

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II  
REVIEW STRUKTUR KONTROL**



**Disusun Oleh :**

**Rasyid Nafsyarie / 2311102011**

**IF-11-06**

**Dosen Pengampu :**

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

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## **I. DASAR TEORI**

Struktur kontrol dalam pemrograman adalah mekanisme yang mengatur alur eksekusi program. Dalam Golang, terdapat beberapa struktur kontrol yang penting untuk dipahami:

### **Pernyataan Kondisional**

If-Else Pernyataan if-else digunakan untuk mengeksekusi blok kode berdasarkan kondisi tertentu. Switch digunakan untuk memilih satu dari beberapa blok kode yang akan dieksekusi.

### **Pernyataan Perulangan**

For loop adalah satu-satunya jenis perulangan dalam Golang, namun sangat fleksibel.

Range digunakan untuk melakukan iterasi pada struktur data seperti array, slice, map, atau string.

### **Pernyataan Kontrol Tambahan**

Break digunakan untuk menghentikan eksekusi loop atau switch secara prematur.

Continue digunakan untuk melanjutkan ke iterasi berikutnya dalam loop.

Goto meskipun jarang digunakan, goto memungkinkan lompatan ke label yang ditentukan dalam kode.

### **Kesimpulan**

Struktur kontrol dalam Golang menyediakan cara yang efisien dan fleksibel untuk mengatur alur program. Pemahaman yang baik tentang struktur kontrol ini sangat penting dalam pengembangan aplikasi yang efektif dan efisien menggunakan Golang.

## II. GUIDED

1. Berisi source code dan output dari kegiatan praktikum yang telah dilaksanakan. Source Code diberi penjelasan maka akan menjadi nilai ++

### Sourcecode

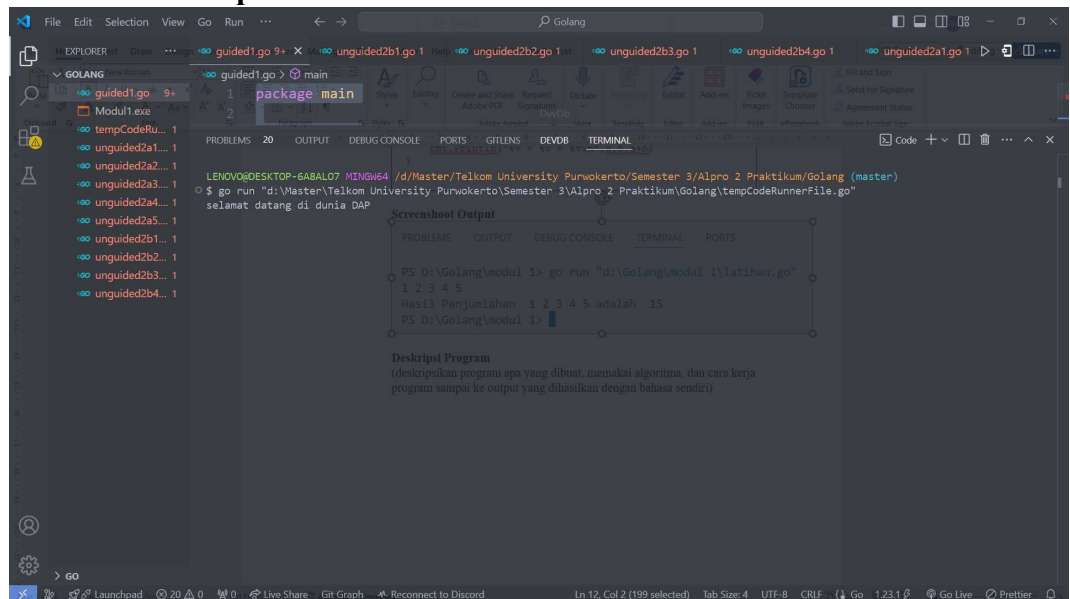
```
package main

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



### Deskripsi Program

Di dalam fungsi main, kita mendeklarasikan variabel greetings yang berisi pesan sambutan dan dua variabel integer a dan b untuk menyimpan angka yang akan dijumlahkan.

2. Berisi source code dan output dari kegiatan praktikum yang telah dilaksanakan. Source Code diberi penjelasan maka akan menjadi nilai ++

### Sourcecode

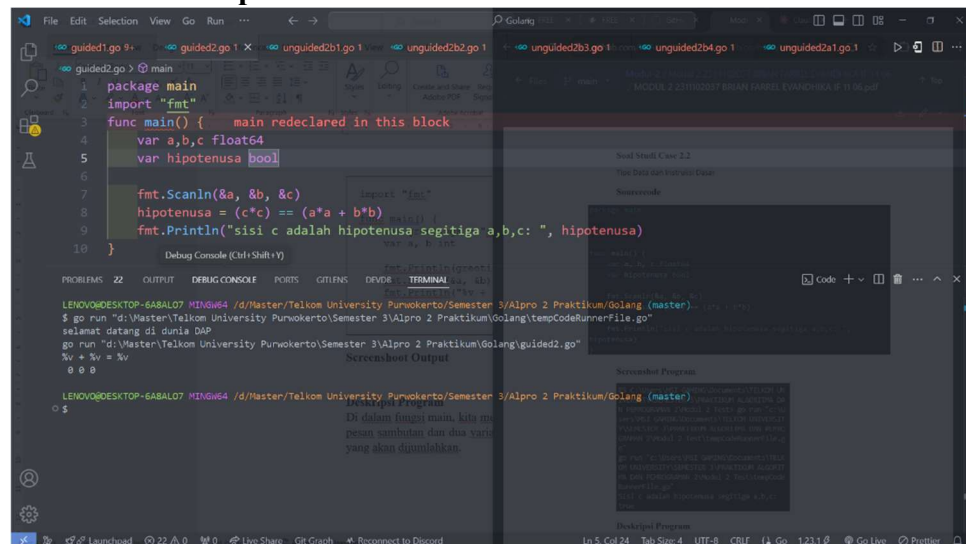
```
package main
```

```
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



## Deskripsi Program

Dalam kode ini, kita menggunakan tipe data float64 untuk menyimpan nilai sisi segitiga, yang memungkinkan kita untuk menggunakan angka desimal. Kode ini juga memeriksa apakah sisi c merupakan hipotenusa dari segitiga yang dibentuk oleh sisi a dan b.

3. Berisi source code dan output dari kegiatan praktikum yang telah dilaksanakan. Source Code diberi penjelasan maka akan menjadi nilai ++

## 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

The screenshot shows a Go IDE with the following code in the editor:

```

func main() {
    f1 := 1
    f2 := 1

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

```

The terminal output shows the execution of the program:

```

$ go run "d:\Master\Telkom University Purwokerto\Semester 3\Alpro 2 Praktikum\Golang\guided3.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

Kode ini berfungsi untuk menghitung deret Fibonacci hingga 100, Di dalam loop, kita memperbarui nilai f0, f1, dan f2 untuk menghitung angka Fibonacci berikutnya dan mencetaknya.

### III. UNGUIDED

#### Soal Studi Case 2A 1

Telusuri program berikut dengan cara mengkompilasi dan mengeksekusi program. Silakan masukan data yang sesuai sebanyak yang diminta program. Perhatikan keluaran yang diperoleh. Coba terangkan apa sebenarnya yang dilakukan program tersebut?

```
1 package main
2 import "fmt"
3
4 func main() {
5     var (
6         satu, dua, tiga string
7         temp string
8     )
9     fmt.Print("Masukan input string: ")
10    fmt.Scanln(&satu)
11    fmt.Print("Masukan input string: ")
12    fmt.Scanln(&dua)
13    fmt.Print("Masukan input string: ")
14    fmt.Scanln(&tiga)
15    fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)
16    temp = satu
17    satu = dua
18    dua = tiga
19    tiga = temp
20    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
21 }
```

#### Sourcecode

```
package main

import "fmt"

func main() {
    var (
        satu_2311102011, dua, tiga string
        temp                string
    )

    fmt.Print("Masukan input string: ")
    fmt.Scanln(&satu_2311102011)
    fmt.Print("Masukan input string: ")
    fmt.Scanln(&dua)
    fmt.Print("Masukan input string: ")
    fmt.Scanln(&tiga)
    fmt.Println("Output awal = " + satu_2311102011 + " "
+ dua + " " + tiga)
    temp = satu_2311102011
    satu_2311102011 = dua
    dua = tiga
    tiga = temp
}
```

```

        fmt.Println("Output akhir = " + satu_2311102011 + "
" + dua + " " + tiga)
    }

```

## Screenshoot Output

The screenshot shows a Go IDE with the following code in the editor:

```

5 func main() {
6     satu_2311102011, dua, tiga string
7     temp ..... string
8 }
9
10
11
12 fmt.Println("Masukan input string: ")
13 fmt.Scanln(&satu_2311102011)
14 fmt.Println("Masukan input string: ")
15 fmt.Scanln(&dua)
16 fmt.Println("Masukan input string: ")
17 fmt.Scanln(&tiga)

```

The terminal output shows the execution of the program:

```

$ go run "d:\Master\Telkom University Purwokerto\Semester 3\Alpro 2 Praktikum\Golang\tempCodeRunnerFile.go"
Masukan input string: Rasyid
Masukan input string: Nafsyarie
Masukan input string: Ganteng
Output awal = Rasyid Nafsyarie Ganteng
Output akhir = Nafsyarie Ganteng Rasyid

```

## Deskripsi Program

Program ini bertujuan untuk menukar nilai dari tiga string yang dimasukkan oleh pengguna dan menampilkan hasil sebelum dan sesudah penukaran. Variabel sementara temp digunakan untuk memudahkan penukaran nilai-nilai antar variabel.

## Soal Studi Case 2A 2

### Sourcecode

```

package main

import "fmt"

func main() {
    var tahun_2311102011 int
    var kabisat bool

    fmt.Print("Tahun : ")
    fmt.Scanln(&tahun_2311102011)

```

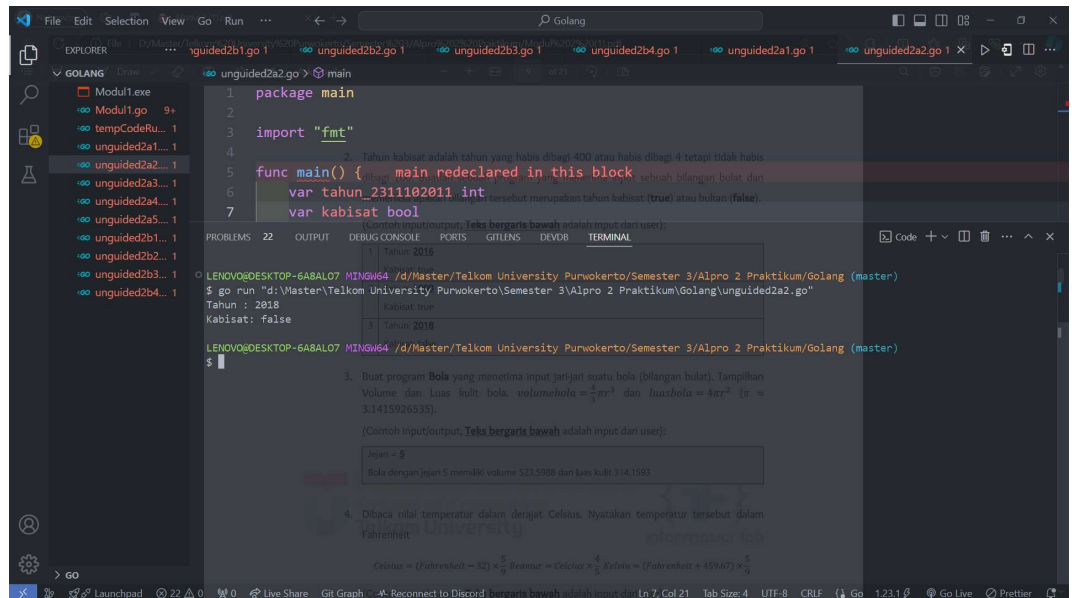
```

    if tahun_2311102011%400 == 0 {
        kabisat = true
    } else if tahun_2311102011%100 == 0 {
        kabisat = false
    } else if tahun_2311102011%4 == 0 {
        kabisat = true
    } else {
        kabisat = false
    }

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

```

## Screenshoot Output



## Deskripsi Program

Dengan menggunakan kode di atas, kita dapat dengan mudah menentukan apakah suatu tahun adalah tahun kabisat atau bukan.



## Soal Studi Case 2A 3

### Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var jejari_2311102011 float64
    const pi = 3.1415926535

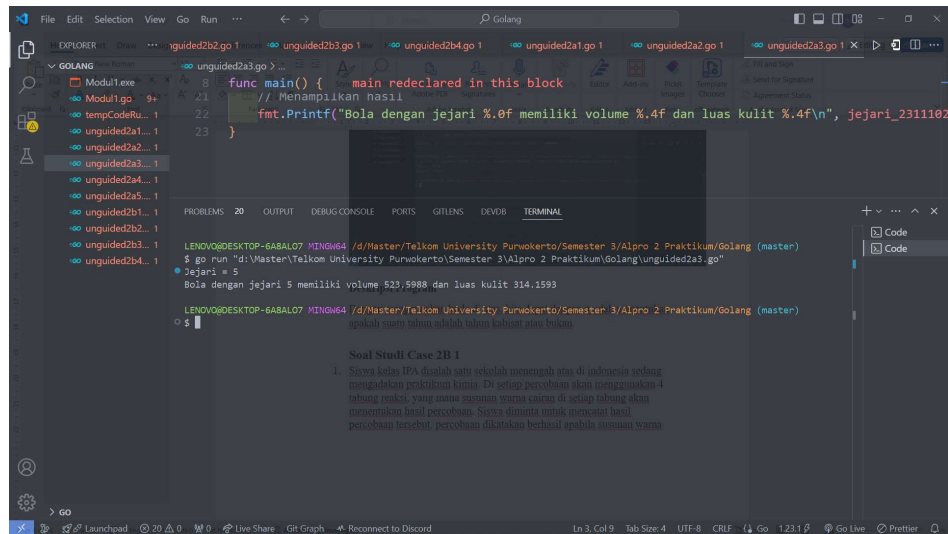
    fmt.Print("Jejari = ")
    fmt.Scanln(&jejari_2311102011)

    // Menghitung volume bola
    volume := (4.0 / 3.0) * pi *
    math.Pow(jejari_2311102011, 3)

    // Menghitung luas permukaan bola
    luas := 4 * pi * math.Pow(jejari_2311102011, 2)

    // Menampilkan hasil
    fmt.Printf("Bola dengan jejari %.0f memiliki volume
    %.4f dan luas kulit %.4f\n", jejari_2311102011, volume,
    luas)
}
```

## Screenshoot Output



## Deskripsi Program

Kode ini memberikan contoh yang jelas tentang bagaimana kita dapat menggunakan bahasa pemrograman Go untuk melakukan perhitungan matematis sederhana. Menggunakan rumus yang sudah ada untuk menghitung volume dan luas permukaan bola.

## Soal Studi Case 2A 4

### Sourcecode

```
package main

import (
    "fmt"
)

func main() {

    var celsius_2311102011 float64

    fmt.Print("Masukkan suhu dalam derajat Celsius: ")
    fmt.Scanf("%f", &celsius_2311102011)

    reaumur := (4.0 / 5.0) * celsius_2311102011

    fahrenheit := (celsius_2311102011 * 9.0 / 5.0) +
    32.0

    kelvin := celsius_2311102011 + 273.15
```

```

        fmt.Printf("Temperatur : %.2f\n",
celsius_2311102011)
        fmt.Printf("Derajat Reaumur: %.2f\n", reaumur)
        fmt.Printf("Derajat Fahrenheit: %.2f\n", fahrenheit)
        fmt.Printf("Derajat Kelvin: %.2f\n", kelvin)
    }
}

```

## Screenshoot Output

The screenshot shows a Go IDE with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The code editor displays a Go program that converts temperature from Celsius to Reaumur, Fahrenheit, and Kelvin. The terminal shows the execution of the program, which prompts the user to enter a temperature in Celsius. The user enters 121, and the program outputs the equivalent values in Reaumur (9.60), Fahrenheit (53.60), and Kelvin (285.15).

```

package main

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

func main() {
    // ... (code for temperature conversion) ...
}

```

```

$ go run "d:\Master\Telkom University Purwokerto\Semester 3\Alpro 2 Praktikum\Golang\unguided2a3.go"
Masukkan suhu dalam derajat Celsius: 121
Temperatur : 121.00
Derajat Reaumur: 9.60
Derajat Fahrenheit: 53.60
Derajat Kelvin: 285.15

```

## Deskripsi Program

Kode diatas adalah cara yang sederhana dan efektif untuk mengonversi suhu dari derajat Celsius ke berbagai satuan lainnya. Dengan menggunakan rumus yang telah dijelaskan sebelumnya, kode ini menghitung nilai suhu dalam Reaumur, Fahrenheit, dan Kelvin.

## Soal Studi Case 2A 5

### Sourcecode

```

package main

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

func main() {
    // ... (code for temperature conversion) ...
}

```

```

var a, b, c, d, e int
reader := bufio.NewReader(os.Stdin) // Membuat
reader untuk menangani input karakter

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

// Mencetak hasil karakter dari input angka
(berdasarkan tabel ASCII)
fmt.Printf("Hasil karakter dari angka yang diinput:
%c%c%c%c%c\n", a, b, c, d, e)

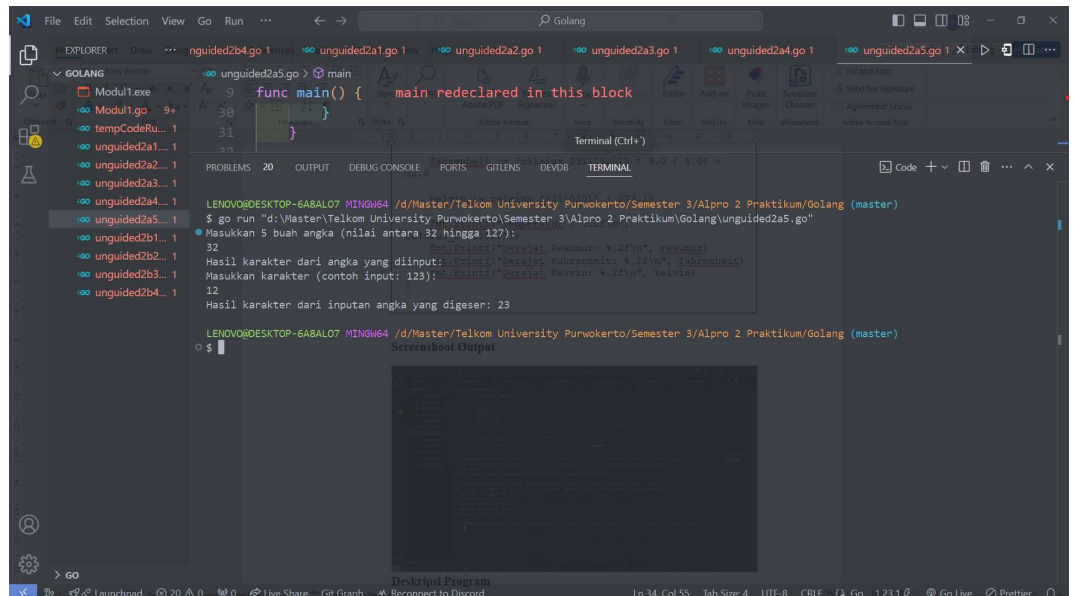
// Meminta input karakter berupa angka "123" dari
pengguna
fmt.Println("Masukkan karakter (contoh input:
123):")
inputStr, _ := reader.ReadString('\n') // Membaca
input berupa string sampai newline

// Menggeser karakter sesuai dengan keinginan
(contoh: 1 -> 2, 2 -> 3, 3 -> 4)
shiftedStr := ""
for _, char := range inputStr {
    // Skip newline or other non-visible characters
    if char != '\n' && char != '\r' {
        shiftedStr += string(char + 1) // Menggeser
setiap karakter
    }
}

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

```

## Screenshoot Output



```
func main() {  
    // ...  
    fmt.Println("Masukkan 5 buah angka (nilai antara 32 hingga 127):")  
    for i := 0; i < 5; i++ {  
        fmt.Scanln(&num)  
        if num < 32 || num > 127 {  
            fmt.Println("Nilai tidak valid. Masukkan angka antara 32 hingga 127.")  
            continue  
        }  
        char := string(num) // Konversi angka ke karakter ASCII  
        result += char  
    }  
    fmt.Println("Hasil karakter dari inputan angka yang digeser: ", result)  
}
```

Terminal Output:

```
LENOVODESKTOP-6A8AL07 MINGW64 /d/Master/Telkom University Purwokerto/Semester 3/Alpro 2 Praktikum/GoLang (master)  
$ go run "d:\Master\Telkom University Purwokerto\Semester 3\Alpro 2 Praktikum\GoLang\unguided2a5.go"  
Masukkan 5 buah angka (nilai antara 32 hingga 127):  
32  
Hasil karakter dari angka yang diinput: 0  
Masukkan karakter (contoh input: 123):  
123  
Hasil karakter dari inputan angka yang digeser: 07  
Masukkan karakter (contoh input: 123):  
123  
Hasil karakter dari inputan angka yang digeser: 07n  
Masukkan karakter (contoh input: 123):  
12  
Hasil karakter dari inputan angka yang digeser: 07n5  
Masukkan karakter (contoh input: 123):  
123  
Hasil karakter dari inputan angka yang digeser: 07n75  
Masukkan karakter (contoh input: 123):  
23  
Hasil karakter dari inputan angka yang digeser: 07n755
```

## Deskripsi Program

Program meminta pengguna untuk memasukkan lima angka yang nilainya harus berada dalam rentang 32 hingga 127. Angka-angka ini kemudian dikonversi menjadi karakter ASCII.

## Soal Studi Case 2B 1

1. Siswa kelas IPA disalah satu sekolah menengah atas di Indonesia sedang mengadakan praktikum kimia. Di setiap percobaan akan menggunakan 4 tabung reaksi, yang mana susunan warna cairan di setiap tabung akan menentukan hasil percobaan. Siswa diminta untuk mencatat hasil percobaan tersebut. Percobaan dikatakan berhasil apabila susunan warna zat cair pada gelas 1 hingga gelas 4 secara berturut-turut adalah 'merah', 'kuning', 'hijau', dan 'ungu' selama 5 kali percobaan berulang.

Buatlah sebuah program yang menerima input berupa warna dari ke 4 gelas reaksi sebanyak 5 kali percobaan. Kemudian program akan menampilkan true apabila urutan warna sesuai dengan informasi yang diberikan pada paragraf sebelumnya, dan false untuk urutan warna lainnya.

## Sourcecode

```
package main  
  
import (  
    "fmt"  
)
```

```

func main() {
    var warna_2311102011 [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_2311102011[i][0],
&warna_2311102011[i][1], &warna_2311102011[i][2],
&warna_2311102011[i][3])
    }

    for i := 0; i < 5; i++ {
        fmt.Printf("percobaan %d: %s %s %s %s\n", i+1,
warna_2311102011[i][0], warna_2311102011[i][1],
warna_2311102011[i][2], warna_2311102011[i][3])
        if warna_2311102011[i] != urutanBenar {
            berhasil = false
        }
    }

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

```

## Screenshoot Output

```

LENOVO@DESKTOP-GABAL07 MINGW64 /d:/Master/Telkom University Purwokerto/Semester 3/AIpro 2 Praktikum/Golang (master)
$ go run "d:/Master/Telkom University Purwokerto/Semester 3/AIpro 2 Praktikum/Golang/unguided2b1.go"
Masukkan warna untuk percobaan 1 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 2 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 3 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 4 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 5 (pisahkan dengan spasi): merah kuning hijau ungu
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

LENOVO@DESKTOP-GABAL07 MINGW64 /d:/Master/Telkom University Purwokerto/Semester 3/AIpro 2 Praktikum/Golang (master)
$ go run "d:/Master/Telkom University Purwokerto/Semester 3/AIpro 2 Praktikum/Golang/unguided2b1.go"
Masukkan warna untuk percobaan 1 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 2 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 3 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 4 (pisahkan dengan spasi): ungu kuning hijau merah
Masukkan warna untuk percobaan 5 (pisahkan dengan spasi): merah kuning hijau ungu
percobaan 1: merah kuning hijau ungu
percobaan 2: merah kuning hijau ungu
percobaan 3: merah kuning hijau ungu
percobaan 4: ungu kuning hijau merah
percobaan 5: merah kuning hijau ungu
berhasil : false

LENOVO@DESKTOP-GABAL07 MINGW64 /d:/Master/Telkom University Purwokerto/Semester 3/AIpro 2 Praktikum/Golang (master)
$ go run "d:/Master/Telkom University Purwokerto/Semester 3/AIpro 2 Praktikum/Golang/unguided2b1.go"
Masukkan warna untuk percobaan 1 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 2 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 3 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 4 (pisahkan dengan spasi): merah kuning hijau ungu
Masukkan warna untuk percobaan 5 (pisahkan dengan spasi): merah kuning hijau ungu
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

```

## Deskripsi Program

Kode ini Menggunakan loop for untuk meminta pengguna memasukkan warna. Input diambil dengan `fmt.Scan`, yang akan menyimpan warna ke dalam array `warna`.

## Soal Studi Case 2B 2

2. Suatu pita(string) berisi kumpulan nama-nama bunga yang dipisahkan oleh spasi dan '\_', contoh pita diilustrasikan seperti berikut ini

Pita: mawar=melati-tulip-teratai-kamboja-anggrek

Buatlah sebuah program yang menerima input sebuah bilangan bulat positif(dan tidak nol) N, kemudian program akan meminta input berupa nama bunga secara berulang sebanyak N kali dan nama tersebut disimpan ke dalam pita.

Tampilkan isi pita setelah proses input selesai. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read)

N: <u>3</u> Bunga 1: <u>Kertas</u> Bunga 2: <u>Mawar</u> Bunga 3: <u>Tulip</u> Pita: Kertas - Mawar - Tulip -	N : <u>0</u> Pita :
---	------------------------

Modifikasi program sebelumnya, proses input akan berhenti apabila user mengetikkan 'SELESAI'. Kemudian tampilkan isi pita beserta banyaknya bunga yang ada di dalam pita perhatikan contoh sesi interaksi program seperti dibawah ini (teks bergaris bawah adalah input/read):

Bunga 1: <u>Kertas</u> Bunga 2: <u>Mawar</u> Bunga 3: <u>Tulip</u> Bunga 4: <u>SELESAI</u> Pita: Kertas - Mawar - Tulip - Bunga: 3	Bunga 1: <u>SELESAI</u> Pita : Bunga: 0
---	---

### Sourcecode

```
package main

import (
    "fmt"
    "strings"
)

func main() {
    var N_2311102011 int
    fmt.Print("Masukkan jumlah bunga (N): ")
    fmt.Scan(&N_2311102011)

    var pita strings.Builder
```

```

var bunga string
count := 0

for i := 1; i <= N_2311102011; 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

```

$ go run "d:\Master\Telkom University Purwokerto\Semester 3\Alpro 2 Praktikum\Golang\unguided2b2.go"
Masukkan jumlah bunga (N): 3
Bunga 1: Kertas
Bunga 2: Mawar
Bunga 3: Tulip
Pita: Kertas - Mawar - Tulip
Bunga: 3

```

## Deskripsi Program

Jadi, program ini seperti simulasi membuat rangkaian bunga, di mana pengguna bisa memasukkan berbagai jenis bunga beserta jumlah bunga



yang ada di dalamnya. Dengan menggunakan percabangan looping dan percabangan.

### Soal Studi Case 2B 3

3. Suatu pita(string) berisi kumpulan nama-nama bunga yang dipisahkan oleh spasi dan '\_', contoh pita diilustrasikan seperti berikut ini

Pita: mawar=melati-tulip-teratai-kamboja-anggrek

Buatlah sebuah program yang menerima input sebuah bilangan bulat positif(dan tidak nol) N, kemudian program akan meminta input berupa nama bunga secara berulang sebanyak N kali dan nama tersebut disimpan ke dalam pita.

Tampilkan isi pita setelah proses input selesai. Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read)

N: <u>3</u>	N : <u>0</u>
Bunga 1: <u>Kertas</u>	Pita :
Bunga 2: <u>Mawar</u>	
Bunga 3: <u>Tulip</u>	
Pita: Kertas - Mawar - Tulip -	

Modifikasi program sebelumnya, proses input akan berhenti apabila user mengetikkan 'SELESAI'. Kemudian tampilkan isi pita beserta banyaknya bunga yang ada di dalam pita perhatikan contoh sesi interaksi program seperti dibawah ini (teks bergaris bawah adalah input/read):

Bunga 1: <u>Kertas</u>	Bunga 1: <u>SELESAI</u>
Bunga 2: <u>Mawar</u>	Pita :
Bunga 3: <u>Tulip</u>	Bunga: 0
Bunga 4: <u>SELESAI</u>	
Pita: Kertas - Mawar - Tulip -	
Bunga: 3	

### Sourcecode

```
package main

import (
    "fmt"
)

func main() {
```

```

var berat1_2311102011, berat2 float64

for {
    fmt.Print("masukkan berat belanjaan di kedua
kantong = ")
    fmt.Scan(&berat1_2311102011, &berat2)

    if berat1_2311102011 >= 9 || berat2 >= 9 {
        fmt.Println("sepeda motor pak andi akan
oleng : true")
    } else {
        fmt.Println("sepeda motor pak andi akan
oleng : false")
    }
}
}

```

## Screenshoot Output

```

func main() {
    main_redeclared_in_this_block
    break
}
// 2311102011 Praktikum Algoritma dan Pemrograman 2
//
// if pita.Len() > 0 {
//     pita.WriteString(" - ")
// }
// pita.WriteString(bunga)
// count++
// }
//
// Sebuah program yang menerima input sebuah bilangan bulat positif (dan tidak nol).
// N, kemudian program akan meminta input berupa nama bunga secara berurutan sebanyak
// N kali.
// Contoh:
// Input: 3
// Output:
// Bunga 1: Kertas
// Bunga 2: Mawar
// Bunga 3: Tulip
// Pita: Kertas - Mawar - Tulip
// N: 3
// Pita: Kertas - Mawar - Tulip -
//
// Modifikasi program sebelumnya, proses input akan berhenti apabila user mengakhiri
// 'CTRL+C', kemudian tampilkan isi pita beserta banyaknya bunga yang ada di dalam pita.

```

## Deskripsi Program

Program ini akan terus berjalan dalam sebuah loop tanpa henti (karena menggunakan for tanpa kondisi).

## Soal Studi Case 2B 4

4. buatlah program yang menerima input sebuah bilangan sebagai  $K$ , kemudian menghitung dan menampilkan nilai  $f(K)$  sesuai persamaan di atas. perhatikan contoh sesi interaksi program seperti dibawah ini (teks bergaris bawah adalah input/read):

Nilai $K = \underline{100}$ Nilai $f(K) = 1.0000061880$
--

$\sqrt{2}$  merupakan bilangan irasional. Meskipun demikian, nilai tersebut dapat dihampiri dengan rumus berikut:

$$\sqrt{2} = \prod_{k=0}^{\infty} \frac{(4k+2)^2}{(4k+1)(4k+3)}$$

Modifikasi program sebelumnya yang menerima input integer  $K$  dan menghitung  $\sqrt{2}$  untuk  $K$  tersebut. Hampiran  $\sqrt{2}$  dituliskan dalam ketelitian 10 angka di belakang koma.

Perhatikan contoh sesi interaksi program seperti di bawah ini (teks bergaris bawah adalah input/read):

1	Nilai $K = \underline{10}$ Nilai akar 2 = 1.4062058441
2	Nilai $K = \underline{100}$ Nilai akar 2 = 1.4133387072
3	Nilai $K = \underline{1000}$ Nilai akar 2 = 1.4141252651

### Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func hitungF(k int) float64 {
    pembilang_2311102011 := math.Pow(float64(4*k+2),
2)
    penyebut := float64((4*k+1)*(4*k+3))
    return pembilang_2311102011 / 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

The screenshot shows a Go IDE with the following code in `main.go`:

```

1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func hitungF(k int) float64 {
9     pembilang := math.Pow(float64(4*k+2), 2)
10    penyebut := float64((4*k+1)*(4*k+3))
11    return pembilang / penyebut
12 }
13
14 func hitungAkarDua(k int) float64 {
15     hasil := 1.0
16     for i := 0; i <= k; i++ {
17         hasil *= hitungF(i)
18     }
19     return hasil
20 }
21
22 func main() {
23     var k int
24     fmt.Print("Masukkan nilai K: ")
25     fmt.Scan(&k)
26
27     nilaiF := hitungF(k)
28     fmt.Printf("Nilai f(%d) = %.10f\n", k, nilaiF)
29
30     akarDua := hitungAkarDua(k)
31     fmt.Printf("Nilai akar 2 untuk K = %d: %.10f\n", k, akarDua)
32 }

```

The terminal output shows the program being run with `K=100`:

```

$ go run "d:\Master\Telkom University Purwokerto\Semester 3\Alpro 2 Praktikum\Golang\unguided2b4.go"
Masukkan nilai K: 100
Nilai f(100) = 1.0000061880
Nilai akar 2 untuk K = 100: 1.4133387872

```

## Deskripsi Program

Program ini seperti kalkulator matematika khusus. Pengguna memasukkan satu angka, dan program akan menghitung dua hal yang berbeda menggunakan angka tersebut. Dengan menggunakan 2 function agar lebih fleksibel.