CS 112 Introduction to Programming  
(Spring 2012)

Lecture #1: Introduction

Zhong Shao

Department of Computer Science
Yale University
Office: 314 Watson

http://flint.cs.yale.edu/cs112

Acknowledgements: some slides used in this class are taken directly or adapted from those accompanying the two textbooks: Introduction to Programming in Java: An Interdisciplinary Approach by Robert Sedgewick and Kevin Wayne and Building Java Programs: A Back to Basics Approach by Stuart Reges and Marty Stepp
Overview

What is CS112?
- A broad, programming-centric introduction to computer science

Goals.
- Demystify computer systems
- Empower you to exploit available technology
- Build awareness of substantial intellectual underpinnings

Topics
- **Programming** in Java
- Key program design techniques & problem-solving skills
- Programming tools & important libraries and data structures
- Applications to science, engineering, and commercial computing

“Computers are incredibly fast, accurate, and stupid; humans are incredibly slow, inaccurate, and brilliant; together they are powerful beyond imagination.”  – Albert Einstein
Why Programming?

Why programming? Need to tell computer what to do.

"Please simulate the motion of N heavenly bodies, subject to Newton's laws of motion and gravity."

Prepackaged software solutions. Great, they do exactly what you want.

Programming. Enables you to make a computer do anything you want.

Ada Lovelace

Analytic Engine

well, almost anything [stay tuned]
Languages

**Machine languages.** Tedious and error-prone.

**Natural languages.** Ambiguous and hard for computer to parse.

- Kids Make Nutritious Snacks.
- Red Tape Holds Up New Bridge.
- Police Squad Helps Dog Bite Victim.
- Local High School Dropouts Cut in Half.

[ real newspaper headlines, compiled by Rich Pattis ]

**High-level programming languages.** Acceptable tradeoff.

“Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do.” — Donald Knuth
Why Program?

Why program?
- A natural, satisfying and creative experience.
- Enables accomplishments not otherwise possible.
- Opens new world of intellectual endeavor.

First challenge. Learn a programming language.

Next question. Which one?

Naive ideal. A single programming language.
Our Choice: Java

Java features.
- Widely used.
- Widely available.
- Embraces full set of modern abstractions.
- Variety of automatic checks for mistakes in programs.

Java economy.
- Mars rover.
- Cell phones.
- Blu-ray Disc.
- Web servers.
- Medical devices.
- Supercomputing.
- ...

$100 billion, 5 million developers

James Gosling
http://java.net/jag
Learning a Programming Language

Just like learning any new language
- syntax: “new words”
- grammar: how to put them together
- programming: telling a coherent story
- library: use plots already written

Initially needs efforts, but pays off in the end!
Programming is like Legos...
Topics Covered

Intended mainly as your first programming course! The goals of cs112 are to learn:

program design techniques: how to model real world and manage complexity, e.g.,
- structural programming: top-down vs. bottom-up
- object-oriented programming: abstraction and modularity

problem-solving skills, e.g.,
- how to solve a problem efficiently, elegantly
- debugging: root cause analysis of problems; how to fix them
- novel problem solving skills: recursion; fun algorithms

good programming style, e.g.,
- robust, efficient program, readable code; documentation

how to use important libraries and data structures, e.g.,
- graphics, sound, animation, GUI
- arrays, strings, lists, stacks, queues

programming tools
- Java programming language and supporting tools & compilers
Course Personnel

Instructor: Zhong Shao  zhong.shao@yale.edu

http://www.cs.yale.edu/~shao/

Office: AKW 314

Teaching fellows: office hours and lab sessions to be announced

- Pedro Alves <pedro.alves@yale.edu>
- Yitzchak Lockerman <yitzchak.lockerman@yale.edu>
- Michael (Fitz) Nowlan <michael.nowlan@yale.edu>
- Aaron Segal <aaron.segal@yale.edu>
- Fei Xue <fei.xue@yale.edu>
- ......

Undergraduate lab assistants to be announced
Course Information

Textbook (required):

Introduction to Programming in Java: An Interdisciplinary Approach

Optional:

Building Java Programs: A Back to Basics Approach.

Other recommended books listed online

Information, lecture notes, & assignments are available on the course web site or the Yale class server
Course Workload

Two exams: one midterm, one final
- one exam == two assignments

Assignments: 9 -- 10 in all
- all assignments are posted online (no paper handout)
- all assignments must be submitted online
- Due on the date specified in the schedule page

They are usually fun and rewarding: n-body simulation, recursive fractals, critter simulation, guitar-hero, DNA sequence alignment,
......

One or two bonus [optional] assignments to replace lowest grades
Assignments Workload

All assignments are programming assignments
- from small and easy to large and more interesting

- Time spent on assignments reported by students (fall 2005)
How to do well in this course?

Keep up with the assignments

- The course material is cumulative
- From a former student: “Procrastination will eventually come around to bite you in the ass!”

If you don’t understand something, ask questions (especially “WHY?”).

- “There’s no such thing as a dumb question.”
- Computers are neither magical nor mysterious. Everything can be explained!

Discuss your design with others
Should you Take this Course?

No
- “I hate computers.”
- “I like fuzzy solutions.”
  - Programming forces you to solve problem precisely.
- “I refuse to think logically.”
- “I want to take an easy class.”
  - Hard for those who find difficulty in logical thinking and who don’t pay attention to precision.
Should you Take this Course?

Yes/no

- “I want free gourmet meals and to make lots of money by working for Google.”
- “World of Warcraft rocks hardcore!”
- “Everyone, look at my Facebook farm!”
Should you Take this Course?

Yes

- “Computers and robots are going to take over the world. I want to befriend them so that my life will be spared.”
- “I have to take this class.”
- “I like to solve problems.”
- “I like to learn new perspectives on how to solve problems”
- …
What Kind of Class is This?

Sort of engineering
- building things that realize a desired objective

Sort of philosophy
- formal logic, deductive systems and semantics

Sort of foreign language
- how to communicate with a computer effectively

Sort of psychology
- given some output behavior, what is going on internally?
What Kind Of Class is This?

Philosophy, Psychology, Foreign Language, Engineering, Politics, Sociology, Biology, Physics, and Business rolled into one!
Comment on Course Eval

“We only work with java, but honestly the class isn't about learning a language so much as it is learning how to think like a programmer. This class was great for that purpose, indeed, I would argue that such a class is actually what a QR requirement should be -- not merely quantitative reasoning but thinking in terms of a formal system and how to translate complex tasks into simple tasks that can be accomplished with formal statements.”
Miscellaneous points

Java is an extremely popular language for writing software in the real world

- But you will not be hirable as a Java developer just from taking this class

So why should you take this class?

- It will develop your logical mind
- Gives you a flavor of CS
- 65% of you will have to talk to software developers on a regular basis in whatever profession you choose when you graduate --- this class will help you communicate with them
Comment on Course Eval

“Basically, programming is just problem solving. If you think you're someone who likes to solve problems, then definitely take this class. [...] I do have to warn you about the problem sets, though. They start out very easy, but get pretty hard because you have to build on your prior knowledge. Debugging your program can be the most annoying thing in the world, since hitting the shift key while typing the equals sign can cost you about 5 hours' worth of time looking for it (personal experience). So start on the problem sets early, and don't hesitate to ask the TA's for help--you'll probably need it at least once, such as on the last problem set.”
Programming

It’s about solving problems

Patience is a virtue
  - Don’t wait until the last minute

Don’t blame the computer