CS1110 Fall 2011: David Gries & Steve Marschner

website: www.cs.cornell.edu/courses/cs1110/2011fa/

CS1110: Java
- No prior programming experience
- No calculus
- Non-numerical problems
- Later assignments: processing images, games, playing music

CS1112: Matlab
- No prior programming experience
- One semester of calculus
- Math- & engineering-type problems

1

CS1130: Transition to OO (using Java)
CS1132: Transition to Matlab
- Both require previous programming experience.
- Self-paced, 1-credit (4 weeks), S/U. Engineers take CS1110–CS1132 or CS1112–CS1130.

CS2110 Computers & Programming
- Uses Java
- Prerequisite: CS1110 or CS1130.

CS1110 course outcomes:

(1) Basic understanding of object-oriented and procedural aspects of programming, as expressed in Java.

(2) Fluency in Java —ability to write programs using classes and subclasses, as well as assignments, conditionals, recursion, and loops.

(3) Knowledge of the basic API classes and their specifications.

2

Assignments
(These may change, but this list gives you an idea of what you will be doing)

A4: Color models

A5: Graphics packages —it's turtles all the way

A6: Editing image files (jpg, png, …)

A7: Breakout

3

Methods to increase chances of success in the course.

1. Section/labs. In **the ACCEL Lab**, Carpenter Library 2nd floor. Guided exercises on computer, with TA and consultants walking around, helping. Mandatory.

2. Quizzes. Let you know what material is important for you to know at that point. You will know exactly what the quiz will cover. Everyone expected to get A on each quiz.

3. Lectures: not 45 minutes of talking. See demos of programming and execution of programs almost every lecture. Also, some interactive work with you. We try to make it interesting.

4. Course text: CD at the back of book has 250 2-4 minute lectures, each on 1 specific point. CD missing? See course website.

5. One-on-one sessions beginning 3rd week. Work for 30 minutes with Gries, Marschner, TA, or consultant on the computer.

4

---

6. First prog assignment, everyone eventually scores 10/10. Requires mastery. Submit, get feedback, resubmit, … until it is right.

7. "Interludes", discuss some aspect of computing, internet, or CS to help you understand the computing world we live in. Tidbits on time management, study skills, etc.

8. AEW Workshops. 1 credit, 2 hours. No homework. Small, collaborative classes parallel to course. No class first week. See link on course website, talk to advisors in Olin 167.

9. Iclickers. Everyone: get your own clicker. By Tuesday. We use them to judge the sense of understanding of the class, to encourage staying alert, perhaps to give quizzes, attendance, etc.

10. Piazza. Our "town square" —ask and answer questions.

Course Management System. Visit cms.csuglab.cornell.edu/ Not listed? Email Maria Witlox, mwitlox@cs.cornell.edu, ask to add you to CS1110 CMS. Include your Cornell netid in your email.

Academic Integrity. We ask you not to cheat, in any way, shape, or form. On our side, we try our best to be fair about the amount of work, in grading the work, and in giving you a course grade. For more info, see course website. **Do Quiz 0 on Course CMS.**

Recitations (Labs) in the Engineering ACCEL LAB

ACCEL Lab: into the Engineering Library in Carpenter Hall, walk straight until you come to a staircase on your left, go up the stairs.

Register for ANY recitation/section. But go to the one you want.

Times of the recitation-labs: Attend ONE of them.
Tuesday: 12:20, 1:25, 2:30, 3:35
Wednesday: 12:20, 1:25, 2:30, 3:35

6

Four things to do this weekend

1. Get on Course Management System (CMS). Get to it from link on course website.

Not registered on our CMS? Email Maria Witlox and ask her to register you.
mwitlox@cs.cornell.edu
She needs your netid.

2. Get DrJava onto your computer. See course website, Piazza for info. Problems with it? You can wait, but do try.

3. Academic Integrity. Read about it on course website. Visit CMS and do Quiz 0.

4. Read in course text (see next slide)

course website: www.cs.cornell.edu/courses/cs1110/2010fa/

7

Reading for this and the next lecture:
Sections 1.1, 1.2, 1.3. Lab 1 will give you practice with concepts and details of 1.2, 1.3. You will not understand all the reading because there are many new terms, but doing the reading will enhance next lecture.

PLive: Lesson 0, Lesson page 1.3, Activity 1-4.1.

Summary of lectures: On course website, click "Lecture summaries".

Today and Tuesday:

• Introduce expressions in Java (using DrJava)

• Show you around the CD **ProgramLive**

DrJava. We write programs using the free IDE (Integrated Development Environment) called DrJava. Download it from the course website.

8

Terminology

Programming language (Java, C, Fortran, Matlab, Python): a language in which you write programs, often to be executed on a computer.

Program: A set of instructions, written in a programming language, to be executed (carried out, performed) to get some task done. Like a recipe in a cookbook.

Machine language. The language of instructions that a computer is able to execute (carry out, perform).

Java Compiler. A program that translates a Java program into a machine language form so that it can be executed on a computer.

9

Type: A set of values together with operations on them.

Memorize this definition!
Write it down several times.

Type **integer**:

values: …, –3, –2, –1, 0, 1, 2, 3, 4, 5, …

operations: +, –, *, /, unary –

Type **int**: -2^{31} .. $2^{31}-1$

values: –2147483648, –2147483647, …, –3, –2, –1, 0, 1, 2, 3, 4, 5, …, 2147483646, 2147483647

operations: +, –, *, /, unary –

10

Type: A set of values together with operations on them.

mantissa   exponent

Type **double**:

values: Examples: –22.51E6   equivalent to –22510000
                              or $-22.51 * 10^6$

                   22.51E–6   equivalent to .00002251
                              or $22.51 * 10^{-6}$

An approximation to the real numbers.

operations: +, –, *, /, unary –

Type boolean

Values: **true   false**

Operators: and && 　or ||　 not !

11

Precedence of operators (page 23)

• Unary operators: + – !

• Binary arithmetic: * / %

• Binary arithmetic: + –

• Arithmetic relations: < > <= >=

• Equality relations: == !=

• Logical and: &&

• Logical or: ||

The next lecture will also discuss:
Types boolean (p. 20) and String (p. 22)
You will use these things in Lab 01.

12

2