0f memiliki volume
    %.4f dan luas kulit %.4f\n", jejari_2311102021, volume,
    luas)
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> go run "e:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1\d3.go"
Jejari = 10
Bola dengan jejari 10 memiliki volume 4188.7902 dan luas kulit 1256.6371
```

Deskripsi Program

Meminta pengguna untuk memasukkan nilai jari-jari sebuah bola. Nilai jari-jari ini akan disimpan dalam variabel jejari Menggunakan rumus matematika yang sudah diketahui, program menghitung volume dan luas permukaan bola berdasarkan nilai jari-jari yang diberikan.

VII. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
)

func main() {

    var celsius_2311102021 float64

    fmt.Print("Masukkan suhu dalam derajat Celsius: ")
    fmt.Scanf("%f", &celsius_2311102021)

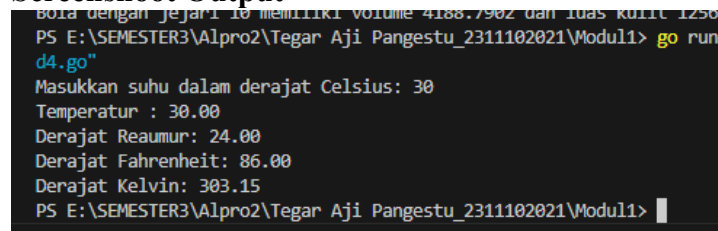
    reaumur := (4.0 / 5.0) * celsius_2311102021

    fahrenheit := (celsius_2311102021 * 9.0 / 5.0) + 32.0

    kelvin := celsius_2311102021 + 273.15

    fmt.Printf("Temperatur : %.2f\n", celsius_2311102021)
    fmt.Printf("Derajat Reaumur: %.2f\n", reaumur)
    fmt.Printf("Derajat Fahrenheit: %.2f\n", fahrenheit)
    fmt.Printf("Derajat Kelvin: %.2f\n", kelvin)
}
```

Screenshoot Output



```
Bola dengan jari 10 memiliki volume 4188.7902 dan luas kulit 1256.
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> go run
d4.go
Masukkan suhu dalam derajat Celsius: 30
Temperatur : 30.00
Derajat Reaumur: 24.00
Derajat Fahrenheit: 86.00
Derajat Kelvin: 303.15
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1>
```

Deskripsi Program

Meminta pengguna untuk memasukkan suhu dalam derajat Celsius. Nilai yang dimasukkan akan disimpan dalam variabel Celsius Menggunakan rumus-rumus konversi yang telah diketahui, program akan mengubah suhu dalam Celsius menjadi: Rumus $(4/5) * \text{Celsius}$ digunakan untuk mengkonversi suhu dari Celsius ke Reaumur. Rumus $(\text{Celsius} * 9/5) + 32$ digunakan untuk mengkonversi suhu dari Celsius ke Fahrenheit. Rumus $\text{Celsius} + 273.15$ digunakan untuk mengkonversi suhu dari Celsius ke Kelvin.

VIII. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import (
    "bufio"
    "fmt"
    "os"
)

func main() {
    var a, b, c, d, e int
    reader := bufio.NewReader(os.Stdin)

    fmt.Println("Masukkan 5 buah angka (nilai antara 32 hingga 127):")
    fmt.Scanf("%d %d %d %d %d", &a, &b, &c, &d, &e)

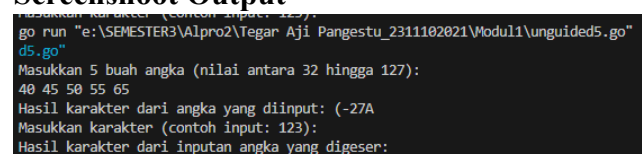
    fmt.Printf("Hasil karakter dari angka yang diinput: %c%c%c%c%c\n", a, b, c, d, e)

    fmt.Println("Masukkan karakter (contoh input: 123):")
    inputStr, _ := reader.ReadString('\n')

    shiftedStr := ""
    for _, char := range inputStr {
        if char != '\n' && char != '\r' {
            shiftedStr += string(char + 1)
        }
    }

    fmt.Printf("Hasil karakter dari inputan angka yang digeser: %s\n", shiftedStr)
}
```

Screenshot Output



```
go run "e:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1\unguided5.go"
d5.go
Masukkan 5 buah angka (nilai antara 32 hingga 127):
40 45 50 55 65
Hasil karakter dari angka yang diinput: (-27A
Masukkan karakter (contoh input: 123):
Hasil karakter dari inputan angka yang digeser:
```

Deskripsi Program

Program meminta pengguna untuk memasukkan lima buah angka dalam rentang 32 hingga 127. Angka-angka ini diasumsikan sebagai kode ASCII untuk karakter. Angka-angka yang dimasukkan kemudian langsung diinterpretasikan sebagai karakter ASCII dan dicetak ke layar. Ini memanfaatkan sifat Go yang memungkinkan konversi implisit antara integer dan rune (representasi Unicode).

IX. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var warna_2311102021 [5][4]string
    urutanBenar := [4]string{"merah", "kuning",
    "hijau", "ungu"}
    berhasil := true

    for i := 0; i < 5; i++ {
        fmt.Printf("Masukkan warna untuk percobaan
%d (pisahkan dengan spasi): ", i+1)
        fmt.Scan(&warna_2311102021[i][0],
&warna_2311102021[i][1], &warna_2311102021[i][2],
&warna_2311102021[i][3])
    }

    for i := 0; i < 5; i++ {
        fmt.Printf("percobaan %d: %s %s %s %s\n",
i+1, warna_2311102021[i][0], warna_2311102021[i][1],
warna_2311102021[i][2], warna_2311102021[i][3])
        if warna_2311102021[i] != urutanBenar {
            berhasil = false
        }
    }

    fmt.Printf("berhasil : %t\n", berhasil)
```

```
}
```

Screenshot Output

```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> go run "e:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1\d6.go"
Masukkan warna untuk percobaan 1 (pisahkan dengan spasi): biru putih merah ungu
Masukkan warna untuk percobaan 2 (pisahkan dengan spasi): biru putih merah ungu
Masukkan warna untuk percobaan 3 (pisahkan dengan spasi): biru putih merah ungu
Masukkan warna untuk percobaan 4 (pisahkan dengan spasi): biru putih merah ungu
Masukkan warna untuk percobaan 5 (pisahkan dengan spasi): biru putih merah ungu
percobaan 1: biru putih merah ungu
percobaan 2: biru putih merah ungu
percobaan 3: biru putih merah ungu
percobaan 4: biru putih merah ungu
percobaan 5: biru putih merah ungu
berhasil : false
```

Deskripsi Program

Program meminta pengguna untuk memasukkan 5 set warna, masing-masing terdiri dari 4 warna. Warna-warna ini disimpan dalam sebuah array dua dimensi bernama warna. Program memiliki sebuah array urutanBenar yang berisi urutan warna yang benar. Setiap set warna yang dimasukkan oleh pengguna akan dibandingkan dengan urutanBenar. Jika ada satu saja set warna yang urutannya tidak sesuai dengan urutanBenar, maka variabel berhasil akan diubah menjadi false.

X. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
    "strings"
)

func main() {
    var N_2311102021 int
    fmt.Print("Masukkan jumlah bunga (N): ")
    fmt.Scan(&N_2311102021)

    var pita strings.Builder
    var bunga string
    count := 0
```

```

        for i := 1; i <= N_2311102021; i++ {
            fmt.Printf("Bunga %d: ", i)
            fmt.Scan(&bunga)

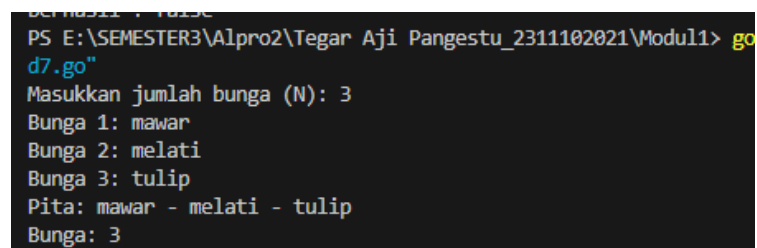
            if strings.ToUpper(bunga) == "SELESAI" {
                break
            }

            if pita.Len() > 0 {
                pita.WriteString(" - ")
            }
            pita.WriteString(bunga)
            count++
        }

        fmt.Printf("Pita: %s\n", pita.String())
        fmt.Printf("Bunga: %d\n", count)
    }

```

Screenshoot Output



```

PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> go
d7.go
Masukkan jumlah bunga (N): 3
Bunga 1: mawar
Bunga 2: melati
Bunga 3: tulip
Pita: mawar - melati - tulip
Bunga: 3

```

Deskripsi Program

Program meminta pengguna untuk memasukkan jumlah bunga yang akan diinput (N). Menggunakan perulangan for, program meminta pengguna untuk memasukkan nama bunga satu per satu. Jika pengguna memasukkan string "SELESAI", perulangan akan dihentikan.

XI. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

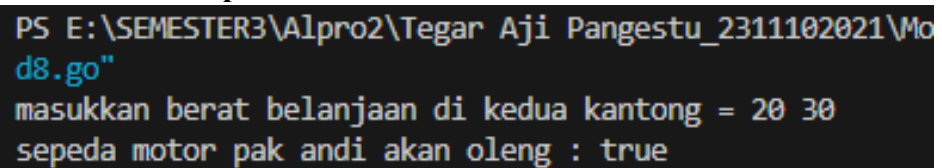
import (
    "fmt"
)

func main() {
    var berat1_2311102021, berat2 float64

    for {
        fmt.Print("masukkan berat belanjaan di kedua kantong = ")
        fmt.Scan(&berat1_2311102021, &berat2)

        if berat1_2311102021 >= 9 || berat2 >= 9 {
            fmt.Println("sepeda motor pak andi akan oleng : true")
        } else {
            fmt.Println("sepeda motor pak andi akan oleng : false")
        }
    }
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Mo
d8.go"
masukkan berat belanjaan di kedua kantong = 20 30
sepeda motor pak andi akan oleng : true
```

Deskripsi Program

Program akan terus meminta pengguna untuk memasukkan dua nilai berat (berat1_2311102021 dan berat2) yang mewakili berat belanjaan di dua kantong yang digendong Pak Andi. Proses input ini akan terus berulang sampai program dihentikan secara manual. akan memeriksa apakah salah satu atau kedua nilai berat yang dimasukkan lebih besar atau sama dengan 9.

XII. UNGUIDED

Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func hitungF(k int) float64 {
    pembilang_2311102021 := math.Pow(float64(4*k+2),
2)
    penyebut := float64((4*k+1)*(4*k+3))
    return pembilang_2311102021 / penyebut
}

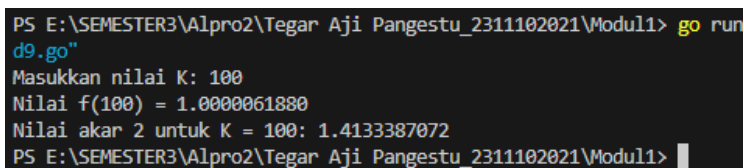
func hitungAkarDua(k int) float64 {
    hasil := 1.0
    for i := 0; i <= k; i++ {
        hasil *= hitungF(i)
    }
    return hasil
}

func main() {
    var k int
    fmt.Print("Masukkan nilai K: ")
    fmt.Scan(&k)

    nilaiF := hitungF(k)
    fmt.Printf("Nilai f(%d) = %.10f\n", k, nilaiF)

    akarDua := hitungAkarDua(k)
    fmt.Printf("Nilai akar 2 untuk K = %d:
%.10f\n", k, akarDua)
}
```

Screenshoot Output



```
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> go run
d9.go"
Masukkan nilai K: 100
Nilai f(100) = 1.0000061880
Nilai akar 2 untuk K = 100: 1.4133387072
PS E:\SEMESTER3\Alpro2\Tegar Aji Pangestu_2311102021\Modul1> █
```

Deskripsi Program

menghitung nilai fungsi berdasarkan rumus yang diberikan. Rumus ini merupakan bagian dari suatu deret tak hingga yang digunakan untuk mendekati nilai akar dua. Nilai k yang dimasukkan oleh pengguna akan menentukan suku ke- k dalam deret tersebut. Menghitung Pendekatan Akar Dua: Fungsi `hitungAkarDua(k)` menggunakan hasil dari fungsi `hitungF` untuk menghitung pendekatan nilai akar dua. Fungsi ini melakukan perkalian berulang dari `hitungF(0)` hingga `hitungF(k)`. Semakin besar nilai k , semakin akurat pendekatan nilai akar duanya. Input dan Output: Dalam fungsi main, program meminta pengguna untuk memasukkan nilai k . Setelah itu, program menghitung nilai $f(k)$ dan pendekatan nilai akar dua berdasarkan nilai k yang dimasukkan. Hasil perhitungan kemudian ditampilkan ke layar dengan 10 angka di belakang koma.