

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 14
KOMPOSISI**



**DISUSUN OLEH:
MOHAMMAD REYHAN ARETHA FATIN
103112400078
S1 IF-12-01**

**DOSEN:
Yohani Setiya Rafika Nur, M. Kom.**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024/2025**

DASAR TEORI

1. Pengantar Komposisi Struktur Kontrol

Dalam pemrograman, struktur kontrol adalah alat yang digunakan untuk mengontrol alur eksekusi program berdasarkan kondisi yang ada. Struktur kontrol dapat dibagi menjadi dua kategori utama: percabangan dan perulangan.

1.1 Percabangan digunakan untuk menentukan jalur eksekusi berdasarkan kondisi tertentu.

2.1 Perulangan digunakan untuk mengeksekusi suatu blok kode berulang kali selama kondisi tertentu terpenuhi.

Pada modul ini, kita akan menggabungkan kedua struktur kontrol ini untuk menyelesaikan masalah yang lebih kompleks. Hal ini dilakukan untuk meningkatkan pemahaman dan kemampuan dalam memecahkan masalah yang lebih rumit, di mana beberapa jenis struktur kontrol harus digunakan secara bersamaan.

CONTOH SOAL

1. Contoh 1

Source Code:

```
coso1 > go coso1.go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      var bilangan, j int
7      fmt.Scan(&bilangan)
8      for j = 1; j <= bilangan; j += 1 {
9          if j%2 != 0 {
10             fmt.Print(j, " ")
11         }
12     }
13 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA
3
1 3
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA
2
1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA
7
1 3 5 7
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA
10
1 3 5 7 9
```

Deskripsi Program:

Program ini bertujuan untuk mencetak bilangan ganjil dari 1 hingga nilai yang dimasukkan oleh pengguna. Program menggunakan fungsi `fmt.Scan` untuk menerima input berupa bilangan bulat (`bilangan`). Kemudian, program menggunakan perulangan `for` untuk iterasi dari 1 hingga nilai input tersebut, dengan pengecekan kondisi menggunakan operator modulus (`%`). Jika nilai iterasi `j` adalah bilangan ganjil (`j%2 != 0`), bilangan tersebut dicetak menggunakan `fmt.Print` dengan spasi sebagai pemisah.

2. Contoh 2

Source Code:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var b1, b2, b3, min, max int
7     fmt.Scan(&b1, &b2, &b3)
8     if b1 > b2 {
9         max = b1
10        min = b2
11    } else {
12        max = b2
13        min = b1
14    }
15    if max < b3 {
16        max = b3
17    }
18    if min > b3 {
19        min = b3
20    }
21    fmt.Println("terbesar", max)
22    fmt.Println("terkecil", min)
23 }
24
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM
1 2 3
terbesar 3
terkecil 1
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM
5 5 5
terbesar 5
terkecil 5
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAMING\PRAKTIKUM
12 31 -43
terbesar 31
terkecil -43
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> |
```

Deksripsi Program:

Program ini untuk menentukan nilai terbesar dan terkecil dari tiga bilangan bulat yang dimasukkan oleh pengguna. Program menerima input tiga bilangan melalui `fmt.Scan` dan membandingkan nilai-nilai tersebut untuk menentukan nilai maksimum dan minimum. Awalnya, perbandingan dilakukan antara bilangan pertama dan kedua untuk menentukan nilai awal maksimum dan minimum. Selanjutnya, bilangan ketiga dibandingkan dengan nilai maksimum dan minimum yang telah ditentukan sebelumnya untuk memastikan hasil akhir. Setelah proses perbandingan selesai, program mencetak nilai terbesar dan terkecil menggunakan `fmt.Println`.

3. Contoh 3

Source Code:

```
coso3 > go coso3.go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      var bilangan, j int
7      fmt.Scan(&bilangan)
8      for j = 1; j <= bilangan; j += 1 {
9          if bilangan%j == 0 {
10             fmt.Print(j, " ")
11         }
12     }
13 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA F
5
1 5
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA F
12
1 2 3 4 6 12
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA F
20
1 2 4 5 10 20
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA F
72
1 2 3 4 6 8 9 12 18 24 36 72
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14>
```

Deksripsi Program:

Program ini untuk mencetak semua bilangan pembagi dari sebuah bilangan yang dimasukkan oleh pengguna. Program menerima input berupa bilangan bulat melalui `fmt.Scan`. Selanjutnya, program menggunakan perulangan `for` untuk mengecek setiap bilangan dari 1 hingga nilai input tersebut. Jika bilangan tersebut dapat membagi nilai input tanpa sisa (dengan menggunakan operator modulus `%`), maka bilangan tersebut dianggap sebagai pembagi dan dicetak menggunakan `fmt.Print` dengan spasi sebagai pemisah. Output dari program ini adalah daftar bilangan pembagi dari nilai yang dimasukkan.

SOAL LATIHAN

1.

Source Code:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan, j, counter int
7     fmt.Scan(&bilangan)
8     for j = 0; j <= bilangan; j++ {
9         if j%2 != 0 {
10             counter++
11         }
12     }
13     fmt.Printf("TERDAPAT %d BILANGAN GANJIL", counter)
14 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAM
3
TERDAPAT 2 BILANGAN GANJIL
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAM
2
TERDAPAT 1 BILANGAN GANJIL
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAM
7
TERDAPAT 4 BILANGAN GANJIL
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAM
10
TERDAPAT 5 BILANGAN GANJIL
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14>
```

Deskripsi Program:

Program ini untuk menghitung jumlah bilangan ganjil dari 0 hingga bilangan yang dimasukkan oleh pengguna. Program meminta pengguna untuk memasukkan sebuah bilangan melalui `fmt.Scan`. Selanjutnya, program menggunakan perulangan `for` untuk memeriksa setiap bilangan dari 0 hingga bilangan yang dimasukkan. Jika bilangan tersebut adalah ganjil (dicek dengan kondisi `j%2 != 0`), maka variabel `counter` akan bertambah 1. Setelah perulangan selesai, program mencetak jumlah bilangan ganjil tersebut dengan menggunakan `fmt.Printf`. Output program ini menampilkan jumlah bilangan ganjil dalam rentang tersebut.

2.

Source Code:

```
latsol2 > go latsol2.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var b int
7      fmt.Scan(&b)
8      if b == 1 {
9          fmt.Println("BUKAN PRIMA")
10         return
11     }
12     isPrime := true
13     for i := 2; i*i <= b; i++ {
14         if b%i == 0 {
15             isPrime = false
16             break
17         }
18     }
19     if isPrime {
20         fmt.Println("PRIMA")
21     } else {
22         fmt.Println("BUKAN PRIMA")
23     }
24 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAMING\PR
5
PRIMA
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAMING\PR
12
BUKAN PRIMA
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAMING\PR
19
PRIMA
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA PROGRAMING\PR
72
BUKAN PRIMA
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> |
```

Deksripsi Program:

Program ini untuk memeriksa apakah bilangan yang dimasukkan oleh pengguna adalah bilangan prima atau bukan. Program meminta pengguna untuk memasukkan sebuah bilangan bulat melalui `fmt.Scan`. Jika bilangan tersebut adalah 1, program langsung mencetak "BUKAN PRIMA" dan menghentikan eksekusi. Untuk bilangan lainnya, program menggunakan perulangan untuk memeriksa apakah bilangan tersebut habis dibagi oleh bilangan lain dari 2 hingga bilangan itu sendiri. Jika ditemukan pembagi selain 1 dan bilangan itu sendiri, variabel `isPrime` diubah menjadi false, dan pemeriksaan dihentikan. Akhirnya, program mencetak "PRIMA" jika bilangan tersebut adalah prima, atau "BUKAN PRIMA" jika tidak.

3.

Source Coding:

```
latsol3 > go latsol3.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var gelas1, gelas2, gelas3, gelas4 string
7     isAllCorrect := true
8     for i := 1; i <= 5; i++ {
9         fmt.Scan(&gelas1, &gelas2, &gelas3, &gelas4)
10        if gelas1 != "merah" || gelas2 != "kuning" ||
11        gelas3 != "hijau" || gelas4 != "ungu" {
12            isAllCorrect = false
13        }
14    }
15    fmt.Println(isAllCorrect)
16 }
```

Output:

```
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA
merah kuning hijau ungu
merah kuning hijau ungu
merah kuning hijau ungu
merah kuning hijau ungu
merah kuning hijau ungu
true
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> go run "d:\ALGORITMA
merah kuning hijau ungu
merah kuning hijau ungu
merah kuning hijau ungu
ungu kuning hijau merah
merah kuning hijau ungu
false
PS D:\ALGORITMA PROGRAMING\PRAKTIKUM\Praktikum modul 14> |
```

Deksripsi Program:

Program ini ditulis dalam bahasa Go untuk memeriksa apakah input warna dari empat variabel (gelas1, gelas2, gelas3, gelas4) sesuai dengan nilai yang diharapkan. Program meminta pengguna memasukkan nilai untuk keempat variabel tersebut sebanyak lima kali melalui perulangan for. Pada setiap iterasi, program memeriksa apakah nilai gelas1 adalah "merah", gelas2 adalah "kuning", gelas3 adalah "hijau", dan gelas4 adalah "ungu". Jika ada salah satu variabel yang nilainya tidak sesuai, variabel isAllCorrect diatur menjadi false. Setelah perulangan selesai, program mencetak nilai dari isAllCorrect, yang akan bernilai true jika semua input sesuai, atau false jika terdapat kesalahan.

DAFTAR PUSTAKA

Prasti Eko Yunanto, S.T., M.Kom.

**MODUL PRAKTIKUM 14 KOMPOSISI ALGORITMA DAN PEMOGRAMAN 1 S1
INFORMATIKA**