



DASAR PEMROGRAMAN KOMPUTER

Minggu 1 - Praktikum

- o Nama : Imam Fahrur Rozi
- o Email : imam.rozi@polinema.ac.id
- o Ruang : Lt. 2 Gd. AF
- o No HP : 085233139738
- o Rumah : Gajayana Inside B9, belakang UIN

Target

Setelah menempuh materi ini, mahasiswa hendaknya mampu:

- o Memahami tentang konsep dasar pemrograman komputer
- o Mengenal tentang struktur program dan langkah mengembangkan program Java

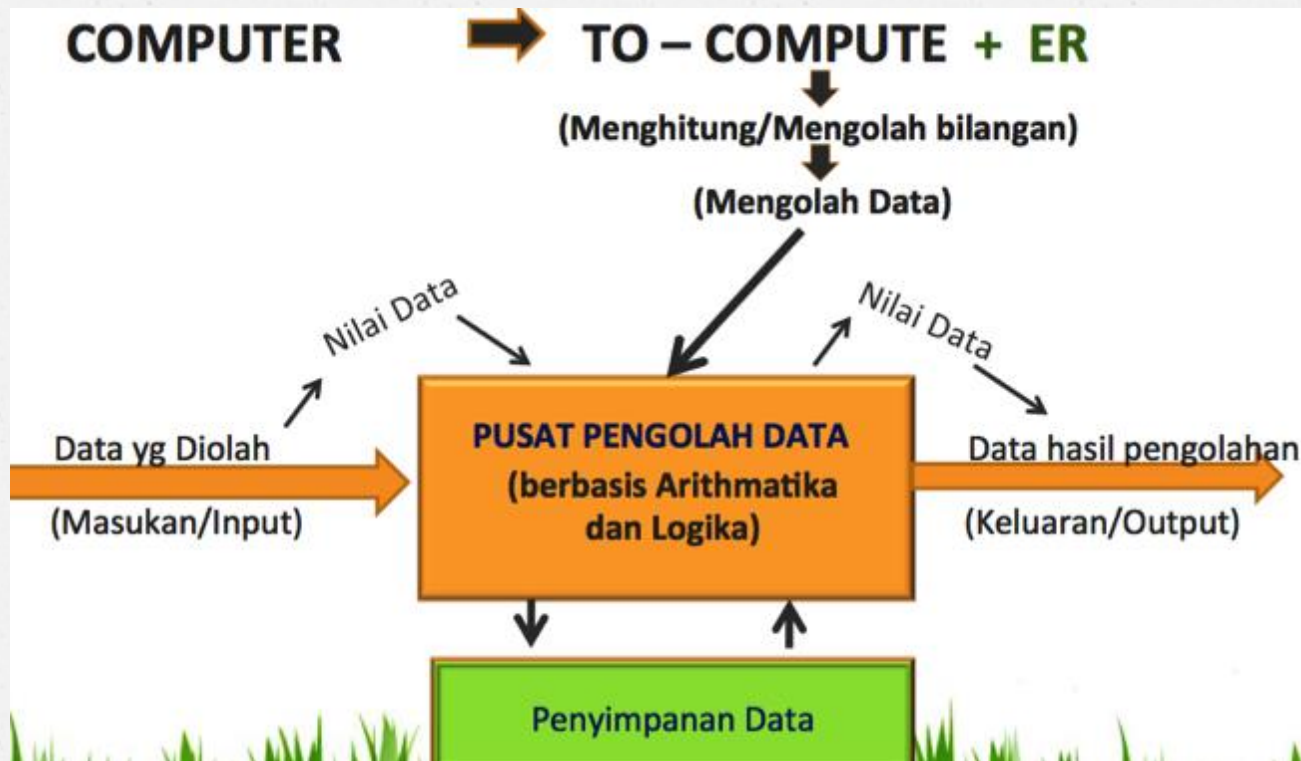
Pokok Bahasan

- o Konsep Dasar Pemrograman
- o Program Bahasa Pemrograman
- o Langkah Membuat Program
- o Translator
- o Tahap Pengembangan Program Java
- o Struktur Dasar Program Java
- o Sintaks Output



Dasar Pemrograman Komputer

Prinsip Kerja Komputer



Program

- **Program** merupakan Sederetan instruksi yang dikerjakan oleh komputer, untuk memproses data hingga menghasilkan output yang diinginkan
- Untuk membuat program, sebelumnya perlu dirancang langkah-langkah penyelesaian masalah → algoritma
- Program ditulis dengan menggunakan **bahasa pemrograman** tertentu

Program

- o Program = Algoritma + Bahasa Pemrograman
- o Belajar membuat **Program**, berarti belajar untuk membuat **algoritma** dan mentranslasikan algoritma tersebut ke **bahasa pemrograman**, hingga program bisa jalan dan didapatkan hasil yang diinginkan

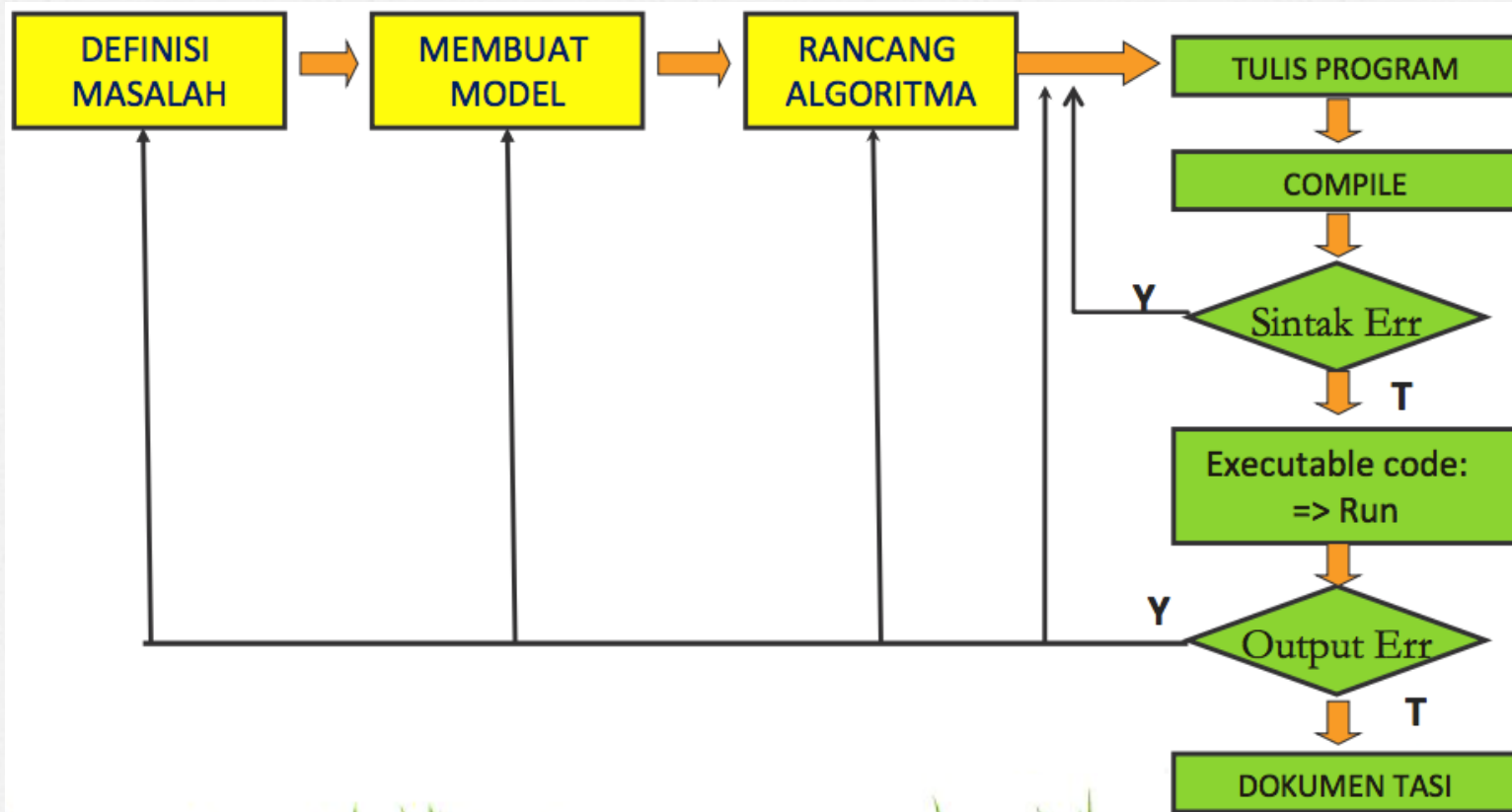
Bahasa Pemrograman

- Adalah Bahasa khusus yang digunakan untuk membuat suatu program komputer
- Tingkat tinggi
 - Lebih dekat dengan bahasa manusia
 - Lebih mudah dipahami
 - Contoh: Java, C, C++, Pascal, dsb
- Tingkat rendah
 - Lebih dekat dengan bahasa mesin
 - Lebih sulit dipahami
 - Contoh: bahasa assembly

Langkah dalam Membuat Program

- o Pamahi masalah/definisi masalah
- o Buat model
- o Buat algoritma
- o Tulis program
- o Compile
- o Running
- o Dokumentasi

Langkah Membuat Program



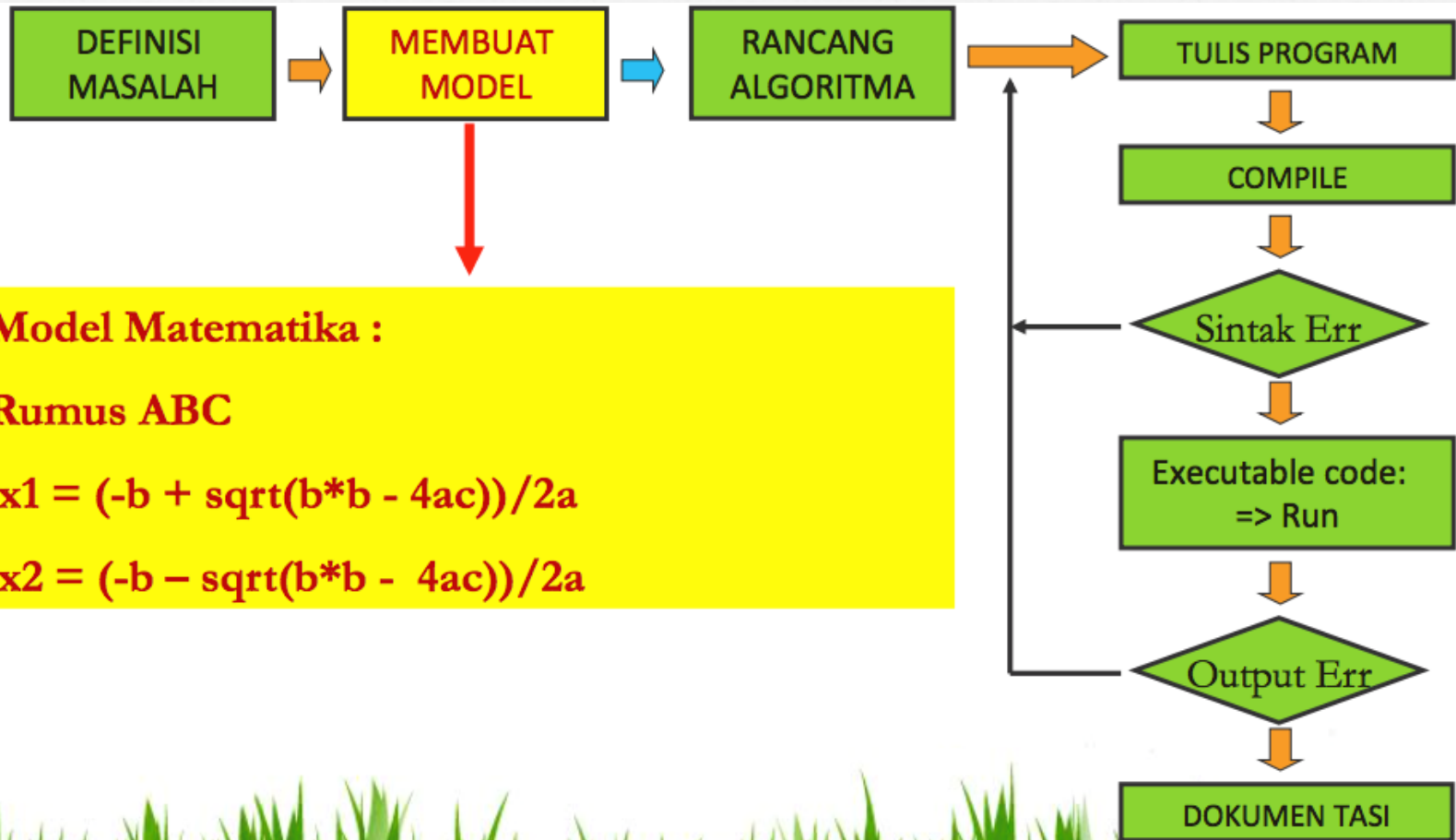
Langkah Membuat Program

(1. Definisi Masalah)



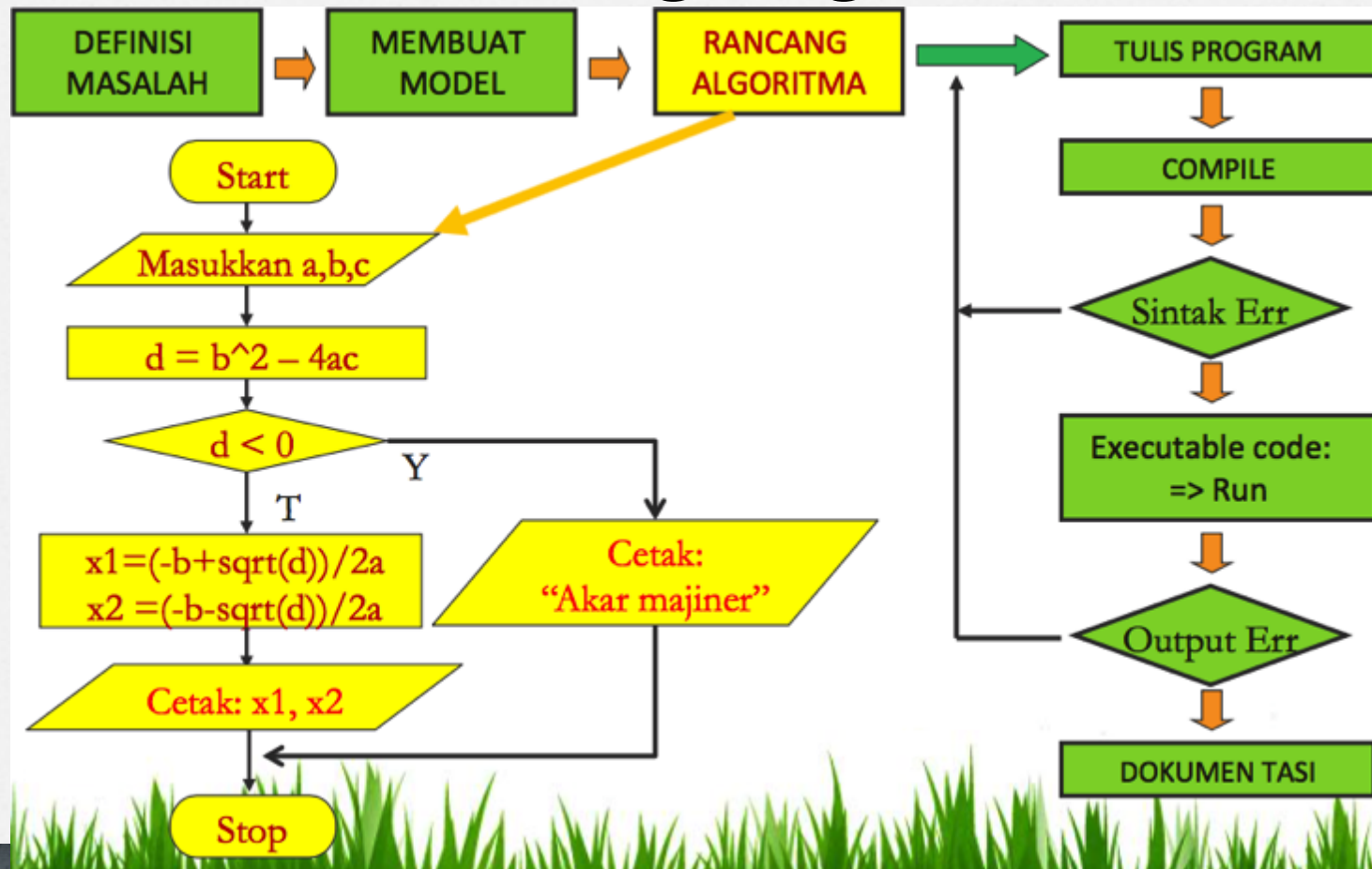
Langkah Membuat Program

(2. Buat Model)

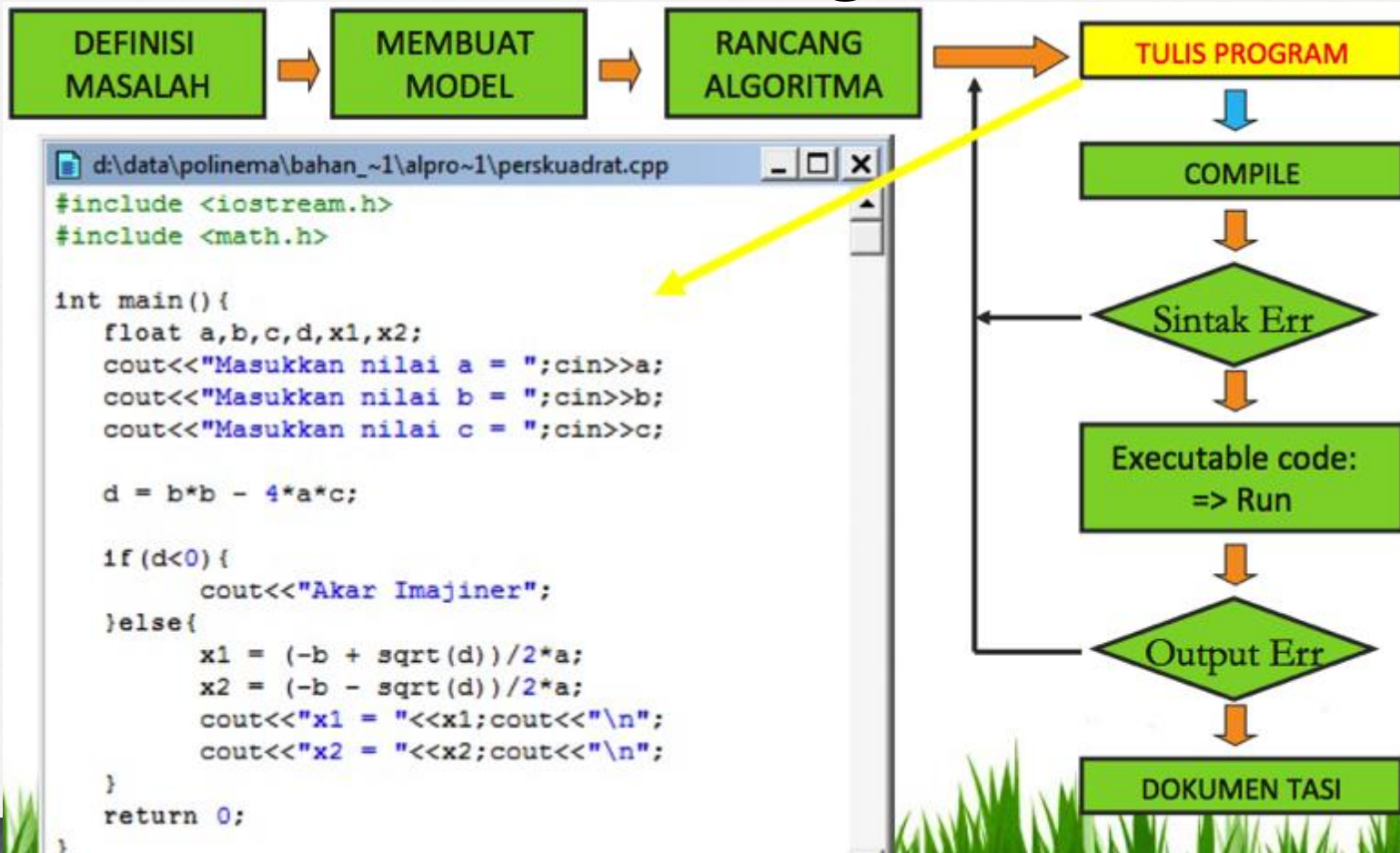


Langkah Membuat Program

(3. Rancang Algoritma)



Langkah Membuat Program (4. Tulis Program)



Translator

- o Dari bahasa pemrograman tingkat tinggi, agar bisa dijalankan oleh mesin perlu diterjemahkan
- o Translator → digunakan untuk menterjemahkan bahasa pemrograman hingga bisa dikenali dan dieksekusi oleh komputer
- o Ada 2 macam:
 - o Interpreter
 - o Compiler

Translator

- Interpreter:

- Input:

- Source Code / Kode Program

- Output:

- Interpreter tidak menghasilkan file baru/file objek

- Source code langsung diterjemahkan oleh interpreter dan langsung dijalankan, tanpa diubah dulu ke file objek

- Contoh:

- PHP

Translator

- o Compiler:
 - o Input:
 - o Source Code / Kode Program
 - o Output:
 - o Compiler menghasilkan file baru/file java bytecode
 - o Source code akan dianalisis oleh compiler dan diterjemahkan menjadi object code dan ditulus ke dalam file objek/file baru (COMPILE TIME)
 - o Object code ini yang selanjutnya akan dijalankan (RUN TIME)
- o Contoh:
 - o Java, C, C++, dsb

Pengenalan Java

- Java dikembangkan oleh Sun Microsystem tahun 1995
- Dibeli oleh Oracle pada tahun 2009
- Merupakan bahasa pemrograman berorientasi objek. Meski demikian di matakuliah ini kita tidak fokus pada konsep berorientasi objek, melainkan pada konsep dasar pemrograman-nya

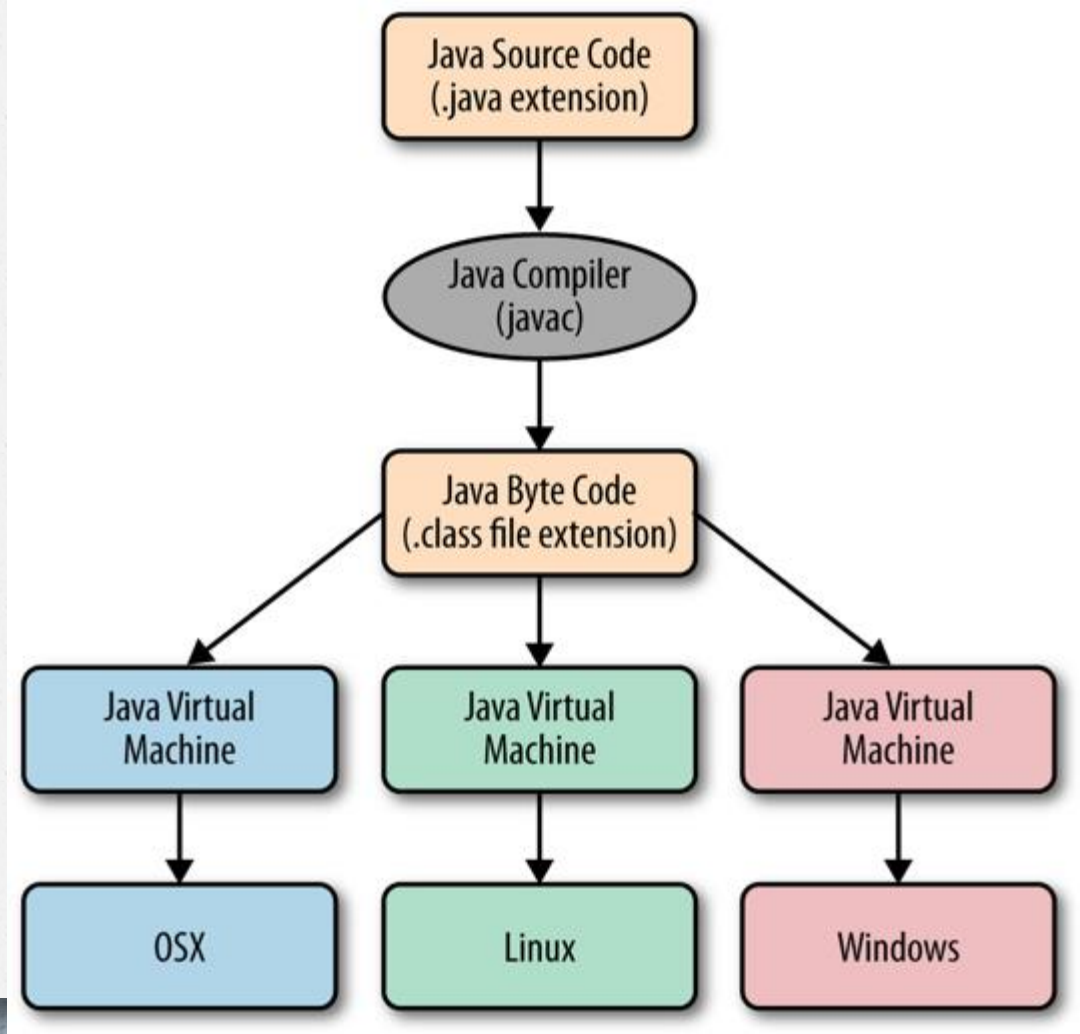
Beberapa Istilah Terkait dengan Java

- JDK (Java Development Kit) : kumpulan tools untuk membuat/mengembangkan program Java. Di dalamnya terdapat java compiler
- JRE (Java Runtime Environment) : lingkungan untuk menjalankan program Java (di dalamnya terdapat JVM)
- JVM (Java Virtual Machine) : bagian dari JRE yang akan menerjemahkan java bytecode hingga menjadi bahasa mesin sesuai dengan platform dimana ia akan dijalankan

Tahap Pengembangan Program Java

COMPILE
(perintah `javac`)

RUN
(perintah `java`)



Struktur Dasar Program Java

```
class>NamaClass {  
    public static void main(String[] args){  
        //statement;  
        //statement;  
        //statement;  
    }  
}
```

Struktur Dasar Program Java

1. Class : Bagian utama dari program di Java. Semua bagian program di java harus berada di dalam class.
2. Fungsi main() : fungsi atau bagian program yang pertama kali akan dijalankan. Program Java yang tidak memiliki fungsi main() maka ia tidak bisa dijalankan langsung.
3. Statement : adalah baris instruksi tertentu. Dan setiap statement diakhiri dengan tanda ;

Sintaks Penulisan Class

- o Digunakan sintaks **class** dan diikuti dengan nama class yang akan dibuat
- o Nama class berupa kata benda
- o Nama class harus diawali dengan HURUF BESAR.
- o Jika terdiri dari 2 kata, keduanya tidak dipisah, dan masing-masing diawali dengan huruf besar

o Contoh:

```
class Kalkulator{  
}  
class AlatHitung{  
}
```

Sintaks Penulisan Fungsi main()

o Format penulisan fungsi main sebagai berikut:

```
public static void main(String[] args){  
  
}
```

Catatan: Untuk lebih detil tentang sintaks public, static akan dijelaskan lebih lanjut di matakuliah Pemrograman Berorientasi Objek

Sintaks Output

o Digunakan sintaks `System.out.print()` atau `System.out.println()`

o Contoh:

```
System.out.println("Hello.....");
```

```
System.out.print("Hello tanpa ganti baris....");
```

```
System.out.print("Nama \t: Imam \n");
```

1. Catatan:
2. `\t` untuk menampilkan tab, `\n` untuk menampilkan ganti baris
3. Perhatikan penggunaan tanda "..."

Contoh Program

```
class Biodata{  
    public static void main(String[] args){  
        System.out.print("NAMA \t: Imam F \n");  
        System.out.println("NIP \t: 198406102008121004");  
    }  
}
```

class

Fungsi main()

Statement

Output program:

```
NAMA      : Imam F  
NIP       : 198406102008121004
```