

**LAPORAN PRAKTIKUM**  
**MODUL: 5**  
**ALGORITMA DAN PEMROGRAMAN**



**DISUSUN OLEH:**

**NAMA: JESIKA METANIA RAHMA ARIFIN**

**NIM: 103112400080**

**KELAS: 12 IF 01**

**DOSEN:**

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

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024/2025**

## DASAR TEORI

### Algoritma Pengulangan dengan For Loop

Ada 2 jenis algoritma pengulangan yang umum ditemui dalam pemrograman, yakni algoritma pengulangan *counted loop* dan *uncounted loop*. Algoritma pengulangan *counted loop* merupakan algoritma pengulangan yang bisa kita gunakan apabila kita sudah mengetahui berapa kali pengulangan yang dibutuhkan. Sementara itu, algoritma pengulangan *uncounted loop* merupakan algoritma pengulangan yang bisa kita gunakan ketika kita belum mengetahui berapa jumlah pengulangan yang dibutuhkan. *Uncounted loop* ini mungkin terdengar sedikit aneh, tetapi kita mungkin membutuhkannya pada kondisi-kondisi tertentu, seperti ketika pengulangan yang dilakukan terus bergantung pada *input* yang kita masukkan.

### Struktur Pengulangan *For Loop*

Struktur pengulangan *For* merupakan struktur pengulangan yang paling umum dijumpai untuk melakukan *counted loop*. Berikut adalah struktur pengulangan *For* secara umum:

```
for (int i = 0; i < 5; i++)  
{  
    perintah  
}
```

Parameter yang digunakan dalam pengulangan *For* bisa kita bagi menjadi 3 bagian, yakni bagian deklarasi 'int i = 0', 'i < 5', serta 'i++'. Berikut adalah penjelasan untuk masing-masing bagian tersebut:

#### *Parameter Pertama*

```
for (int i = 0; i < 5; i++)  
{  
    perintah  
}
```

Bagian ini merupakan bagian deklarasi variabel *counter*. Variabel *counter* ini diperlukan agar algoritma *For* bisa menelusuri berapa kali pengulangan yang dibutuhkan, berapa kali pengulangan yang sudah dilakukan, dan berapa kali pengulangan yang masih harus dilakukan.

Pada bagian deklarasi variabel *counter* ini, kamu dapat menentukan nama variabel *counter*-mu serta dari angka berapa pengulanganmu dimulai. Apabila kita melihat contoh di atas, variabel *counter*-nya dinamakan dengan 'i'. Variabel i ini ditetapkan memiliki nilai 0, sehingga pengulangannya akan dimulai dari angka 0 juga.

#### Parameter Kedua

```
for (int i = 0; i < 5; i++)  
{  
    perintah  
}
```

Setelah menentukan variabel *counter* yang akan kamu gunakan serta menentukan nilai awalnya, kamu tentunya harus menentukan berapa kali pengulangan yang kamu butuhkan. Bagian ini merupakan bagian yang sangat penting, karena *For* sendiri merupakan struktur pengulangan untuk *counted loop*. Jika kamu kesulitan untuk mengisi bagian ini, mungkin struktur pengulangan *uncounted loop* merupakan struktur yang lebih tepat untuk program pengulangan milikmu.

Melihat dari contoh di atas, tertulis bahwa 'i < 5'. Hal ini menyatakan bahwa kita ingin pengulangannya terus terjadi selama i bernilai di bawah 5. Apakah hal ini sudah pasti bahwa i akan melakukan pengulangan sebanyak 5 kali? Tentu tidak ya, teman-teman. Selain kedua parameter ini, kita masih harus menentukan parameter terakhir yang dibutuhkan dalam pengulangan *For*.

#### Parameter Ketiga

```
for (int i = 0; i < 5; i++)  
{  
    perintah  
}
```

### ***GUIDED 1:***

Buatlah program untuk menampilkan baris bilangan dari  $a$  sampai dengan  $b$ .

Masukan terdiri dari dua bilangan bulat  $a$  dan  $b$ , yang mana  $a \leq b$ .

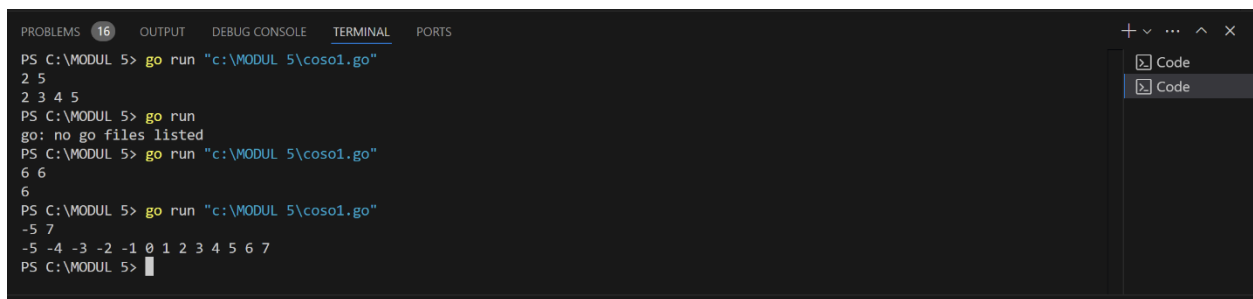
Keluaran berupa baris bilangan dari  $a$  sampai dengan  $b$ .

```
package main

import "fmt"

func main() {
    var a, b int
    var j int
    fmt.Scan(&a, &b)
    for j = a; j <=b; j+=1 {
        fmt.Print(j, " ")
    }
}
```

### ***OUTPUT:***



```
PROBLEMS 16 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\MODUL 5> go run "c:\MODUL 5\coso1.go"
2 5
2 3 4 5
PS C:\MODUL 5> go run
go: no go files listed
PS C:\MODUL 5> go run "c:\MODUL 5\coso1.go"
6 6
6
PS C:\MODUL 5> go run "c:\MODUL 5\coso1.go"
-5 7
-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7
PS C:\MODUL 5> |
```

### ***DESKRIPSI PROGRAM:***

Program di atas merupakan program yang meminta menampilkan baris bilangan dari  $a$  sampai dengan  $b$ . Yang dimana terdiri dari dua bilangan bulat  $a$  lebih dari sama dengan  $b$ .

### ***GUIDED 2:***

Buatlah program yang digunakan untuk menampilkan sejumlah  $n$  luas segitiga, apabila

diketahui sisi alas dan tinggi dari masing-masing segitiga.

Masukan terdiri dari  $n + 1$  baris. Baris pertama merupakan suatu bilangan bulat  $n$ , selanjutnya  $n$  baris berikutnya masing-masing merupakan panjang sisi alas dan tinggi dari segitiga

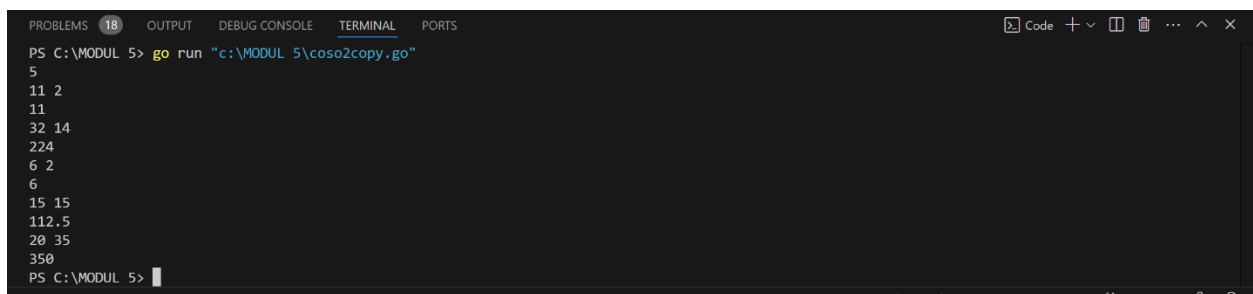
Keluaran terdiri dari  $n$  baris, yang masing-masing menyatakan luas dari segitiga.

```
package main

import "fmt"

func main() {
    var j, alas, tinggi, n int
    var luas float64
    fmt.Scan(&n)
    for j = 1; j <=n; j+=1 {
        fmt.Scan(&alas, &tinggi)
        luas = 0.5 * float64(alas * tinggi)
        fmt.Println(luas)
    }
}
```

### ***OUTPUT:***



```
PS C:\MODUL 5> go run "c:\MODUL 5\coso2copy.go"
5
11 2
11
32 14
224
6 2
6
15 15
112.5
20 35
350
PS C:\MODUL 5>
```

### ***DESKRIPSI PROGRAM:***

*Program diatas merupakan program yang digunakan untuk menampilkan luas segitiga dengan diketahui sisi alas dan tinggi.*

### ***GUIDED 3:***

Buatlah program yang digunakan untuk menghitung hasil perkalian dua buah bilangan tanpa menggunakan operator kali "\*".

Masukan terdiri dari dua bilangan bulat positif.

Keluaran terdiri dari sebuah bilangan yang menyatakan hasil perkalian dari dua bilangan pada masukan.

```
package main

import "fmt"

func main(){
    var j, v1, v2 int
    var hasil int
    fmt.Scan(&v1, &v2)
    hasil = 0
    for j = 1; j <= v2; j+=1 {
        hasil = hasil + v1
    }
    fmt.Println(hasil)
}
```

### ***OUTPUT:***



```
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\MODUL 5> go run "c:\MODUL 5\coso3.go"
2 100
200
PS C:\MODUL 5> go run "c:\MODUL 5\coso3.go"
444 156
69264
PS C:\MODUL 5> █
```

### ***DESKRIPSI PROGRAM:***

*Program di atas merupakan program yang digunakan untuk menghitung hasil perkalian dua buah bilangan tanpa menggunakan operator kali (\*).*

### ***UNGUIDED: 1***

*Buatlah program untuk menjumlahkan sekumpulan bilangan.*

*Masukan terdiri dari suatu bilangan bulat positif  $n$ .*

*Keluaran berupa bilangan hasil penjumlahan dari 1 sampai dengan  $n$ .*

```
package main

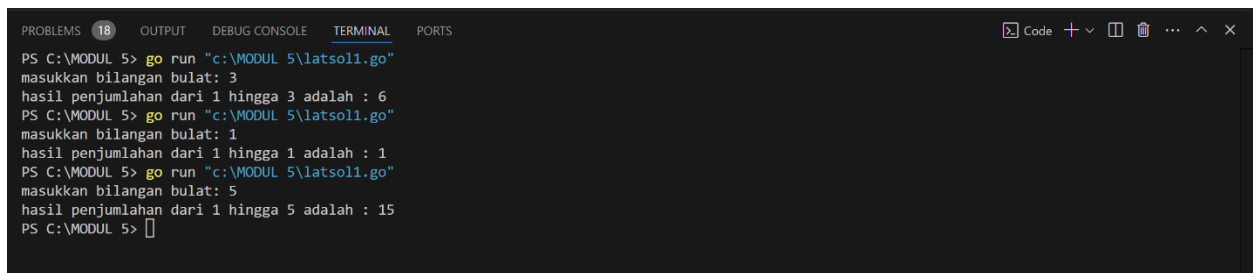
import "fmt"
```

```

func main() {
    var n, sum int
    fmt.Println("masukkan bilangan bulat: ")
    fmt.Scan(&n)
    for i := 1; i <= n; i++ {
        sum += i
    }
    fmt.Printf("hasil penjumlahan dari 1 hingga %d adalah : %d\n", n, sum)
}

```

### **OUTPUT:**



```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\MODUL 5> go run "c:\MODUL 5\latsoli.go"
masukkan bilangan bulat: 3
hasil penjumlahan dari 1 hingga 3 adalah : 6
PS C:\MODUL 5> go run "c:\MODUL 5\latsoli.go"
masukkan bilangan bulat: 1
hasil penjumlahan dari 1 hingga 1 adalah : 1
PS C:\MODUL 5> go run "c:\MODUL 5\latsoli.go"
masukkan bilangan bulat: 5
hasil penjumlahan dari 1 hingga 5 adalah : 15
PS C:\MODUL 5> 

```

### **DESKRIPSI PROGRAM:**

*Program di atas merupakan program untuk menjumlahkan sekumpulan bilangan yang terdiri dari suatu bilangan bulat positif dan menghasilkan bilangan hasil penjumlahan dari 1 sampai dengan n.*

### **UNGUIDED : 2**

Buatlah program yang digunakan untuk menghitung volume sejumlah n kerucut, apabila diketahui panjang jari-jari alas kerucut dan tinggi dari kerucut.

Masukan terdiri dari beberapa baris. Baris pertama adalah bilangan bulat n, selanjutnya n baris berikutnya masing-masing merupakan panjang jari-jari alas kerucut dan tinggi dari kerucut.

Keluaran terdiri dari beberapa baris, yang masing-masingnya menyatakan volume dari n kerucut.

```

package main

```

```
import "fmt"

func main() {
    var panjang_jejari, tinggi float64
    var n int
    fmt.Scan(&n)
    for i := 1; i <= n ; i++ {
        fmt.Print("masukkan panjang, jejari alas, tinggi : ")
        fmt.Scan(&panjang_jejari, &tinggi)
        volume:= 1.0/3.0 * 3.14 * panjang_jejari*panjang_jejari* tinggi
        fmt.Println(volume)
    }
}
```

### **OUTPUT:**

```
PS C:\MODUL 5> go run "c:\MODUL 5\latsol2.go"
1
masukkan panjang, jejari alas, tinggi : 3 4
37.679999999999999
PS C:\MODUL 5> go run "c:\MODUL 5\latsol2.go"
3
masukkan panjang, jejari alas, tinggi : 1 1
1.0466666666666666
masukkan panjang, jejari alas, tinggi : 2 2
8.373333333333333
masukkan panjang, jejari alas, tinggi : 3 3
28.259999999999994
PS C:\MODUL 5> 
```

### **DESKRIPSI PROGRAM:**

*Program di atas merupakan program yang digunakan untuk menghitung volume sejumlah kerucut apabila di ketahui panjang jari jari dan tinggi kerucut.*

### **UNGUIDED: 3**

Buatlah program yang digunakan untuk menghitung hasil pemangkatan dari dua buah bilangan. Program dibuat dengan menggunakan operator perkalian dan struktur kontrol perulangan.

Masukan terdiri dari dua bilangan bulat positif.

Keluaran terdiri dari suatu bilangan yang menyatakan hasil bilangan pertama dipangkatkan dengan bilangan kedua.

```
package main
```



```

import "fmt"
import "math"

func main() {
    var n, t float64
    fmt.Scan(&n, &t)
    pangkat:= math.Pow(n, t)
    fmt.Print(pangkat)
}

```

### **OUTPUT:**

```

PROBLEMS 14 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\MODUL 5> go run "c:\MODUL 5\latsol3.go"
4 2
16
PS C:\MODUL 5> go run "c:\MODUL 5\latsol3.go"
2 10
1024
PS C:\MODUL 5> go run "c:\MODUL 5\latsol3.go"
10 3
1000
PS C:\MODUL 5>

```

### **DESKRIPSI PROGRAM:**

*Program di atas merupakan adalah program yang digunakan untuk menghitung hasil pemangkatan dari dua buah bilangan. Masukkan terdiri dari bilangan bulat positif dan menghasilkan keluaran bilangan yang menyatakan hasil bilangan pertama di pangkatkan dengan bilangan kedua.*

### **UNGUIDED: 4**

Buatlah program yang digunakan untuk menghitung hasil faktorial dari suatu bilangan.

Masukan terdiri dari suatu bilangan bulat non negatif.

Keluaran terdiri dari hasil faktorial dari bilangan bulat n.

```

package main

import "fmt"

func factorial(n int) int {
    if n == 0 {
        return 1
    }
    return n * factorial(n-1)
}

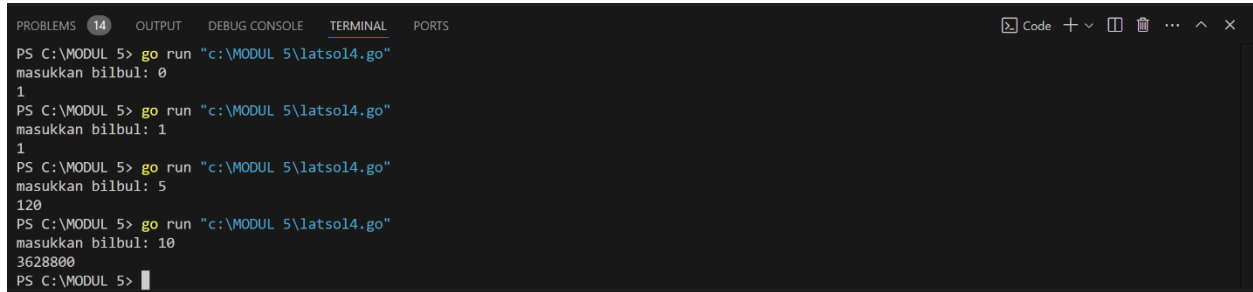
func main() {
    var n int

```

```
fmt.Print("masukkan bilbul: ")
fmt.Scan(&n)
hasil:= factorial(n)
fmt.Println(hasil)

}
```

### ***OUTPUT:***



```
PS C:\MODUL 5> go run "c:\MODUL 5\latso14.go"
masukkan bilbul: 0
1
PS C:\MODUL 5> go run "c:\MODUL 5\latso14.go"
masukkan bilbul: 1
1
PS C:\MODUL 5> go run "c:\MODUL 5\latso14.go"
masukkan bilbul: 5
120
PS C:\MODUL 5> go run "c:\MODUL 5\latso14.go"
masukkan bilbul: 10
3628800
PS C:\MODUL 5> |
```

### ***DESKRIPSI PROGRAM:***

*Program di atas merupakan program yang di buat untuk menghitung hasil factorial dari suatu bilangan yang terdiri dari suatu bilangan bulat non negative dan keluaran terdiri dari bilangan bulat n.*

### ***DAFTAR PUSTAKA***

<https://codingstudio.id/blog/algoritma-pengulangan-for-loop/>