

**LAPORAN PRAKTIKUM  
ALGORITMA PEMROGRAMAN 2  
MODUL II  
REVIEW STRUKTUR CONTROL**



Oleh:

Mansyuroh

NIM:

2311102234

**S1 TEKNIK INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2024**

## I. DASAR TEORI

### a. Struktur Program Go:

- Terdiri dari paket (package), impor pustaka (import), fungsi utama (func main), dan definisi fungsi atau variabel lain.
- Paket main menandakan program utama, dan func main() berisi kode utama.

### b. Menulis, Mengompilasi, dan Menjalankan Program Go:

- Buat file dengan ekstensi .go.
- Gunakan perintah go build untuk mengompilasi menjadi file eksekusi.
- Jalankan dengan perintah ./nama\_file di terminal atau langsung menggunakan go run nama\_file.go.

### c. Tipe Data Dasar dalam Go:

- Mencakup int, float64, bool, string.
- Struktur data kompleks seperti array, slice, map.

### d. Instruksi Dasar dalam Go:

- Pernyataan variabel (var), kondisi (if, switch), perulangan (for), dan pemanggilan fungsi (func).

## II. GUIDED

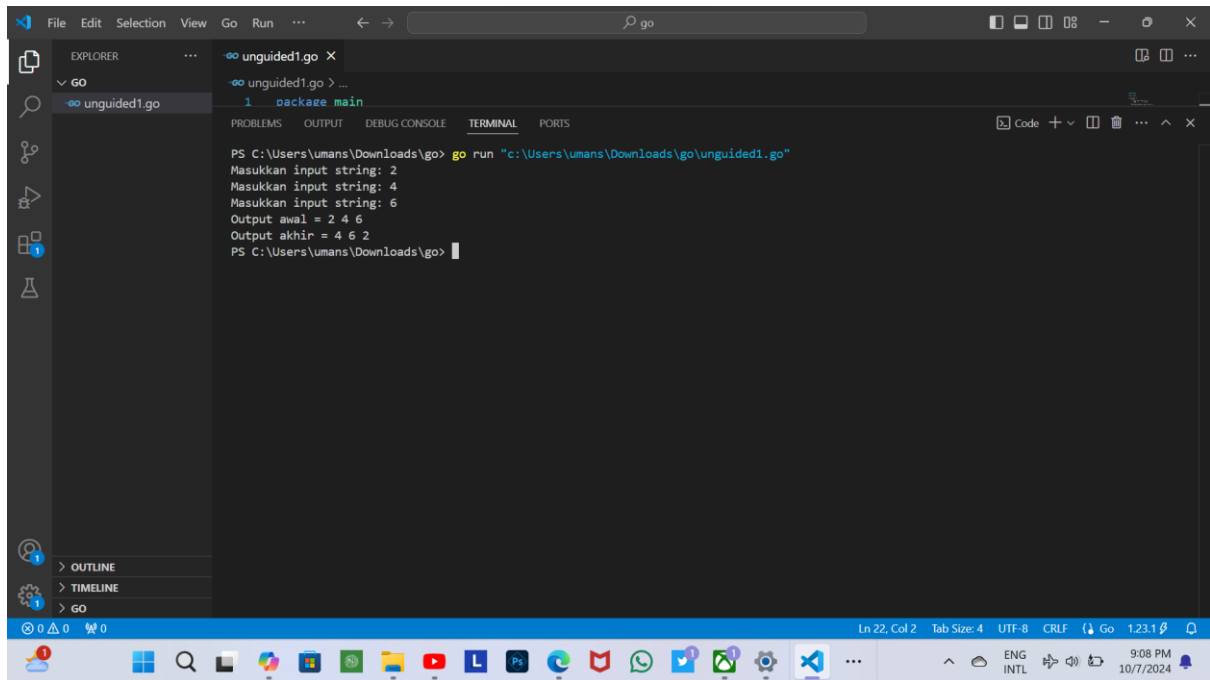
### 1. Source code

```
package main import
"fmt"

func main() { var (    satu, dua, tiga
string    temp string

)  fmt.Print("Masukan input string: ")  fmt.Scanln(&satu)  fmt.Print("Masukan
input string: ")  fmt.Scanln(&dua)  fmt.Print("Masukan input string: ")
fmt.Scanln(&tiga)  fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)  temp =
satu  satu = dua  dua = tiga  tiga = temp  fmt.Println("Output akhir = " + satu + " "
+ dua + " " + tiga) }
```

## Screenshoot program



```
File Edit Selection View Go Run ...  
EXPLORER  
GO  
unguided1.go  
1 package main  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided1.go"  
Masukkan input string: 2  
Masukkan input string: 4  
Masukkan input string: 6  
Output awal = 2 4 6  
Output akhir = 4 6 2  
PS C:\Users\umans\Downloads\go>  
Ln 22, Col 2 Tab Size: 4 UTF-8 CRLF 123.1
```

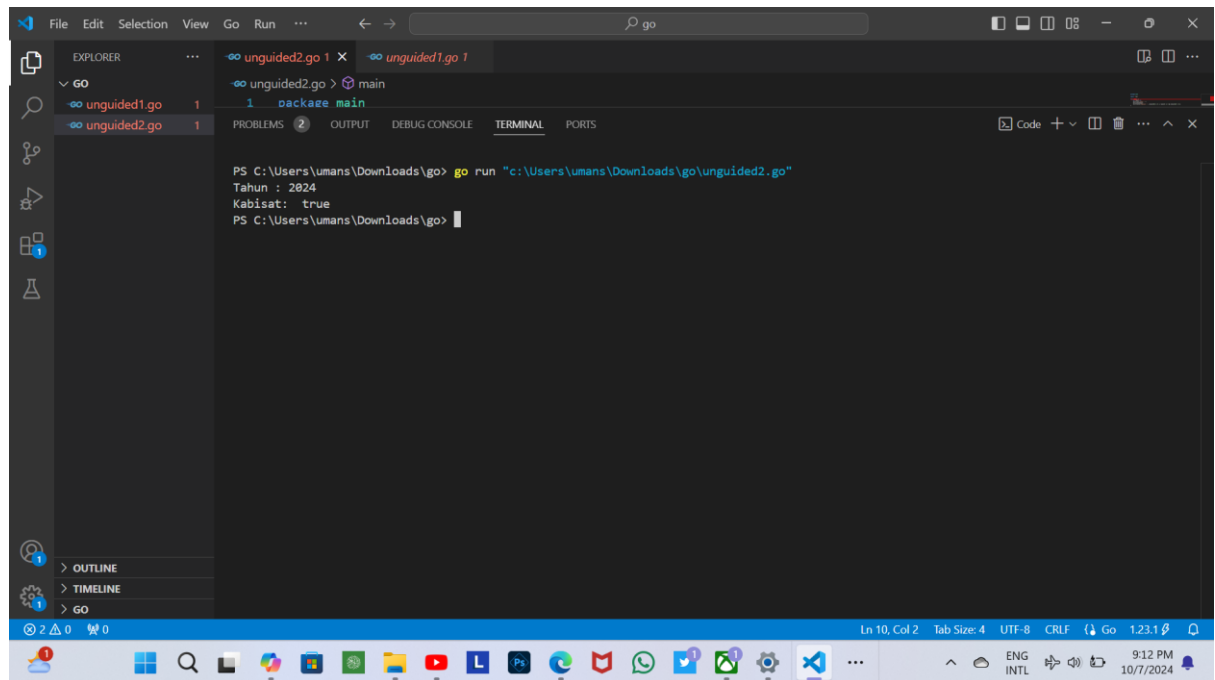
## Deskripsi program

Program ini meminta pengguna memasukkan tiga string, menukar nilainya menggunakan variabel sementara, dan menampilkan hasil penukarannya.

### 2. Source code

```
package main  
  
import "fmt"  
  
func main() {  
    year := 0  
    fmt.Scan(&year)    fmt.Println(year%4 == 0 &&  
year%100 != 0 || year%400 == 0)  
}
```

## Screenshoot program



```
File Edit Selection View Go Run ...  
EXPLORER  
go  
unguided1.go  
unguided2.go  
PROBLEMS  
OUTPUT  
DEBUG CONSOLE  
TERMINAL  
PORTS  
Code  
Ln 10, Col 2 Tab Size: 4 UTF-8 CRLF Go 1.23.1  
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided2.go"  
Tahun : 2024  
Kabisat: true  
PS C:\Users\umans\Downloads\go>
```

## Deskripsi program

Tahun kabisat adalah tahun yang habis dibagi 4, kecuali jika habis dibagi 100, kecuali jika juga habis dibagi 400. Syarat yang tepat untuk menentukan tahun kabisat adalah  $\text{year} \% 4 == 0 \ \&\& \ \text{year} \% 100 != 0 \ || \ \text{year} \% 400 == 0$ .

### 3. Source code

```
package main
import "fmt"

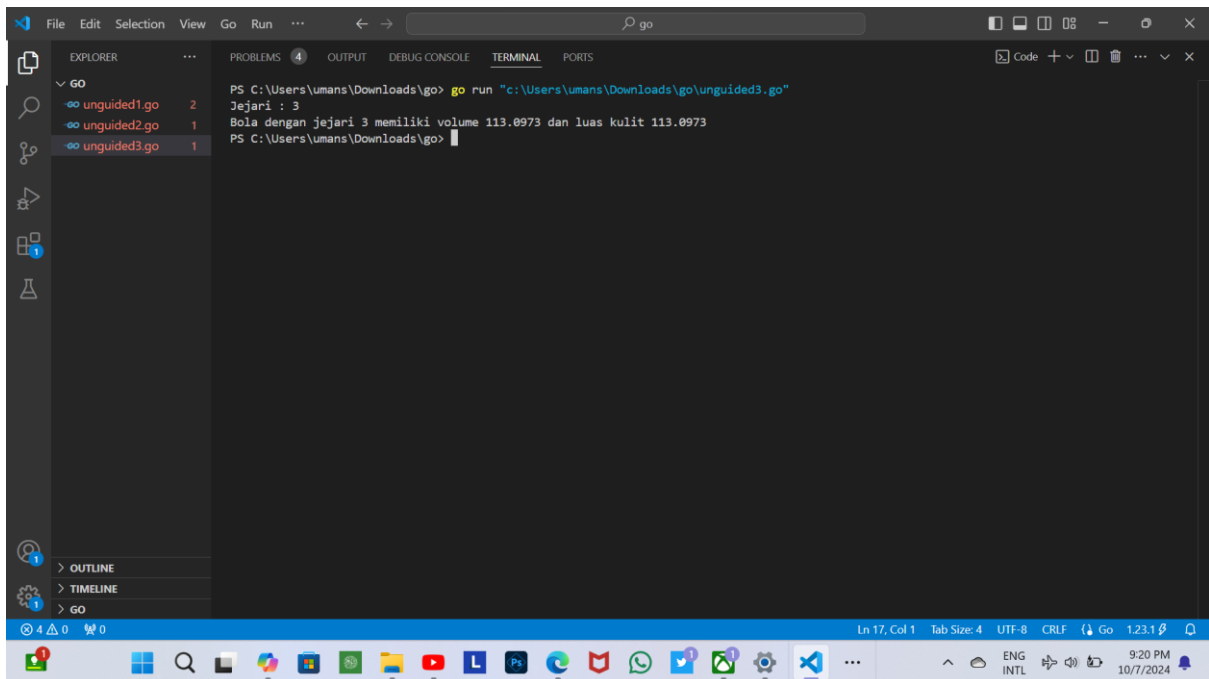
func main(){
    var r, phi, volume, luas float64

    fmt.Print("Jejari : ")
    fmt.Scan(&r)

    phi = 3.1415926535
    volume = 4.0 / 3.0 * phi * r * r * r
    luas = 4 * phi * r * r

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

### Screenshoot program

The screenshot shows the Visual Studio Code interface with a Go project. The Explorer pane on the left shows a folder named 'GO' containing three files: 'unguided1.go', 'unguided2.go', and 'unguided3.go'. The Terminal pane on the right shows the command 'go run "c:\Users\umans\Downloads\go\unguided3.go"' being executed in a PowerShell prompt. The output of the program is displayed in the terminal: 'Jejari : 3', 'Bola dengan jejari 3 memiliki volume 113.0973 dan luas kulit 113.0973'. The status bar at the bottom indicates the current line is 17, column 1, with a tab size of 4, UTF-8 encoding, and CRLF line endings.

### Deskripsi program

Program ini menggunakan paket math untuk operasi pangkat dan konstanta  $\pi = 3.1415926535$  untuk menghitung volume dan luas permukaan bola. Input jari-jari dari pengguna disimpan dalam

variabel jejari. Formula volume bola adalah  $(4/3) * \pi * r^3$  dan luas permukaan bola adalah  $4 * \pi * r^2$ . Program akan menghitung volume dan luas permukaan bola secara otomatis berdasarkan jari-jari yang diberikan.

### III. UNGUIDED

#### 1. Source code

```
package main
import "fmt"

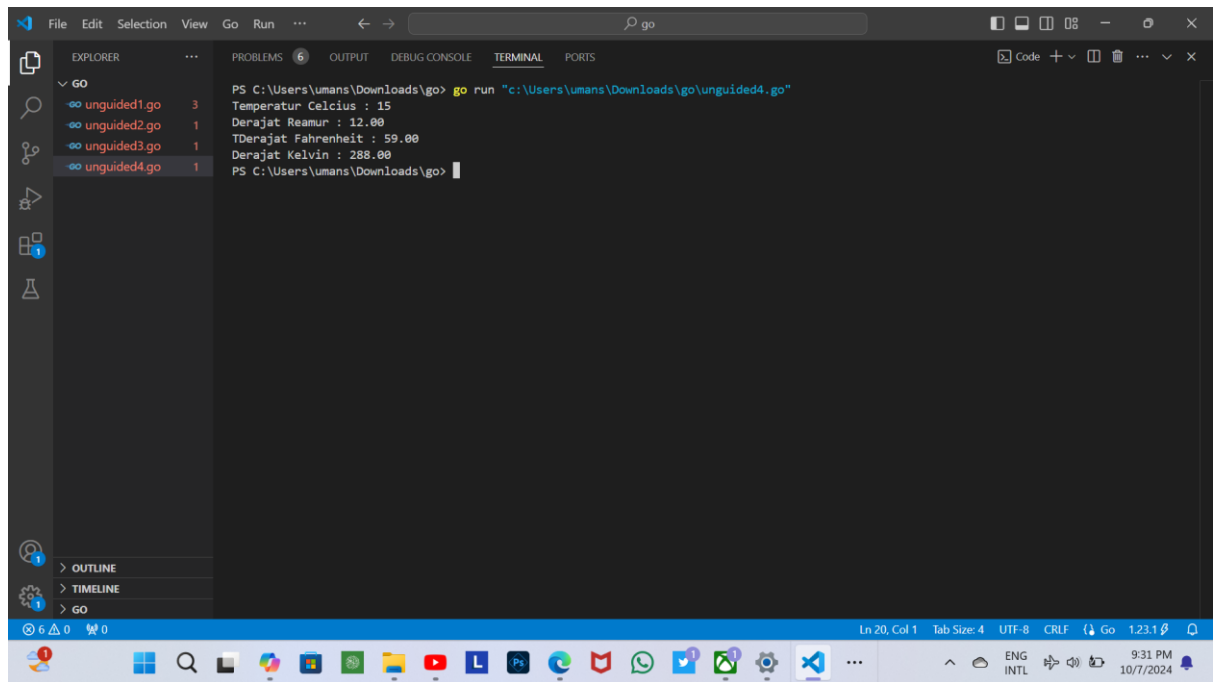
func main() {
    var R, F, K, suhu float64

    fmt.Print("Temperatur Celcius : ")
    fmt.Scan(&suhu)

    R = suhu * ( 4.00 / 5.00 )
    F = ( 9.00 / 5.00 ) * suhu + 32
    K = suhu + 273

    fmt.Printf("Derajat Reamur : %.2f\n", R)
    fmt.Printf("TDerajat Fahrenheit : %.2f\n", F)
    fmt.Printf("Derajat Kelvin : %.2f\n", K)
}
```

## Screenshoot program



```
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided4.go"
Temperatur Celcius : 15
Derajat Reamur : 12.00
Derajat Fahrenheit : 59.00
Derajat Kelvin : 288.00
PS C:\Users\umans\Downloads\go>
```

## Deskripsi program

Program ini meminta pengguna memasukkan temperatur dalam derajat Celsius, yang disimpan dalam variabel celsius. Temperatur kemudian dikonversi ke Fahrenheit, Reamur, dan Kelvin menggunakan rumus yang sesuai. Hasil konversi ditampilkan dengan dua angka di belakang koma, memastikan presisi yang baik.

## 2. Source code

```
package main

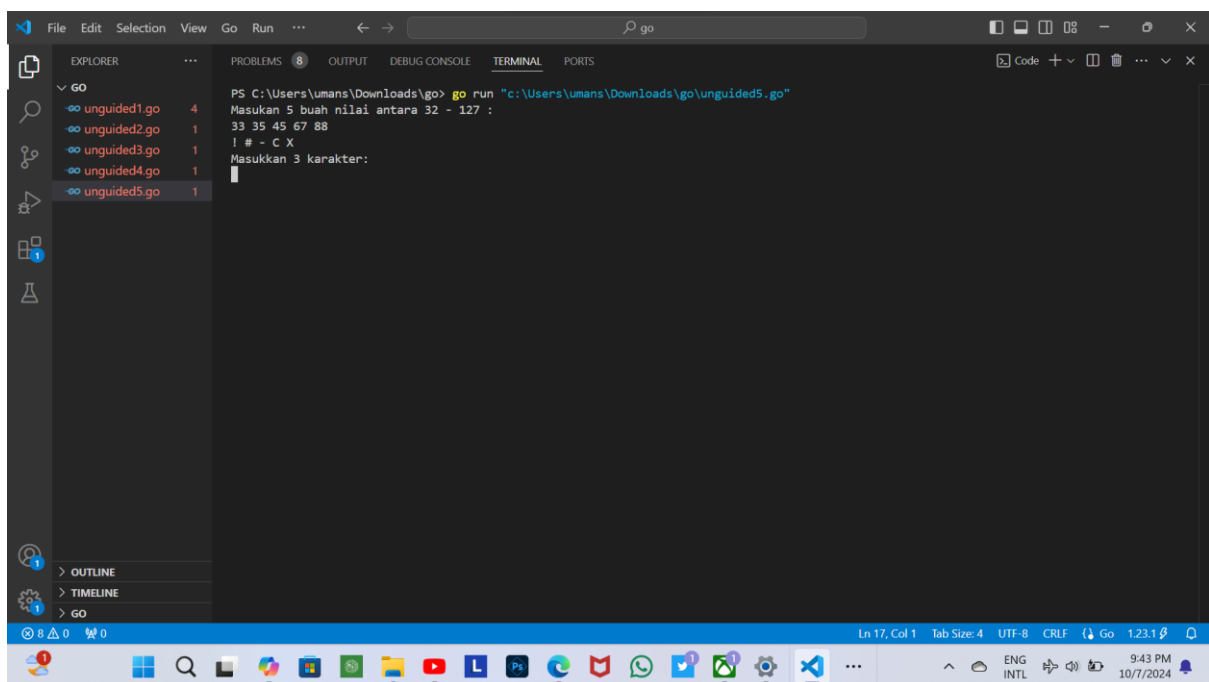
import "fmt"

func main() {
    var int1, int2, int3, int4, int5 int
    var char1, char2, char3 rune

    fmt.Println("Masukan 5 buah nilai antara 32 - 127 : ")
    fmt.Scan(&int1, &int2, &int3, &int4, &int5)
    fmt.Printf("%c %c %c %c %c\n", int1, int2, int3, int4,
int5)

    fmt.Println("Masukkan 3 karakter:")
    fmt.Scanf(" %c %c %c", &char1, &char2, &char3)
    fmt.Printf("%c%c%c\n", char1, char2, char3)
}
```

## Screenshoot program



## Deskripsi program



Program ini meminta pengguna memasukkan 5 integer dalam rentang 32 hingga 127, yang disimpan dalam array. Selanjutnya, program meminta 3 karakter, memperhatikan karakter newline saat pembacaan. Program mencetak karakter yang sesuai dengan nilai integer yang dimasukkan dan karakter satu posisi setelahnya dengan menambahkan 1 ke nilai ASCII. Output diformat menggunakan fmt.Printf, dan input karakter menggunakan fmt.Scanf.

### 3. Source code

```
package main

import "fmt"

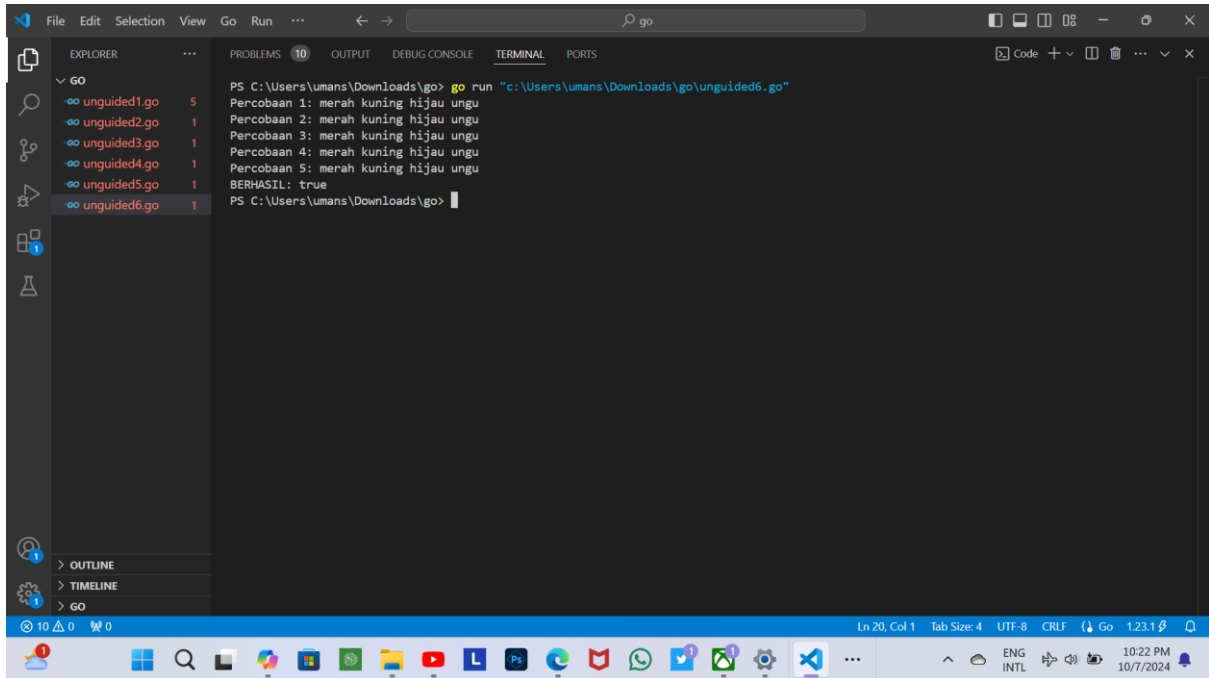
func main() {
    var warna [4]string
    var berhasil bool = true

    for i := 1; i <= 5; i++ {
        fmt.Printf("Percobaan %d: ", i)
        fmt.Scan(&warna[0], &warna[1], &warna[2], &warna[3])

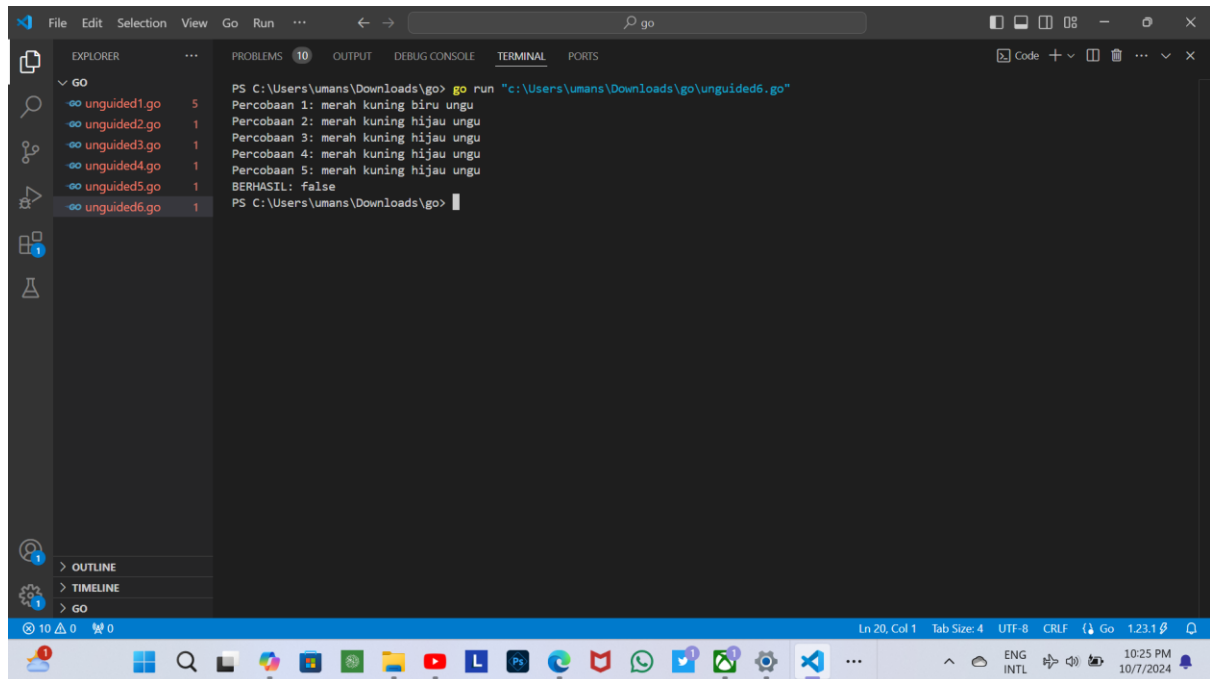
        if warna[0] != "merah" || warna[1] != "kuning" ||
warna[2] != "hijau" || warna[3] != "ungu" {
            berhasil = false
        }
    }

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

## Screenshot program



```
File Edit Selection View Go Run ...  
EXPLORER  
GO  
  unguided1.go 5  
  unguided2.go 1  
  unguided3.go 1  
  unguided4.go 1  
  unguided5.go 1  
  unguided6.go 1  
PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided6.go"  
Percobaan 1: merah kuning hijau ungu  
Percobaan 2: merah kuning hijau ungu  
Percobaan 3: merah kuning hijau ungu  
Percobaan 4: merah kuning hijau ungu  
Percobaan 5: merah kuning hijau ungu  
BERHASIL: true  
PS C:\Users\umans\Downloads\go>
```



```
File Edit Selection View Go Run ...  
EXPLORER  
GO  
  unguided1.go 5  
  unguided2.go 1  
  unguided3.go 1  
  unguided4.go 1  
  unguided5.go 1  
  unguided6.go 1  
PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided6.go"  
Percobaan 1: merah kuning biru ungu  
Percobaan 2: merah kuning hijau ungu  
Percobaan 3: merah kuning hijau ungu  
Percobaan 4: merah kuning hijau ungu  
Percobaan 5: merah kuning hijau ungu  
BERHASIL: false  
PS C:\Users\umans\Downloads\go>
```

## Deskripsi program

Program ini meminta pengguna untuk memasukkan 4 warna selama 5 percobaan dan memeriksa kesesuaiannya dengan "merah", "kuning", "hijau", dan "ungu". Outputnya menunjukkan apakah semua input berhasil (BERHASIL: true) atau tidak (BERHASIL: false).

## 4. Source code

```
package main

import (
    "fmt"
    "strings"
)

func main() {
    var pita string
    var count int
    var N int

    fmt.Print("N: ")
    fmt.Scanln(&N)

    if N > 0 {
        for i := 1; i <= N; i++ {
            var bunga string
            fmt.Printf("Bunga %d: ", i)
            fmt.Scanln(&bunga)
            pita += bunga + " - "
        }
        fmt.Println("Pita:", strings.TrimSpace(pita))
    } else {
        fmt.Println("Pita: ")
    }

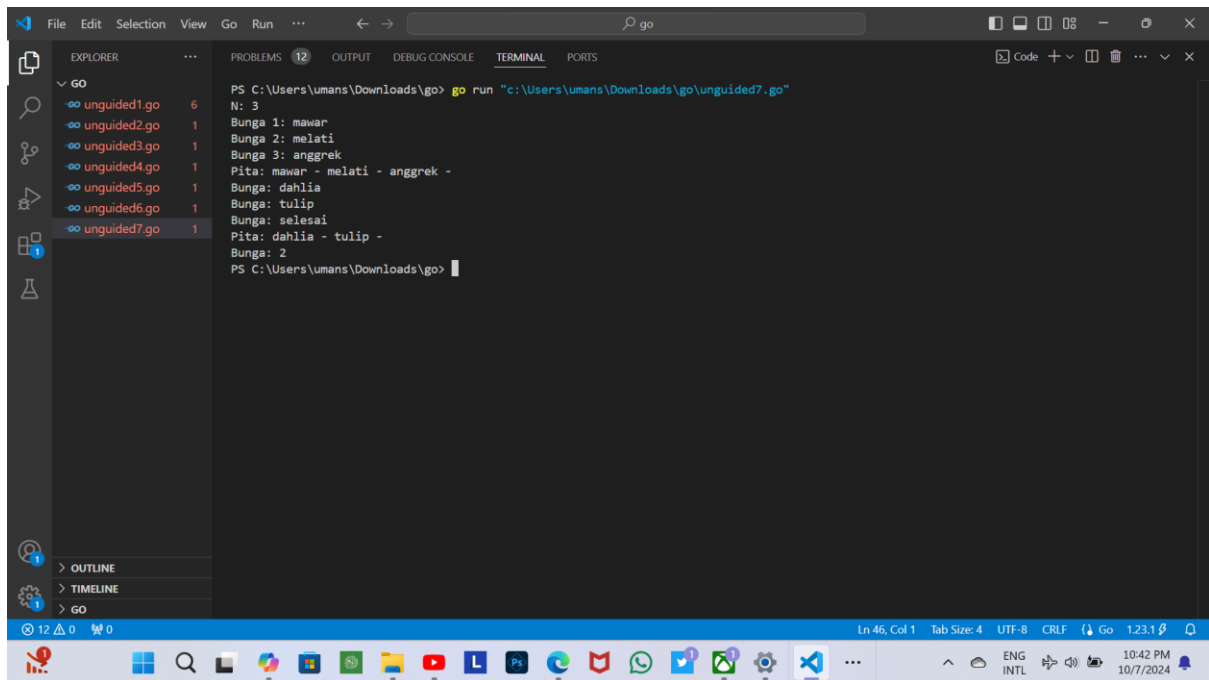
    pita = ""
    count = 0

    for {
        var bunga string
        fmt.Print("Bunga: ")
        fmt.Scanln(&bunga)

        if strings.ToUpper(bunga) == "SELESAI" {
            break
        }
        pita += bunga + " - "
        count++
    }

    fmt.Println("Pita:", strings.TrimSpace(pita))
    fmt.Printf("Bunga: %d\n", count)
}
```

## Screenshot program



```
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided7.go"
N: 3
Bunga 1: mawar
Bunga 2: melati
Bunga 3: anggrek
Pita: mawar - melati - anggrek -
Bunga: dahlia
Bunga: tulip
Bunga: selesai
Pita: dahlia - tulip -
Bunga: 2
PS C:\Users\umans\Downloads\go>
```

## Deskripsi program

Program ini meminta pengguna untuk memasukkan jumlah bunga dan nama bunga sebanyak N , lalu terus memasukan input bunga hingga "SELESAI" diketik. Program mencetak daftar bunga dan jumlah bunga yang dimasukkan.

## 5. Source code

```
package main

import "fmt"

func main() {
    var kantong1, kantong2 float64

    for {
        fmt.Print("Masukan berat belanjaan di kedua kantong: ")
        fmt.Scan(&kantong1, &kantong2)

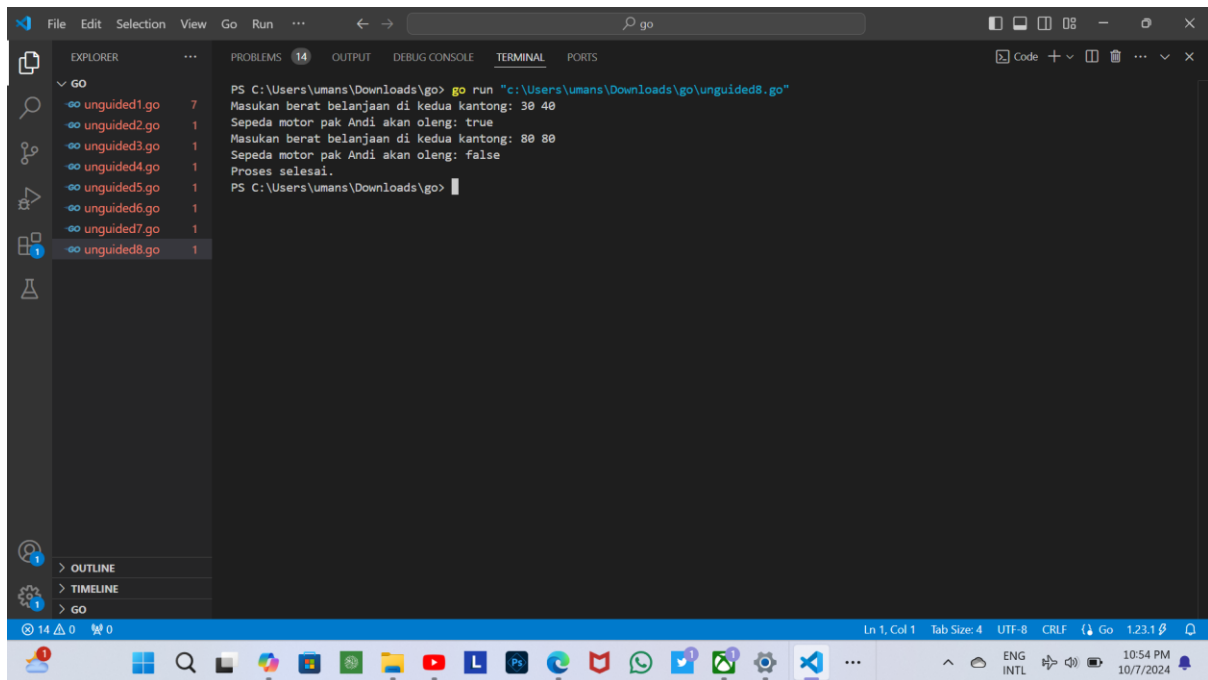
        if kantong1 < 0 || kantong2 < 0 {
            fmt.Println("Proses selesai.")
            break
        }

        selisih := kantong1 - kantong2
        if selisih < 0 {
            selisih = -selisih
        }

        if selisih >= 9 {
            fmt.Println("Sepeda motor pak Andi akan oleng: ")
        } else {
            fmt.Println("Sepeda motor pak Andi akan oleng: ")
        }

        totalBerat := kantong1 + kantong2
        if totalBerat > 150 {
            fmt.Println("Proses selesai.")
            break
        }
    }
}
```

## Screenshoot program



The screenshot shows the Visual Studio Code interface with a Go project. The Explorer panel on the left lists files: `GO`, `unguided1.go`, `unguided2.go`, `unguided3.go`, `unguided4.go`, `unguided5.go`, `unguided6.go`, `unguided7.go`, and `unguided8.go`. The Terminal panel on the right shows the execution of `go run "c:\Users\umans\Downloads\go\unguided8.go"`. The output is as follows:

```
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided8.go"
Masukan berat belanjaan di kedua kantong: 30 40
Sepeda motor pak Andi akan oleng: true
Masukan berat belanjaan di kedua kantong: 80 80
Sepeda motor pak Andi akan oleng: false
Proses selesai.
PS C:\Users\umans\Downloads\go>
```

## Deskripsi program

Program ini meminta pengguna untuk memasukkan berat belanjaan di dua kantong. Jika salah satu berat negatif, program akan berhenti. Program menghitung selisih berat dan menentukan apakah "Sepeda motor pak Andi akan oleng" berdasarkan selisih tersebut. Jika total berat melebihi 150, program juga berhenti.

## 6. Source code

```
package main

import (
    "fmt"
    "math"
)

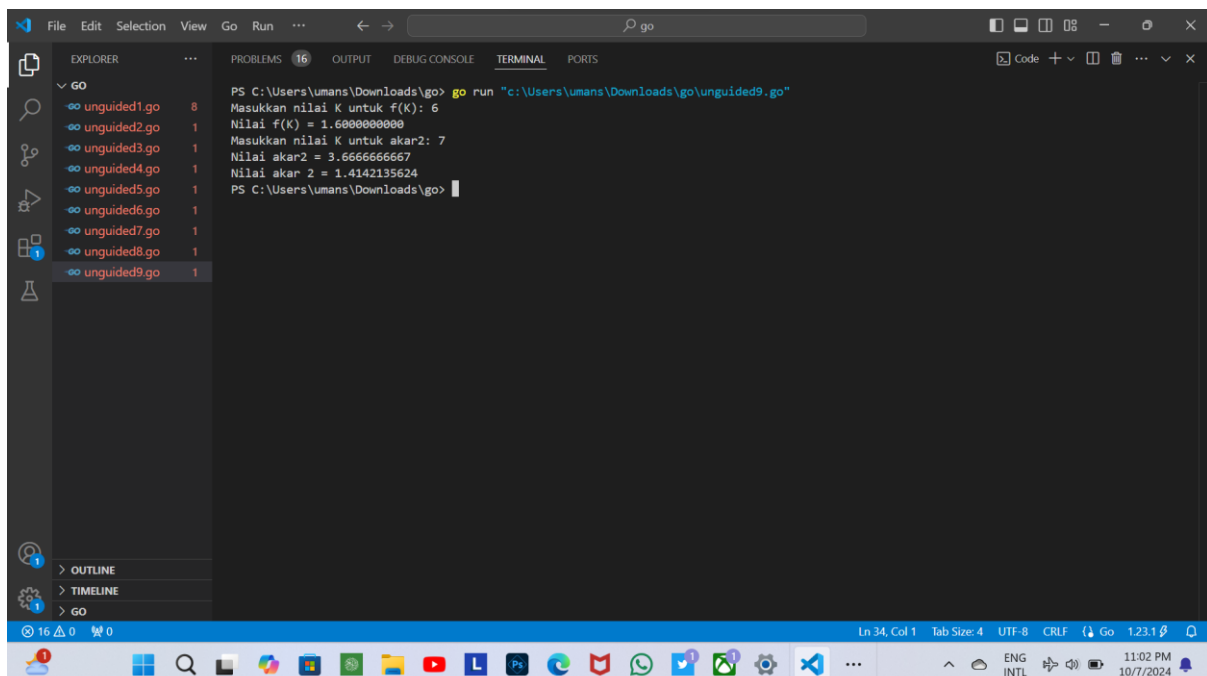
func main() {
    // Meminta input untuk nilai K dan menghitung f(K)
    var k float64
    fmt.Print("Masukkan nilai K untuk f(K): ")
    fmt.Scanln(&k)
    fK := calculateFK(k)
    fmt.Printf("Nilai f(K) = %.10f\n", fK)

    // Meminta input untuk nilai K dan menghitung akar2
    var K int
    fmt.Print("Masukkan nilai K untuk akar2: ")
    fmt.Scanln(&K)
    akar2 := calculateAkar2(K)
    fmt.Printf("Nilai akar2 = %.10f\n", akar2)
    fmt.Printf("Nilai akar 2 = %.10f\n", math.Sqrt(2))
}

// Fungsi untuk menghitung f(K)
func calculateFK(k float64) float64 {
    return (4 * k) / (k + k + 3)
}

// Fungsi untuk menghitung akar2
func calculateAkar2(K int) float64 {
    return (1 + float64(K) + 3) / (1 + 2)
}
```

## Screenshoot program



The screenshot shows the Visual Studio Code interface with a Go project. The Explorer pane on the left shows a file named `unguided9.go`. The Terminal pane on the right shows the output of the program when run. The output is as follows:

```
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided9.go"
Masukkan nilai K untuk f(K): 6
Nilai f(K) = 1.6000000000
Masukkan nilai K untuk akar2: 7
Nilai akar2 = 3.6666666667
Nilai akar 2 = 1.4142135624
PS C:\Users\umans\Downloads\go>
```

## Deskripsi program

Program ini meminta pengguna untuk memasukkan nilai K untuk menghitung fungsi dan nilai K lainnya untuk menghitung akar2. Dua fungsi terpisah digunakan untuk melakukan perhitungan, dan hasilnya ditampilkan dengan format yang tepat, termasuk nilai akar 2.

## 7. Source code

```
package main

import (
    "fmt"
)

func cariFaktor(b int) []int {
    var faktor []int
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            faktor = append(faktor, i)
        }
    }
    return faktor
}

func cekPrima(b int) bool {
    faktor := cariFaktor(b)
    // Bilangan prima harus memiliki tepat dua faktor: 1 dan b
    return len(faktor) == 2
}

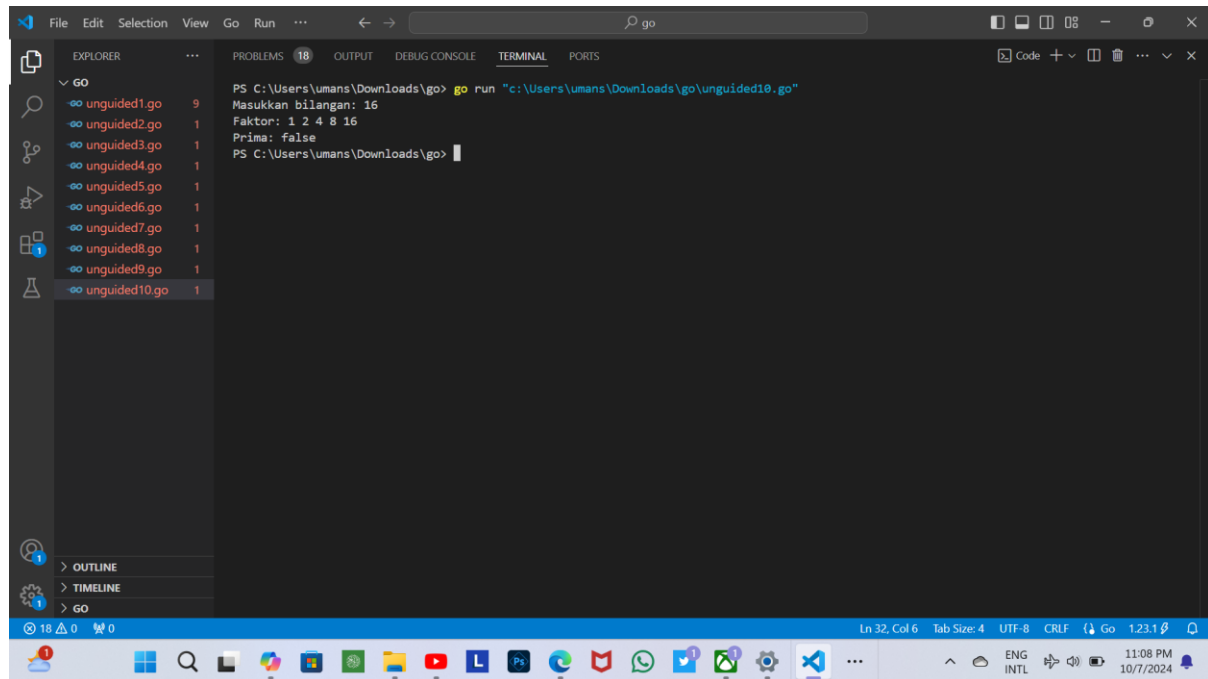
func main() {
    var b int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&b)

    faktor := cariFaktor(b)
    fmt.Printf("Faktor: ")
    for _, f := range faktor {
        fmt.Printf("%d ", f)
    }
    fmt.Println()

    if cekPrima(b) {
        fmt.Println("Prima: true")
    } else {
        fmt.Println("Prima: false")
    }
}
```



## Screenshoot program



```
File Edit Selection View Go Run ...  
EXPLORER  
GO  
  unguided1.go 9  
  unguided2.go 1  
  unguided3.go 1  
  unguided4.go 1  
  unguided5.go 1  
  unguided6.go 1  
  unguided7.go 1  
  unguided8.go 1  
  unguided9.go 1  
  unguided10.go 1  
PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\umans\Downloads\go> go run "c:\Users\umans\Downloads\go\unguided10.go"  
Masukkan bilangan: 16  
Faktor: 1 2 4 8 16  
Prima: false  
PS C:\Users\umans\Downloads\go>  
Ln 32, Col 6 Tab Size: 4 UTF-8 CRLF 1.23.1 10/7/2024
```

## Deskripsi program

Program ini meminta pengguna untuk memasukkan sebuah bilangan, kemudian menghitung dan menampilkan semua faktor dari bilangan tersebut. Program juga memeriksa apakah bilangan tersebut adalah bilangan prima dan mencetak hasilnya.







