What is Programming?

“If you don’t think carefully, you might think that programming is just typing statements in a programming language.”

-- Ward Cunningham

“Any fool can write code that a computer can understand. Good programmers write code that humans can understand.”

-- Martin Fowler

“Refactoring: Improving the Design of Existing Code”

Discussion: What Is Good Software?

• What are its outcomes?
• What are the characteristics of the software?
• How can we write good software?
• What are short-term vs long-term goals?

Characteristics of Good Software?

• Free of bugs
  - Robust, reliability, stability
• Code is easy to read, extend, maintain
  - Readability, extensibility, maintainability
• Application is easy to use
  - Usability
• Efficiency
• Scalability

Course Content

• Software Design Principles
• Java
  - Statically typed language
• Software development, productivity tools
  - Eclipse
  - Version Control Systems
  - Some UNIX

What to Expect from this Class

• Programming intensive
  - Variety of assignments and projects
  - More freedom in design, "ilities"
    - Larger portion of your grade
    - Correctness is NOT enough
  - Building on large library of classes
  - Read others’ code! Learn from the good and the bad
  - Building larger applications
• Compare/Contrast with Python
  - PL design: what’s the best PL for your needs
• Learning on your own
  - Online resources
Learning Objectives

• Discuss software development and practices knowledgeably, using appropriate terminology
• Design, implement, test, and document efficient applications of increasing size and complexity
• Understand the designs and implementations of others
• Use a version control system, such as Subversion or CVS
• Use many of the capabilities of the Eclipse IDE
• Test and debug large applications systematically, using standard tools
• Understand design principles such as DRY and shy
• Discuss the benefits and limitations of a statically typed language

Feedback from an Alumnus

“I am schooling everyone at work on OO design and Java. Seriously, keep pounding OO design principles in. It is incredibly practical. I'm teaching CS majors and Computer Engineering grads about this. It's crazy how some (good) technical schools don't stress this more.”

Class Details

• Course Web Site
  ➢ Example code, lecture slides, readings, resources
  ➢ Course wiki
• Optional Textbooks
  ➢ Use plentiful online resources instead
• Participation
  ➢ Class discussions

• Programming Assignments
  ➢ Various sizes
  ➢ To start, a lot of short ones
• 2 Projects
• 2 Exams

Course Dynamics

• Professor's Responsibilities:
  ➢ Be prepared for class
  ➢ Provide constructive feedback to students
  ➢ Treat students with respect
  ➢ Challenge and encourage students
  ➢ Make material as clear as possible

• Student’s Responsibilities
  ➢ Be prepared for class (do readings and homework)
  ➢ Give attention and effort in class to learning
  ➢ Ask questions (during class and via email)
  ➢ Use professor's office hours
  ➢ Let professor know if something is going wrong
  ➢ Treat other students and professor with respect

My Bio

• From Dallastown, PA
• B.S., Gettysburg College
• M.S., Duke University
• Ph.D., University of Delaware
• For fun: pop culture, gardening
• Volunteer with Rockbridge Animal Alliance

From Dallastown, PA
B.S., Gettysburg College
M.S., Duke University
Ph.D., University of Delaware
For fun: pop culture, gardening
Volunteer with Rockbridge Animal Alliance

From Dallastown, PA
B.S., Gettysburg College
M.S., Duke University
Ph.D., University of Delaware
For fun: pop culture, gardening
Volunteer with Rockbridge Animal Alliance
My Research Interests

- General: Software engineering
- Automated testing of web applications
  - Develop algorithms
  - Implement in tools
  - Empirical studies
    - Try ideas out, see what actually happens, analyze

Student Survey

- Class years
- Where you’re from
- Any experience with Java

What is Java?
... and, why should I learn it?

- From Sun Microsystems
  - 1995, James Gosling and Patrick Naughton
  - Specifications
- Object-oriented
- Rich and large library
- Develop cross-platform applications
  - Web, desktop, embedded
  - Frameworks
- Widely used

What is Java?

- Java Programming Language
- Java Virtual Machine
- Java Class Libraries

Overview:
Writing, Executing Java Programs

Program.java
Written in Java Programming Language

javac
Compiles
Program.class
Bytecode: machine code for a virtual CPU

javaw
Executes
Java Programming Language

Java Programming Language

- Similar to Python
- But entirely object-oriented*

Step 1:

Program.java
Written in Java Programming Language

javac
Compiler
Program.class
Bytecode: machine code for a virtual CPU
Java Virtual Machine (JVM)

- Same bytecode executes on each platform
- Don’t need to provide the source code

Java Virtual Machine (JVM)

- Emulates the CPU
  - Usually specified in software (rather than hardware)
- Executes the program’s bytecode
  - Bytecode: virtual machine code
- JVMs available for each Java-supported platform
  - Enables program portability
- HotSpot VM
  - Code dynamically compiled to machine code
  - Garbage Collection

Traditional Way

- Executable is not portable

Java Class Libraries

- Pre-defined classes
  - Included with Java Development Kit (JDK) and Java Runtime Environment (JRE)
  - View the available classes online: http://download.oracle.com/javase/8/docs/api/

Benefits of Java

- Rapid development of programs
  - Large library of classes, including GUIs, Enterprise-level applications, Web applications
- Portability
  - Run program on multiple platforms without recompiling
- Statically-typed language
  - Compiling: find some errors before execution!
  - Can give performance boost by doing optimizations
Which ‘Java’ is this class about?

• Java programming language
• Java class libraries

• Use the JVM but won’t learn about how it works
  ➢ For more information: http://docs.oracle.com/javase/specs/

Aside: JavaScript vs Java

• JavaScript is not Java
  ➢ JavaScript is a scripting language, primarily embedded in HTML, executed by Web browsers

aside: JavaScript code

Java Development Kit

• JDK: Java Development Kit
• SDK: Software Development Kit
• Free from Oracle
• Contains
  ➢ javac: Java compiler
  ➢ java: Java Virtual Machine
  ➢ Java class libraries


Java Development Kit: JDK

• Installed on Linux machines
  ➢ Java 1.8 should be reachable using default path
  ➢ To see which executable you’re executing use `which java`
  ➢ Should be /usr/bin/java
  ➢ Run `java -version` to determine which version you’re running
  ➢ You can download the JDK for your machine from http://www.oracle.com/technetwork/java/javase/downloads/index.html
• JRE is for running Java applications
  ➢ Does not include the compiler

Summary

• Today
  ➢ Overview of Java programming language
• Next time:
  ➢ Example program
  ➢ Data types
• Your To Do
  ➢ Review UNIX commands
  ➢ Get comfortable with the command line
    • Run your Python scripts from the command line
  ➢ Check out course web site