

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL XIV
“KOMPOSISI”



DISUSUN OLEH:
FEROS PEDROSA VALENTINO
103112400055
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

Komposisi adalah mekanisme yang memungkinkan penggabungan tipe yang lebih sederhana untuk menciptakan tipe yang lebih kompleks. Ini adalah konsep dasar dalam pemrograman berorientasi objek yang memungkinkan penggunaan kembali kode dan membangun program yang lebih fleksibel dan modular.

Overview Struktur Kontrol

- Perulangan :: Eksekusi perintah secara berulang-ulang.
- Percabangan :: Memilih perintah yang akan dieksekusi sesuai dengan kondisi tertentu.

Komposisi:

- Memungkinkan adanya komposisi perulangan dengan percabangan.
- Aksi dari perulangan ataupun percabangan bisa berupa perulangan atau percabangan yang lain.
- Contoh kasusnya:
Pencarian faktor bilangan, pencarian nilai terbesar pada kumpulan bilangan.

CONTOH SOAL

1. Coso1

Source code:

```
package main

import "fmt"

func main() {
    var bilangan, j int
    fmt.Scan(&bilangan)
    for j = 1; j <= bilangan; j += 1 {
        if j%2 != 0 {
            fmt.Print(j, " ")
        }
    }
}
```

Output:

```
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso1\coso1.go"
3
1 3
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso1\coso1.go"
2
1
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso1\coso1.go"
7
1 3 5 7
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso1\coso1.go"
10
1 3 5 7 9
```

Deskripsi program:

Program di atas ditulis dalam bahasa Go dan berfungsi untuk mencetak semua bilangan ganjil mulai dari 1 hingga nilai yang dimasukkan oleh pengguna. Pertama deklarasikan variabel bilangan bertipe integer yang digunakan untuk menyimpan inputan pengguna yang, variabel j bertipe integer yang digunakan untuk variabel penghitung didalam loop. Selanjutnya program meminta pengguna memasukkan suatu bilangan bulat. Lalu program menggunakan loop untuk iterasi dari satu hingga nilai bilangan. Di dalam loop, terdapat kondisi if yang memeriksa apakah nilai j adalah bilangan ganjil dengan menggunakan operator modulus. Jika j adalah bilangan ganjil ($j\%2$ tidak sama dengan 0), maka program akan mencetak nilai j diikuti dengan spasi.

2. Coso2

Source code:

```
package main

import "fmt"

func main() {
    var b1, b2, b3, max, min int
    fmt.Scan(&b1, &b2, &b3)
    if b1 > b2 {
        max = b1
        min = b2
    } else {
        max = b2
        min = b1
    }
    if max < b3 {
        max = b3
    }
    if min > b3 {
        min = b3
    }
    fmt.Println("Terbesar", max)
    fmt.Println("Terkecil", min)
}
```

Output:

```
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso2\coso2.go"
1 2 3
Terbesar 3
Terkecil 1
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso2\coso2.go"
5 5 5
Terbesar 5
Terkecil 5
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso2\coso2.go"
12 31 -43
Terbesar 31
Terkecil -43
```

Deskripsi program:

Program di atas ditulis dalam bahasa Go dan berfungsi untuk menentukan nilai terbesar dan terkecil dari tiga bilangan yang diinputkan oleh pengguna. Pertama deklarasikan variabel `b1`, `b2`, `b3` bertipe integer yang digunakan untuk menyimpan nilai inputan pengguna dan variabel `max`, `min` bertipe integer yang digunakan untuk menyimpan nilai maksimum dan minimum. Selanjutnya program meminta pengguna memasukkan tiga bilangan bulat. Lalu program menggunakan struktur kontrol `if` untuk membandingkan `b1` dan `b2`. Jika `b1` lebih besar dari `b2`, maka `max` diatur menjadi `b1` dan `min` menjadi `b2`. Sebaliknya, jika `b2` lebih besar atau sama dengan `b1`, maka `max` diatur menjadi `b2` dan `min` menjadi `b1`. Selanjutnya, program memeriksa apakah `max` lebih kecil dari `b3`, dan jika iya, maka `max` diupdate menjadi `b3`. Demikian pula, program memeriksa apakah `min` lebih besar dari `b3`, dan jika benar, maka `min` diupdate menjadi `b3`. Terakhir, program mencetak nilai terbesar dan terkecil yang telah ditentukan dengan menggunakan fungsi `fmt.Println`.

3. Coso3

Source code:

```
package main

import "fmt"

func main() {
    var bilangan, j int
    fmt.Scan(&bilangan)
    for j = 1; j <= bilangan; j += 1 {
        if bilangan%j == 0 {
            fmt.Print(j, " ")
        }
    }
}
```

Output:

```
● PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso3\coso3.go"
5
1 5
● PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso3\coso3.go"
12
1 2 3 4 6 12
● PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso3\coso3.go"
20
1 2 4 5 10 20
● PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\coso3\coso3.go"
72
1 2 3 4 6 8 9 12 18 24 36 72
```

Deskripsi program:

Program di atas ditulis dalam bahasa Go dan berfungsi untuk menampilkan semua faktor bilangan dari nilai yang diinputkan pengguna. Pertama deklarasi variabel bilangan yang bertipe integer digunakan untuk menyimpan inputan pengguna dan variabel j yang bertipe integer digunakan untuk variabel penghitung didalam loop. Selanjutnya program meminta pengguna untuk memasukkan suatu bilangan bulat positif. Lalu program menggunakan loop yang dimulai dari satu hingga nilai bilangan. Dalam setiap iterasi, program memeriksa apakah j adalah faktor dari bilangan dengan menggunakan kondisi if yang memeriksa apakah sisa pembagian bilangan dengan j sama dengan 0 ($\text{bilangan} \% j == 0$). Jika kondisi tersebut terpenuhi, maka j dicetak ke layar diikuti dengan spasi.

LATIHAN SOAL

1. Latihan soal 1

Source code:

```
package main

import "fmt"

func main() {
    var bilangan int
    fmt.Scan(&bilangan)

    if bilangan <= 0 {
        fmt.Println("Masukan harus bilangan bulat positif.")
    } else {
        jumlahGanjil := 0
        for i := 1; i <= bilangan; i++ {
            if i%2 != 0 {
                jumlahGanjil++
            }
        }

        fmt.Printf("Terdapat %d bilangan ganjil\n", jumlahGanjil)
    }
}
```

Output:

```
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol1\latsol1.go"
3
Terdapat 2 bilangan ganjil
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol1\latsol1.go"
2
Terdapat 1 bilangan ganjil
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol1\latsol1.go"
7
Terdapat 4 bilangan ganjil
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol1\latsol1.go"
10
Terdapat 5 bilangan ganjil
```

Deskripsi program:

Program di atas ditulis dalam bahasa Go dan berfungsi untuk memberikan contoh penggunaan kontrol alur dan perulangan dalam pemrograman. Pertama deklarasikan variabel bilangan bertipe integer. Lalu program meminta agar pengguna memasukkan suatu bilangan bulat positif. Setelah itu, program memeriksa apakah bilangan yang dimasukkan kurang dari atau sama dengan nol. Jika ya, program akan menampilkan pesan “Masukan harus bilangan bulat positif.”. Jika bilangan yang dimasukkan valid (positif), program kemudian menghitung jumlah bilangan ganjil dari 1 hingga bilangan yang dimasukkan. Ini dilakukan dengan menggunakan loop for yang iterasi dari 1 hingga bilangan tersebut, dan di dalam loop, program memeriksa apakah angka tersebut ganjil dengan menggunakan operator modulus. Setiap kali ditemukan bilangan ganjil, variabel jumlahGanjil

akan bertambah satu. Setelah loop selesai, program mencetak jumlah total bilangan ganjil yang ditemukan dalam rentang tersebut.

2. Latihan soal 2

Source code:

```
package main

import "fmt"

func main() {
    var bilangan int
    fmt.Scan(&bilangan)

    if bilangan <= 1 {
        fmt.Println("bukan prima")
    } else {
        prima := true
        for i := 2; i*i <= bilangan; i++ {
            if bilangan%i == 0 {
                prima = false
                break
            }
        }

        if prima {
            fmt.Println("prima")
        } else {
            fmt.Println("bukan prima")
        }
    }
}
```

Output:

```
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol2\latsol2.go"
5
prima
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol2\latsol2.go"
12
bukan prima
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol2\latsol2.go"
19
prima
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol2\latsol2.go"
72
bukan prima
```

Deskripsi program:

Program di atas ditulis dalam bahasa Go dan berfungsi untuk menentukan apakah sebuah bilangan bulat positif yang dimasukkan oleh pengguna adalah bilangan prima atau bukan prima. Pertama deklarasikan variabel bilangan bertipe integer. Lalu program meminta pengguna untuk memasukkan suatu bilangan bulat positif. Setelah itu program mengecek inputan apakah bilangan tersebut kurang dari atau sama dengan 1. Jika ya, program langsung mencetak “bukan prima” karena bilangan prima harus memiliki tepat dua faktor. Jika bilangan valid, program kemudian menetapkan variabel prima dengan nilai true, yang akan digunakan untuk menandakan apakah bilangan tersebut adalah prima. Selanjutnya, program menggunakan loop for untuk memeriksa faktor-faktor dari bilangan tersebut, mulai

dari 2 hingga akar kuadrat dari bilangan. Jika ditemukan bahwa bilangan tersebut dapat dibagi habis oleh salah satu angka dalam rentang tersebut, maka variabel `prima` diubah menjadi `false` dan loop dihentikan. Setelah loop selesai, program memeriksa nilai variabel `prima`. Jika masih `true`, program mencetak “prima”; jika tidak, program mencetak “bukan prima”.

3. Latihan soal 3

Source code:

```
package main

import "fmt"

func main() {
    var gelas1, gelas2, gelas3, gelas4 string

    isAllCorrect := true

    for i := 1; i <= 5; i++ {
        fmt.Printf("Percobaan %d: ", i)

        fmt.Scan(&gelas1, &gelas2, &gelas3, &gelas4)

        if gelas1 != "merah" || gelas2 != "kuning" || gelas3 != "hijau" || gelas4
        != "ungu" {
            isAllCorrect = false
        }
    }

    fmt.Println("BERHASIL:", isAllCorrect)
}
```

Output:

```
PS C:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol3\latsol3.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:\semester1_alpro1> go run "c:\semester1_alpro1\alpro1_week14\latsol3\latsol3.go"
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
```

Deskripsi program:

Program di atas ditulis dalam bahasa Go dan berfungsi untuk melakukan serangkaian percobaan input warna dari empat gelas. Dalam program ini, pengguna diminta untuk memasukkan warna dari empat gelas yang diharapkan berurutan sebagai “merah”, “kuning”, “hijau”, dan “ungu”. Program ini melakukan lima percobaan input, yang mana setiap percobaan meminta pengguna untuk memasukkan warna gelas satu per satu.

Pertama deklarasikan variabel gelas1, gelas2, gelas3, gelas4 bertipe string untuk menyimpan inputan pengguna serta variabel isAllCorrect yang ditetapkan dengan nilai true. Selanjutnya dengan loop yang berulang sebanyak 5 kali program akan meminta pengguna memasukkan warna tersebut dari percobaan 1 sampai percobaan 5. Jika dalam salah satu dari lima percobaan terdapat urutan warna yang salah (tidak sesuai dengan pola tersebut), maka variabel isAllCorrect diatur menjadi false. Setelah semua percobaan selesai, program menampilkan hasil akhir dengan

mencetak “BERHASIL: true” jika semua input pada kelima percobaan benar, atau “BERHASIL: false” jika setidaknya ada satu percobaan yang salah.

DAFTAR PUSTAKA

Prasti Eko Yunanto, S.T., M.Kom. (2024). MODUL PRAKTIKUM 14 – KOMPOSISI ALGORITMA DAN PEMROGRAMAN 1 S1 INFORMATIKA (MODUL 14)