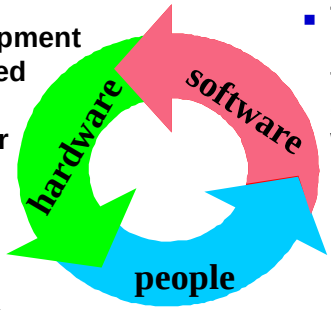
The Computer System

- The equipment associated with a computer system.



- The set of instructions that tell a computer what to do.

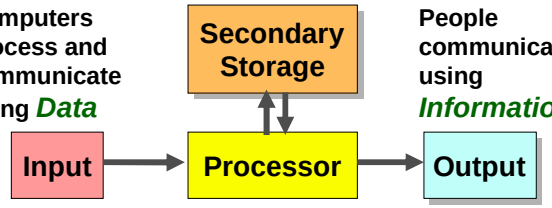
- Use the power of the computer for some purpose.

 1

Hardware = Physical Computer

Computers process and communicate using *Data*

People communicate using *Information*



- **Input** receives data (keyboard, mouse)
- **Processor** processes data (CPU, RAM Memory)
- **Output** produces information (Monitor, Printer)
- **Secondary storage** (Hard Drive, CD)

 2

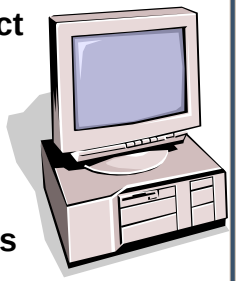
People = End Users & Programmers

- **End User's**
  - Utilize computer resources
  - Utilize software applications
- **Programmers**
  - **Analyze** a problem and create a solution algorithm
  - **Code** the solution algorithm into a specific programming language
  - **Verify** program works using known test data

 3

Software = Computer Programs

- ***Program:*** A set of step by step instructions that direct the computer to do a task that you want it to do and produce the results you want.
- ***Programming Language:*** A set of rules that instructs the computer what operations to perform.



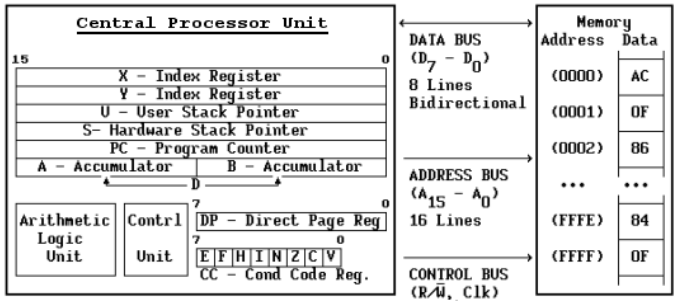
 4

Programming Language Generations

- 1st = Machine Language
  - Actual bits that CPU processes
- 2nd = Assembly Language
  - Each assembly instruction corresponds to one machine code instruction
  - Requires an assembler to convert assembly source code to machine code
- 3rd = High-level Language
  - Uses human words for keywords
  - Abstract and general purpose
  - Requires a compiler or interpreter to run
  - Compiles for different CPU's

MC6809 Simple 8-bit Computer

All Computer Systems contain minimally one CPU (Central Processor Unit) and Memory that are interconnected by the data, address, and control buses



First Generation: Machine Language

- Lowest level programming language because it represents data and program instructions as binary 0/1. Generally, hexadecimal is used for human interaction.
- All programming languages are eventually converted into machine language.
- Will be run on only one type of CPU

0000	
• • •	
D000	86
D001	12
D002	8B
D003	0C
D004	B7
D005	D1
D006	00
D007	BB
D008	D1
D009	10
D00A	B7
D00B	D1
D00C	01
• • •	
FFFF	

Second Generation: Assembly Language

Address	Instructions	Data	Assembly Language	
D000	86	12	LDA	#$12
D002	8B	0C	ADDA	#$0C
D004	B7	D100	STA	$D100
D007	BB	D110	ADDA	$D110
D00A	B7	D101	STA	$D101
D00D	8B	1E	ADDA	#$1E
D00F	B7	D01B	BCC	$D019
D012	86	00	LDA	#$00
D014	B7	D110	STA	$D110
D017	23	D007	BRA	$D007
D01A	3F		SWI	

Third Generation: High-Level Language

```
1.   // The File Name for this program is JavaExample1.java
2.   import java.util.Scanner; // Imports Scanner class
3.   public class JavaExample1 // Class has same name as file
4.   {
5.     // Program run starts here at public main method
6.     public static void main(String[] args)
7.     {
8.       Scanner inpEntry = new Scanner(System.in);
9.       System.out.println("Enter your first name: ");
10.      String sFirstName = inpEntry.nextLine();
11.      System.out.println("Enter your last name: ");
12.      String sLastName = inpEntry.nextLine();
13.      inpEntry.close();
14.      System.out.println("\n" + sFirstName + " your name "
15.        + "will appear as follows on the enrollment list:\n"
16.        + sLastName + ", "+ sFirstName);
17.    }
18.  }
```


High-Level Languages to Machine Code

- **Compiler**
  - Converts *HLL Source Code* into *Machine Code* file
  - Compiler targets only one type CPU
    - Intel: x86, 386, 486, Pentium 1-4
    - Motorola: 68k, Power PC, 68HC11
  - Compiler targets only one type OS
    - Microsoft: DOS, Windows
    - Unix, Linux, Solaris OS, Apple Macintosh, CPM
- **Interpreter**
  - Executes *HLL Source Code* line by line directly
  - Scripting Languages such as JavaScript, Python, Ruby, or BASIC utilize an interpreter to execute programs
  - Excellent *portability*

Historical Development of HLL

- **FORTRAN**: 1957, Compiled language, Developed for engineering and science applications.
- **COBOL**: 1959, Compiled language, Developed for business applications.
- **BASIC**: 1965, Interpreted language, Easy to program, Personal non-production applications; Resurrected by Microsoft in DOS and Visual Basic.
- **Pascal**: 1971, Compiled language, Developed at ETH Switzerland and used by higher education to teach Structured Programming methodologies.
- **C**: 1975, Compiled language, Procedural Oriented (verbs), Highly efficient fast programs, Usually eliminated need for assembly language programming. Structured programming.
- **ADA**: 1980, Compiled language, Developed as common HLL for Military applications; First to support Multitasking, concurrent execution of applications. Structured programming.

Common Object Oriented Languages

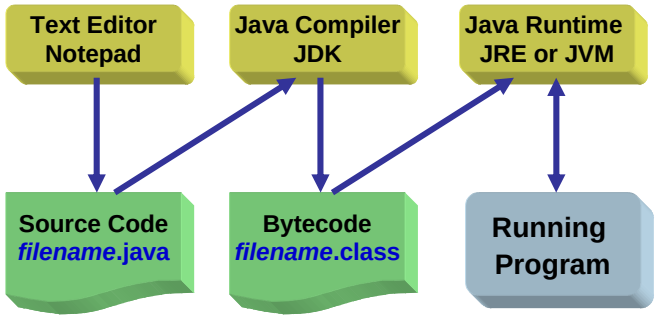
- **C++**: 1985, Compiled language
  - Added keywords to C so that could be used as Object Oriented Programming language
  - OOP focus is objects (nouns) instead of tasks (verbs)
- **Java**: 1994, Pseudo-Compiled language
  - Simplified Object Oriented Programming language
  - Supports Networking and Security
  - Supports Multithreaded for multitasking.
  - Compiler generates Bytecode which runs on JVM
  - Achieves OS and CPU Independence
- **Microsoft C#** : 1998, Uses .Net Framework
  - Much closer to Java then C++ and pseudo compiled
  - For Windows only products using Common Language Runtime (CLR like JVM)

Java Features

- **Simple** – Concise and cohesive features
- **Secure** – Creating Internet applications
- **Portable** – Any computer with JVM
- **Object Oriented** – Supports Philosophy
- **Robust** – Strictly typed and run-time checking
- **Multi-threaded** – Concurrent processing support
- **High performance** – Java bytecode is optimized for high speed execution
- **Distributed** – Designed with network and Internet support in mind
- **Dynamic** – Verifies and resolves access to objects at run time

 13

How Java Compiles and Runs Programs



 14

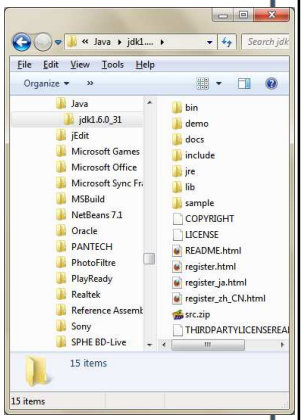
Java Programming Tools

- **Java Development Kit is the Java Compiler (freeware)**
  Java SE (Standard Edition) Development Kit version 6 update 31
  http://www.oracle.com/technetwork/java/javase/downloads/jdk-6u31-download-1501634.html
  - Oracle has now acquired Sun Microsystems
  - Download the version based on the operating system (80 MB)
    - Windows x86 = 32-bit versions (use this one if not sure)
    - Windows x64 = 64-bit versions
    - Linux versions also listed
  - Mac OSX already has JDK installed as part of operating system
- **Java SE Development Kit 6 Documentation**
  Java 6 API (Application Programmers Interface) documentation
  http://www.oracle.com/technetwork/java/javase/downloads/jdk-6u25-doc-download-355137.html
    - You can Download zip file at link above (58 MB)
    - doc sub-folder can be created within Java JDK folder
    - Searchable Online documentation available at: http://docs.oracle.com/javase/6/docs/

 15

The Subfolders of JDK 6

Folder	Description
/bin	The Java development tools and commands
/demo	Demonstration of Java capabilities with several Applets
/docs	Documentation for JDK can be downloaded and unzipped here
/include	C header files for combining C code with Java code
/jre	Root directory of Java Runtime Environment (JRE)
/lib	Libraries of code required by the development tools
/sample	Sample applications using various java components



 16

Java Programming Tools

- **NotePad++** (windows) freeware http://notepad-plus-plus.org
  - Freeware text editor can be used for single Java file programs
  - Need to install NppExec plugin and setup. Run using [F6]
  - Can be installed and run from USB drive http://portableapps.com
- **DrJava** (All OS's) freeware http://drjava.org
  - Light-weight Integrated Development Environment
  - Can compile and run single java file or multifile project
  - Can be installed and run from USB drive http://portableapps.com
    - Install JPortable and JPortable Launcher
    - Use JPortable Launcher to run drjava.jar file
- **Eclipse IDE** (All OS's) freeware http://eclipse.org
  - Heavy-weight Integrated Development Environment
  - Many features, steep learning curve, and requires projects
- **Netbeans IDE** (All OS's) freeware http://netbeans.org
  - Heavy-weight Integrated Development Environment
  - Many features, steep learning curve, and requires projects

First Java Program

- Using NotePad++ or DrJava enter this code and save to USB drive with filename Example2.java

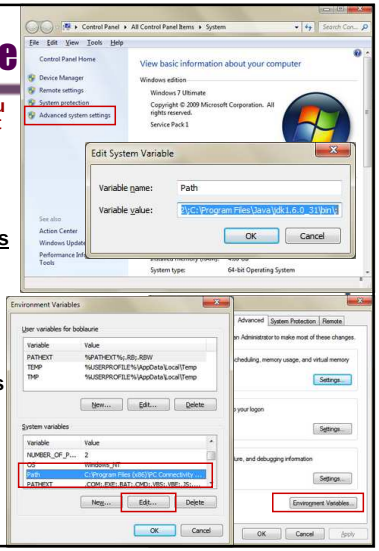
```
1.  // The File Name for this program is Example2.java
2.  // Author: Robert Laurie
3.  public class Example2 // Class has same name as file
4.  {
5.    // Program run starts here at public main method
6.    public static void main(String[] args)
7.    {
8.      System.out.println("My first Java Program "
9.        + "\nRobert Laurie");
10.   }
11. }
```

```
C:\Program Files\Java\jdk1.6.0_31\bin\java Example2
Process started >>>
My first Java Program
Robert Laurie
<<< Process finished.
```


Setting PATH Variable

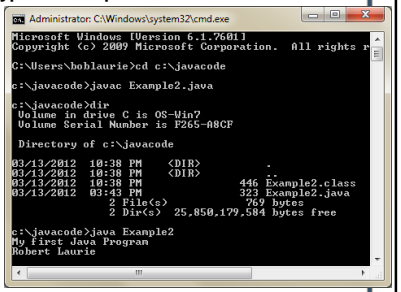
- The following is required only if you intend to run java from cmd prompt
- Proceed with extreme caution! Your system might crash if you make a mistake.

1. Start | Control Panel | System
2. Click Advanced System Settings
3. Click Environmental Variables
4. Edit Path Variable by inserting the path to the bin folder of your Java installation
5. Each path designator ends with a semicolon and no spaces should exist between them.
6. Click OK and OK
7. You may need to Restart Computer



Running from Command Prompt Window

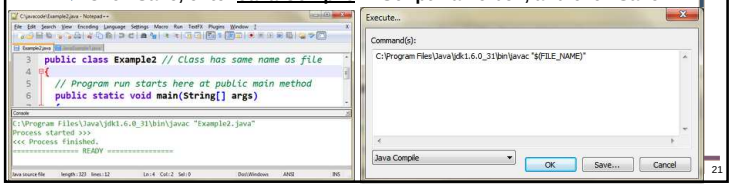
- Go to Start All Programs | Accessories | Command Prompt
  - or go to Start | Search and type `cmd` to open command window
- Go to Java Programs folder using DOS command `cd`
  `cd c:\javacode`
- Examine files in folder using DOS command `dir`
  `dir`
- Compile java program named Example2.java
  `javac Example2.java`



- Examine files to verify `Example2.class` created use `dir`
- Run Java program using JVM without file extension
  `java Example2`

Edit, Compile, and Run using NotePad++

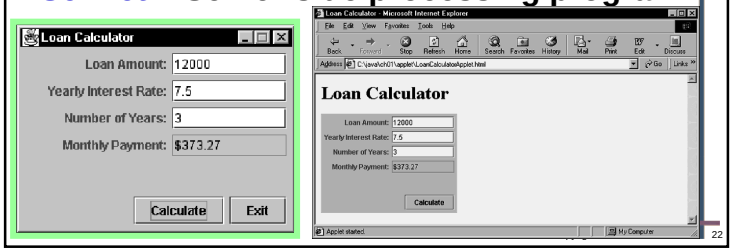
- Install NotePad++ to computer or to PortableApps USB drive
  - Go to menu Plugin > Plugin Manager > Show Plugin Manager
  - Check the checkbox for NPP_Exec plugin > Install > Restart
  - Push [F6] key to access this DOS Execution plugin
  - Enter the in the commands window the two lines:
    ```
    cd "$(CURRENT_DIRECTORY)"
    c:\Program Files\Java\jdk1.6.0_31\bin\javac "$(FILE_NAME)"
    ```
  - Click Save, enter Java Compile in Script Name box, and click Save
  - Push [F6] key again to access Npp_Exec plugin
  - Enter the in the commands window the two lines:
    ```
    cd "$(CURRENT_DIRECTORY)"
    c:\Program Files\Java\jdk1.6.0_31\bin\java "$(NAME_PART)"
    ```
  - Click Save, enter Java Compile in Script Name box, and click Save



21

Applications, applets, and servlets

- Application = Program that runs under OS utilizing Java Virtual Machine
- Applet = Program that runs within a web browser after retrieved from the Internet
- Servlet = Server-side processing program



22

Homework 1 – 25 points

1. Pick your favorite geometry formula (e.g., area of a square, perimeter of a triangle …) and implement it in Java. Also display your name as the author of the program in the output.
2. Create a program document *YourName*_hw1.doc with Title Page and create a Program Design section to include program specifications, test data, input/output, and design using flowcharts or pseudo-code.
3. Implement your program design using NotePad++ or DrJava and name your file *YourName*_hw1.java
4. Compile using Java SE 6 JDK compiler and debug until all syntax errors are eliminated: `javac YourName_hw1.java`
5. Note upon successful compile `javac` creates the file *YourName*_hw1.class
6. Demonstrate your code runs without logic errors by running the Java Virtual Machine on program: `java YourName_hw1`
7. Create a Program Implement section in *YourName*_hw1.doc and place your source code in this section. Do a screen capture or text copy of the output of your program and paste in this section.
8. Print out the *YourName*_hw1.doc word document and be ready to submit this for grading at the end of Class 2, Week 2.

 23