

VERSION 1.0 JULY, 2022



PEMROGRAMAN LANJUT

MODUL 1 - PROGRAM CORRECTNESS I

TIM PENYUSUN :

- WILDAN SUHARSO, S.KOM., M.KOM
- HANIA PRATIWI NINGRUM
- FILDZAH LATHIFAH

PRESENTED BY : LAB. TEKNIK INFORMATIKA UNIVERSITAS MUHAMMADIYAH MALANG

PEMROGRAMAN LANJUT

CAPAIAN PEMBELAJARAN MATA KULIAH

1. Mahasiswa mampu memahami tipe-tipe error dan mampu memperbaiki error pada program.
2. Mahasiswa mampu memahami concept of spesification.
3. Mahasiswa mampu memahami excepcetion handling dan mengimplementasikannya pada program.

KEBUTUHAN HARDWARE & SOFTWARE

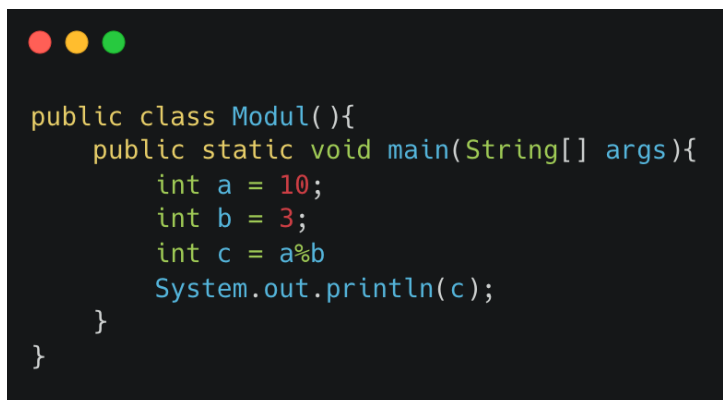
- Laptop/PC
- Netbeans/IntelliJ/Eclipse

MATERI POKOK

1. Types Of Errors

a. Syntax Error

Error tata bahasa (sintaks) merupakan jenis error yang paling banyak terjadi dalam pembuatan program. Namun error ini paling mudah terdeteksi karena umumnya compiler atau interpreter dari masing-masing bahasa program akan melakukan pengecekan sebelum program dijalankan (saat dikompilasi). Lokasi baris yang menyebabkan error juga biasanya sudah ditunjukkan. Hanya perlu kejelian untuk memperbaikinya. Contoh dalam Java :



```
public class Modul(){  
    public static void main(String[] args){  
        int a = 10;  
        int b = 3;  
        int c = a%b  
        System.out.println(c);  
    }  
}
```

b. Logic Error

Jenis error yang satu ini merupakan jenis error yang paling susah dideteksi karena terjadinya bukan karena kesalahan penulisan (sintaks) atau kesalahan proses runtime, namun kesalahan dari sisi programmer, dalam hal ini algoritma

yang digunakan. Karena logikanya salah, tentunya output yang dihasilkan juga akan salah. Untuk mendeteksi letak kesalahannya, bukanlah hal yang mudah. Terkadang kita harus merunut algoritma yang digunakan baris per baris (line byline). Contoh dalam Java :

```
import java.util.Scanner;

public class Modul(){
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);
        int jari;
        float luas;

        system.out.println("Masukkan jari-jari lingkaran : ");
        jari = input.nextInt();

        luas = 2 * 3.14 * jari * jari;
        system.out.println("Luas lingkaran adalah : " + luas);
    }
}
```

c. Runtime Error

Tingkatan error selanjutnya adalah runtime error. Dimana error ini akan terdeteksi saat program dijalankan (*di-running*). Penyebabnya beragam, pada umumnya karena terjadi kesalahan dalam proses input, perhitungan dan juga dalam proses output. Sebagai contoh yang banyak terjadi adalah error runtime karena pembagian suatu bilangan dengan nol. Lihat contoh program berikut ini! Secara sintaks tentu tidak terdapat error, namun jika dijalankan, operasi pembagian pada baris ke-5 akan menyebabkan error “division by zero”. Contoh dalam Java :

```
public class Modul(){
    public static void main(String[] args){
        int a = 10; b = 0;

        system.out.println(a);
        system.out.println(b);
        System.out.println(a/b);
    }
}
```

2. The Concept Of a Specification

Pada sebuah program pasti terdapat spesifikasi-spesifikasi khusus yang harus ditaati untuk menggunakan sebuah program. Seperti misalnya pada sebuah program yang menggunakan fitur login. Disana user diminta untuk memasukkan password dengan kriteria tertentu yang telah ditentukan. Seperti password harus terdiri dari minimal 8 karakter, harus terdapat kombinasi angka dan huruf dan harus terdiri dari kombinasi huruf besar dan kecil. Spesifikasi password tersebut harus dimiliki sebuah program guna mencegah terjadi error saat menggunakan program.

3. Defensive Programming (Exception Handling)

Pada bahasa Java, terdapat suatu metode yang dinamakan Exception Handling. Exception handling ini digunakan untuk menangani error atau keadaan tertentu yang terjadi ketika program sedang berjalan. Ketika terjadi error saat program berjalan (runtime error) program akan mengarahkan alur eksekusi program ke fungsi khusus yang dinamakan handler. Namun jika tidak terjadi error yang menyebabkan exception maka semua handler akan diabaikan dan program dijalankan secara normal. Contoh :

```
import java.util.Scanner;

public class Modul(){
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);
        int a;

        try{
            System.out.println("Masukkan angka : ");
            a = input.nextInt();
            if(a <= 0){
                throw new Exception("Angka tidak boleh <= 0");
            }else{
                System.out.println("Angka yang anda masukkan adalah : " + a);
            }
        }catch(int x){
            System.out.println("program menangkap eksepsi yang bernilai " + x);
        }
    }
}
```

REFERENSI

<http://achmatim.net/2008/11/17/mengenal-error-dalam-program/>

<https://www.javatpoint.com/exception-handling-in-java>

<https://docs.oracle.com/javase/tutorial/essential/exceptions/index.html>

LEMBAR KERJA

TUGAS 1

Perbaiki program java dibawah ini dengan menambahkan exception, amati error yang terjadi. Kemudian jalankan program.

```
public class modul1 {
    public static void main(String[] args) {
        int[] arr = new int[5];
        arr[5] = 25;
    }
}
```

TUGAS 2

Buatlah program inputan nim dengan specification sebagai berikut :

1. Hanya nim mahasiswa informatika dengan kode "10370311".
2. Tidak boleh kurang atau lebih dari 15 digit.
3. Jika nim salah tampilkan pesan error, kemudian meminta inputan kembali.
4. Buatlah exception yang berbeda untuk setiap error. Contoh jika nim-nya kurang dari 15 digit dan nim lebih lebih dari 15 digit.

Simpan kode tugas 2 yang telah anda kerjakan karena akan digunakan di Modul selanjutnya.

Kerjakan dengan rapi ya, agar minim terjadi kesalahan syntax dan mudah dibaca.

RUBRIK PENILAIAN

Kriteria	Nilai
Tugas 1	20
Tugas 2	35
Pemahaman	45
Total	100

Silahkan dikerjakan tanpa copas dan jangan lupa berdoa sebelum praktikum! Good Luck!!