COMPUTATIONAL PHYSICS WITH JAVA

SEGUN EMMANUEL (ABODE)

FRIDAY 6TH, OCTOBER 2016

OUTLINE

- **OVERVIEW OF COMPUTATIONAL PHYSICS**
- TOOLS OF COMPUTATIONAL PHYSICS
- INTRODUCTION TO JAVA
- COMPONENTS OF JAVA
- HANDS-ON

WHAT IS COMPUTATIONAL PHYSICS?

- Computational Physics may be broadly defined as 'the science of using computers to assist in the solution of physical problems, and to further physics research'.

- Computational physics is a tool for solving complex numerical problems in physics

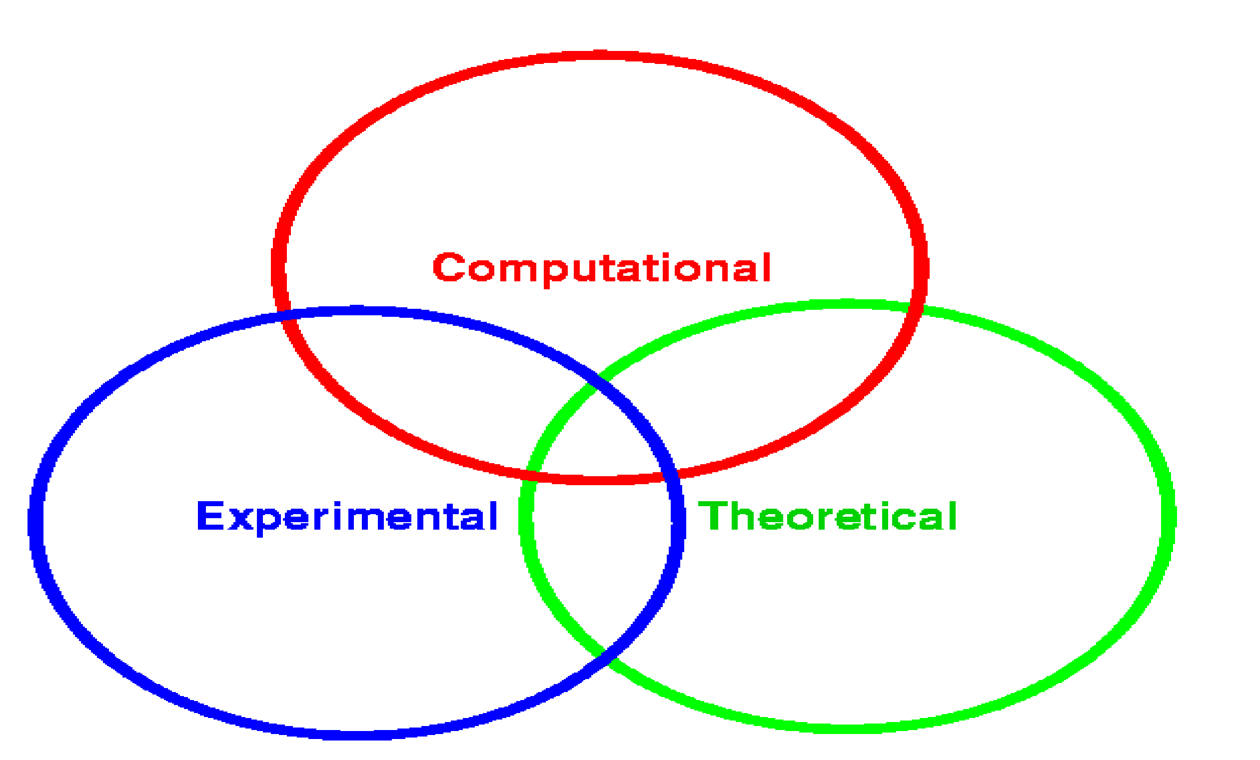
WHAT IS COMPUTATIONAL PHYSICS?

- Both experimental and theoretical physics are incomplete without the option to compute whenever it is neccessary.

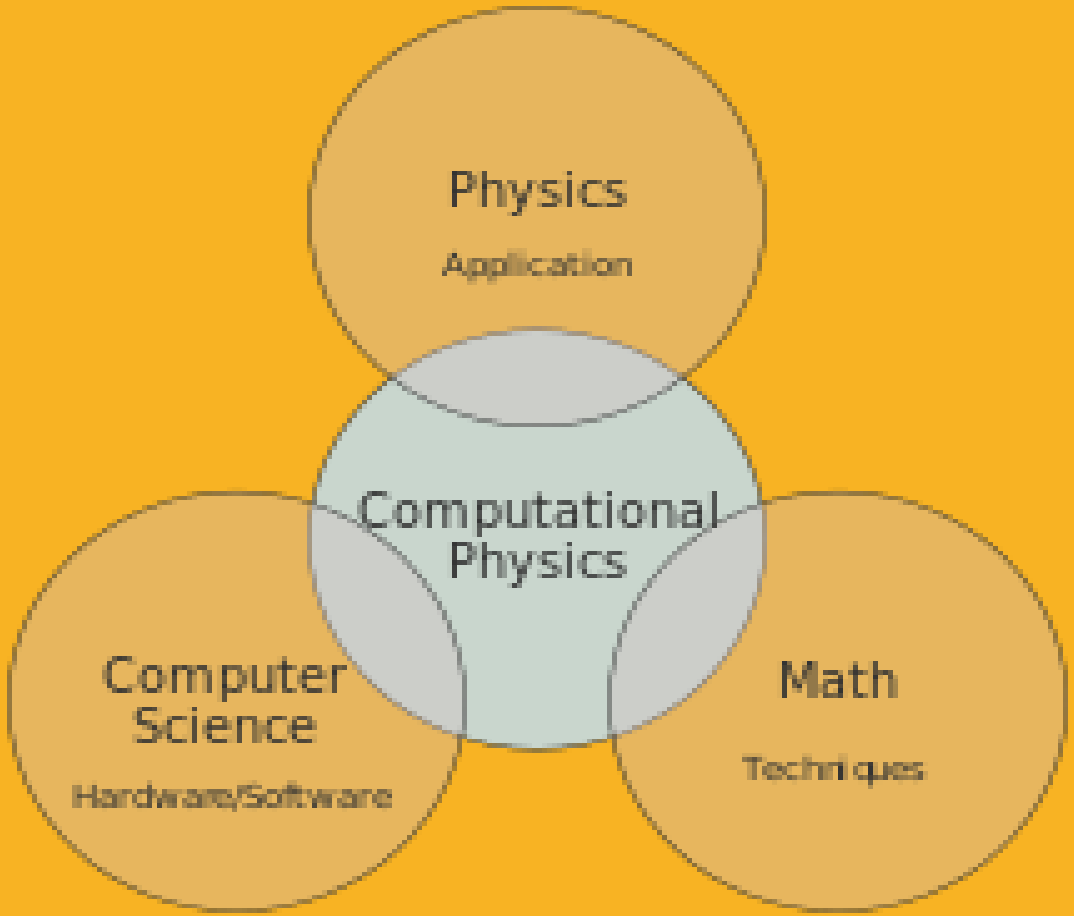
- The goal of computational physics is not to replace theory or experiment, but to enhance our understanding of physical processes

WHAT IS COMPUTATIONAL PHYSICS?

- Computational physics bridges both theory and experiment physics



MULTIDISCIPLINARY NATURE OF COMPUTATIONAL PHYSICS



APPLICATION IN PHYSICS

- Large scale quantum mechanical calculations in nuclear, atomic, molecular and condensed matter physics
- Large scale calculations in such fields as hydrodynamics, astrophysics, plasma physics, meteorology and geophysics
- Simulation and modelling of complex physical systems such as those that occur in condensed matter physics, medical physics and industrial applications
- Computer algebra; development and applications etc

WHY DO WE NEED COMPUTATIONAL PHYSICS?

- We need computational physics when:

  we cannot solve problems analytically

  we have too much data to process

- Many, if not the most, problems in contemporary physics could never be solved without computers

OUTLINE

- OVERVIEW OF COMPUTATIONAL PHYSICS
- TOOLS OF COMPUTATIONAL PHYSICS
- INTRODUCTION TO JAVA
- COMPONENTS OF JAVA
- HANDS-ON

TOOLS OF COMPUTATIONAL PHYSICS

- Mathematical software packages such as MATHEMATICA, MAPLE or MATLAB etc
- Programming Languages such as JAVA, C#, PYTHON, FORTRAN, C, C++ etc

ADVANTAGES OF MATHEMATICAL SOFTWARE PACKAGES

- They facilitate the very rapid coding up of numerical problems

- Good for small and medium projects

DISADVANTAGES OF MATHEMATICAL SOFTWARE PACKAGES

- They produce executable code which is interpreted, rather than compiled.
- They are not suitable for full-blown research projects, since the code which they produce generally runs far too slowly

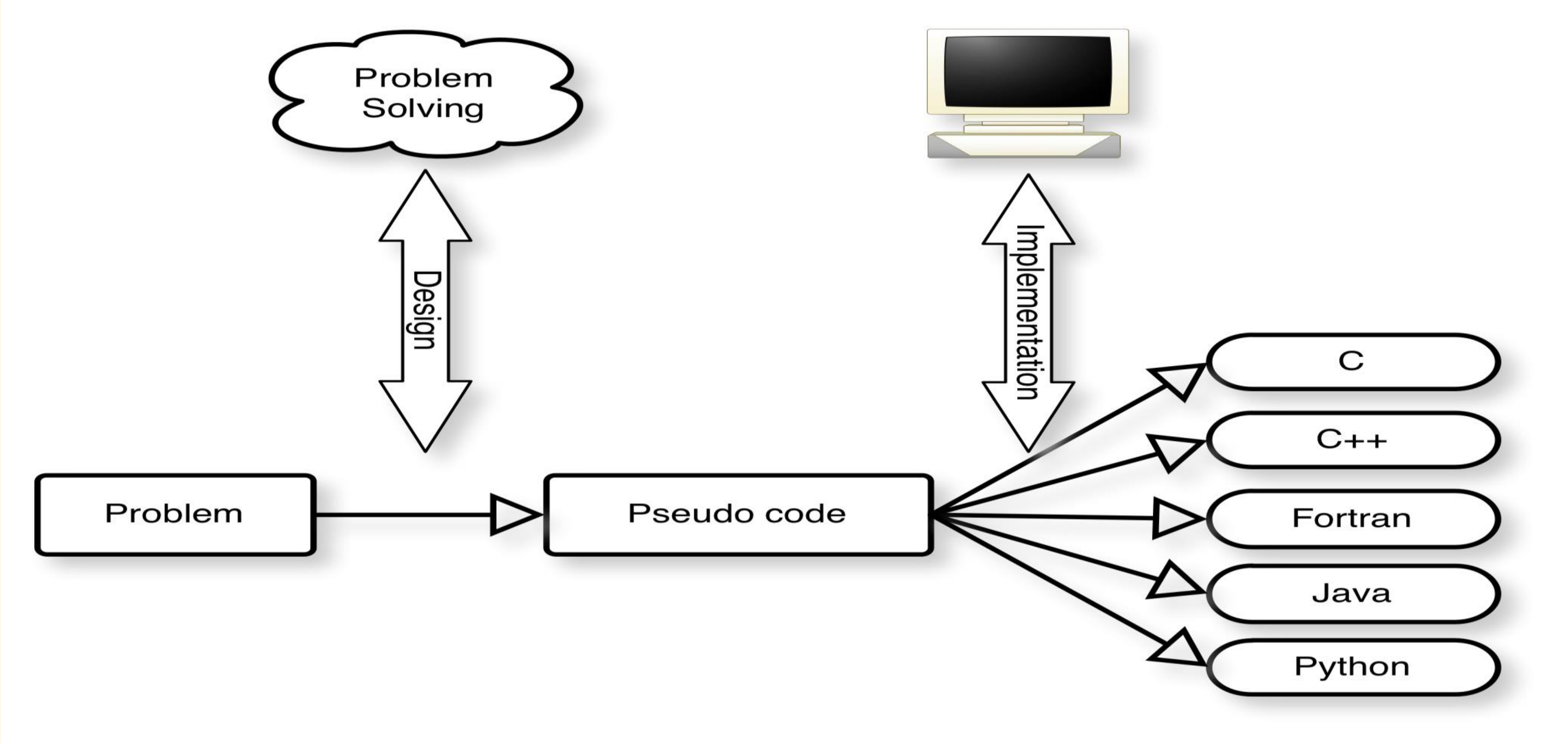
PROGRAMMING LANGUAGES

- Interpreted Languages e.g BASIC, Perl, PHP, VBScript, Power Shell etc
- Compiled Languages e.g Fortran, Java, C#, C++, Python etc.

INTERPRETED VS COMPILED LANGUAGES

- Programming with compiled languages gives more control, power, flexibility for numerically and logically intensive tasks
- Compiled code is translated directly from a high-level language into machine code instructions, which, by definition, are platform dependent.
- Interpreted code is translated from a high-level language into a set of meta-code instructions which are platform independent.
- Interpreted code is nowhere near as efficient, in terms of computer resource utilization, as compiled code

APPROACHING PHYSICS PROBLEMS



OUTLINE

- OVERVIEW OF COMPUTATIONAL PHYSICS
- TOOLS OF COMPUTATIONAL PHYSICS
- **INTRODUCTION TO JAVA**
- COMPONENTS OF JAVA
- HANDS-ON

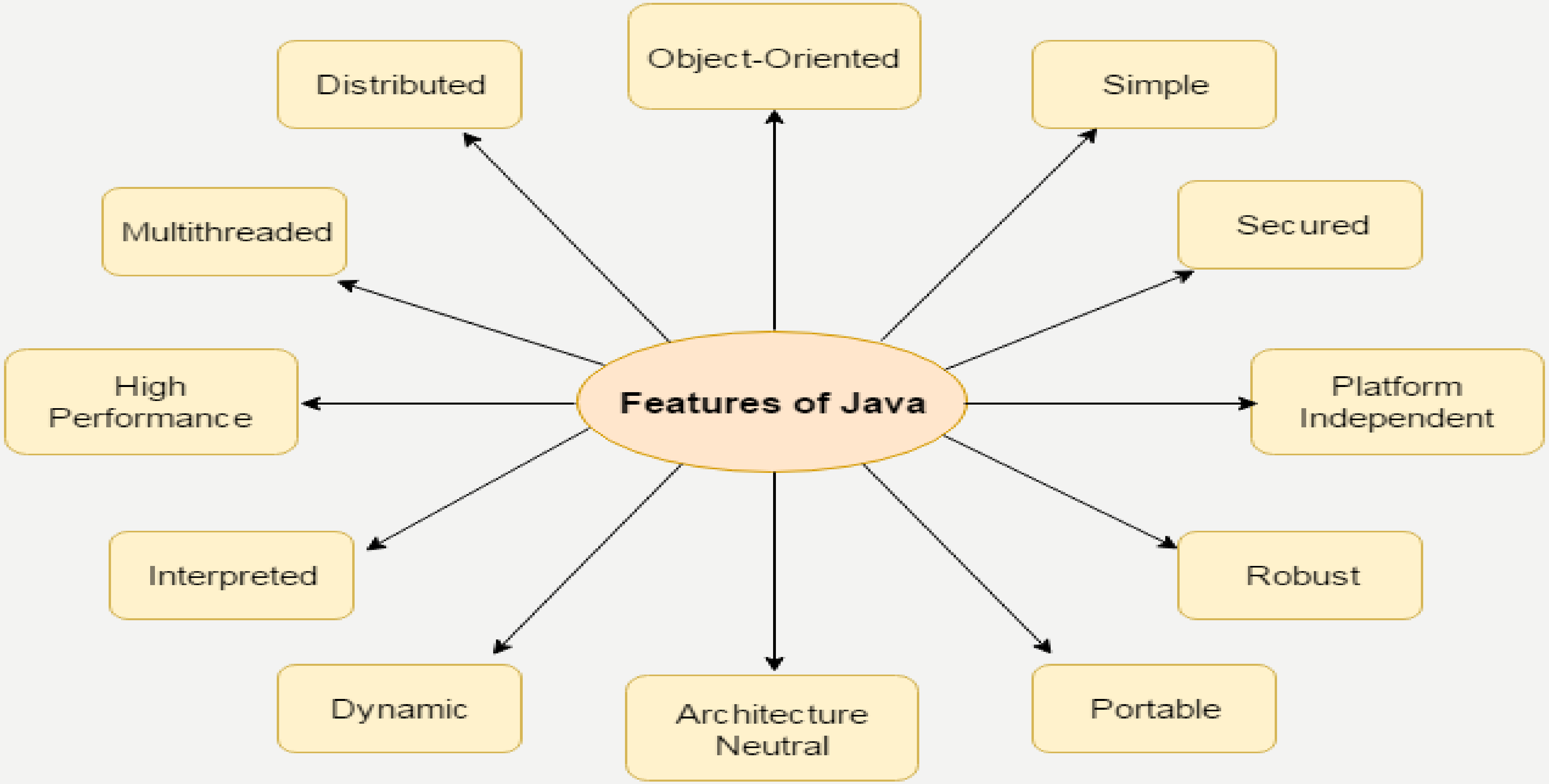
INTRODUCTION TO JAVA

- Java was originally developed by james gosling at sun microsystems in 1995
- Acquired by oracle corporation 2010

INTRODUCTION TO JAVA

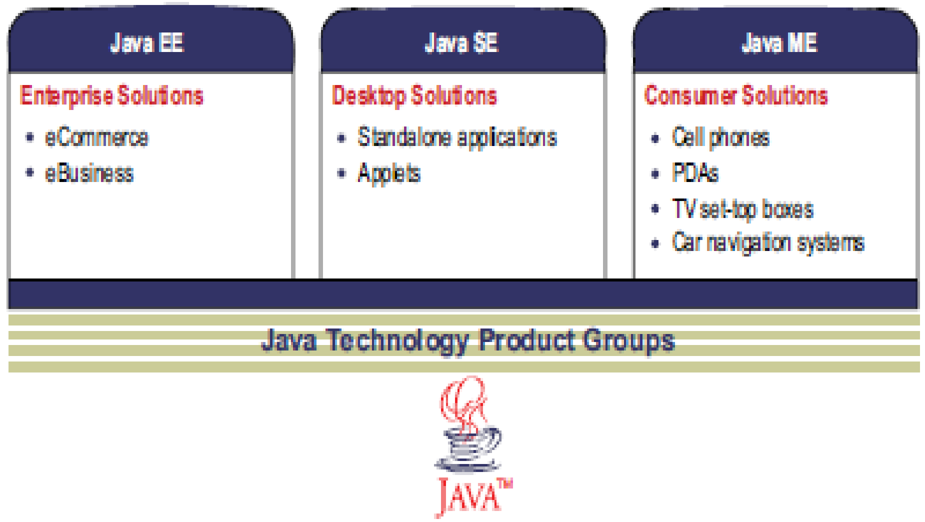
- The Java programming language is an excellent choice for learning, teaching, or doing computational physics.
- It is a well-designed, modern programming language that is simultaneously easy to learn and very powerful.
- It includes a range of features tailored for scientific computing, including features for handling vectors, inverting and diagonalizing matrices, performing Fourier transforms, making graphs, and creating 3D graphics.

FEATURES OF JAVA



FEATURES OF JAVA

- Java SE (Java Standard Edition)
- Java EE (Java Enterprise Edition)
- Java ME (Java Micro Edition)



PRIMARY GOALS OFJAVA

- Provides an easy-to-use language by:
  - Avoiding many pitfalls of other languages
  - Being object-oriented
  - Enabling users to create streamlined and clear code
- Enables users to run more than one thread of activity
- Loads classes dynamically; that is, at the time they are actually needed
- Supports changing programs dynamically during runtime by loading classes from disparate sources
- Furnishes better security

PLACES WHERE JAVA IS USED

- Java is implemented over a number of places in modern world.
- It is implemented as:

- ➢ Standalone Application
- ➢ Web Application
- ➢ Enterprise Application
- ➢ Mobile Application

- ➢ Games
- ➢ Smart Card
- ➢ Embedded System
- ➢ Robotics
- ➢ Desktop etc

REQUIREMENTS

- JAVA VIRTUAL MACHINE (JVM)

- JAVA RUNTIME ENVIRONMENT(JRE)
- contains JVM

- JAVA DEVELOPMENT KIT (JDK)
- contains JVM and Compiler

THE JAVA VIRTUAL MACHINE (JVM)

- JVM provides definitions for the:
  - Instruction set (central processing unit [CPU])
  - Register set
  - Class file format
  - Stack
  - Garbage-collected heap
  - Memory area
  - Fatal error reporting
  - High-precision timing support

JVM™ TASKS

The JVM performs three main tasks:

- Loads code
- Verifies code
- Executes code

GARBAGE COLLECTION

- Allocated memory that is no longer needed should be deallocated.
- In other languages, deallocation is the programmer's responsibility.
- The Java programming language provides a system-level thread to track memory allocation.
- Garbage collection has the following characteristics:
  - Checks for and frees memory no longer needed
  - Is done automatically
  - Can vary dramatically across JVM implementations

JRE vs JDK

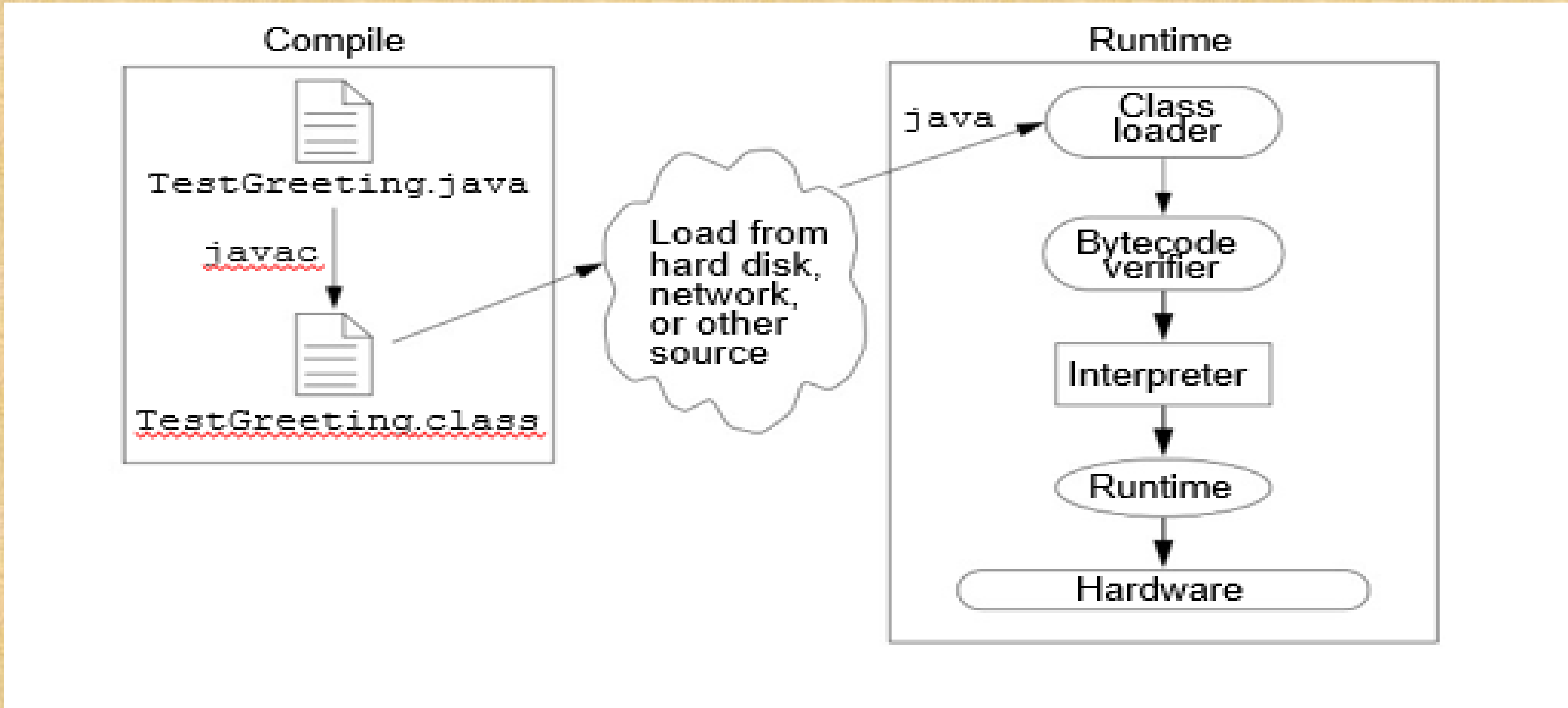
- **JRE** provides the libraries, the Java Virtual Machine (JVM), and other components to run applets and applications written in the Java programming language
- **JDK** is a superset of **JRE**, and contains everything that is in **JRE**, plus tools such as the compilers and debuggers necessary for developing applets and applications

SETTING UP THE ENVIRONMENT

- Install the Java Development Kit
- Set path of the jdk directory

- Create the java program
- Compile and run the java program

THE JAVA APPLICATION ENVIRONMENT PERFORMS AS FOLLOWS:



OUTLINE

- OVERVIEW OF COMPUTATIONAL PHYSICS
- TOOLS OF COMPUTATIONAL PHYSICS
- INTRODUCTION TO JAVA
- **COMPONENTS OF JAVA**
- HANDS-ON

COMPONENTS OF JAVA

- Class : Blueprint for an Object
- Object : Instance of a class
- Constructor: Object Creator
- Method : Behavior or actions
- Variable: Attribute or properties

COMPONENTS OF JAVA

- CLASS: Blueprint of an Object
- e.g.
- public class Animal {
- }

COMPONENTS OF JAVA

- METHOD: Behavior or action of an Object
- e.g.

```
public void eat(){
// eating goes here
}
public void move(){
// moving goes here
}
```

COMPONENTS OF JAVA

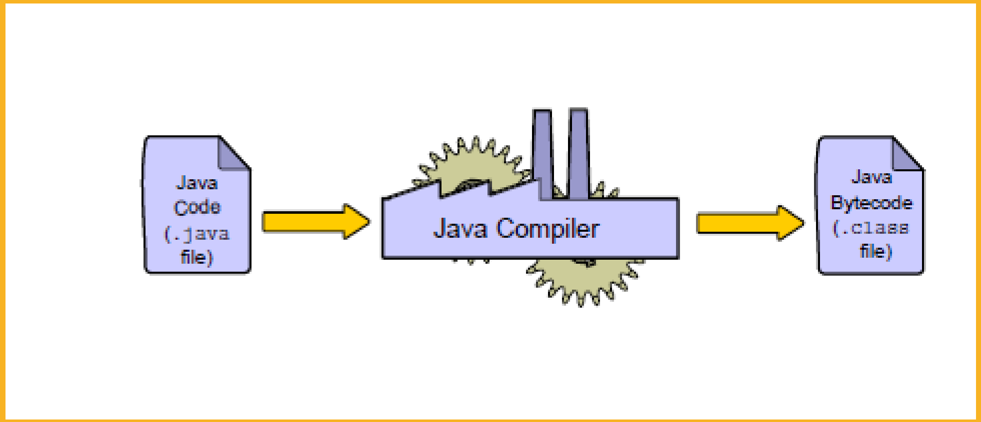
- VARIABLE: Attribute or property of an Object
- e.g.

```
int no_of_leg = 4;
String color = Blue;
```

SIMPLE JAVA CODE

```
public class Greeting{
   public static void main(String args[]){
   System.out.println("Hello Java");
     }
}
```

- TO SAVE: Greeting.java
- TO COMPILE: javac Greeting.java
- TO RUN: java Greeting



JAVA IDE TOOLS

- Java Integrated Development Environment(IDE) Tools:

- Eclipse
- Netbeans
- Intelli J

OUTLINE

- OVERVIEW OF COMPUTATIONAL PHYSICS
- TOOLS OF COMPUTATIONAL PHYSICS
- INTRODUCTION TO JAVA
- COMPONENTS OF JAVA
- **HANDS-ON**

HANDS-ON

- **Case study:** Equation of Uniformly Accelerated Motion

$$v = u + at$$

$$s = ut + \frac{1}{2}at^2$$

$$s = \frac{1}{2}(u + v)t$$

$$v^2 = u^2 + 2as$$

a = acceleration
v = final velocity
u = initial velocity
t = time taken
s = displacement

MISSING PARAMETERS

$v = u + at$	s
$s = \frac{1}{2}(u+v)t$	a
$s = ut + \frac{1}{2}at^2$	v
$v^2 = u^2 + 2as$	t

v
u
t
s
a

DETECT THE UNKNOWN

```
input v, u, a, t, s;
input result;
 IF s = ?      THEN result = s
 ELSE IF  v = ?  THEN result = v
 ELSE IF  u= ?  THEN result = u
 ELSE IF  a = ?  THEN result = a
 ELSE IF  t = ?  THEN result = t
```

ALGORITHM FOR EQUATION 1

```
• IF  s = null THEN
    CASE   result   OF
      v:      v = u + at
              print v
              break
      u:      u = v – at
              print u
              break
      a:   a = (v – u)/t
              print a
              break
      t:      t= (v – u)/a
              print t
    END CASE
  END IF
```

PROJECTS

- Deployed Oracle Anti-Money Laundry solution (MANTAS) in CENTIF Guinee Conakry and Bissau respectively.
- Deployed WebSphere, DB2 Purescale and Tivoli Workload Scheduler on Central Security Clearing System (CSCS) AIX Servers (Nigeria Stock Exchange).

- **E-Commerce:** alabanigeria.com gorgeouszone.com

- **Website:** uglybeat.com 9jalearn.com jetlinkghana.com chipbitssystems.com mazinwosu.com (In view)

- **Enterprise:** Conoil Supply Chain Management Solution, Jetlink Stock Inventory Solution, River State Gov't Procurement Solution (SEEFOR Project)

- **Mobile:** Mazi Nwosu (In view)

CONCLUSION