

SYSC 1101 Object-Oriented Software Development

Winter 2006 Course Introduction

Copyright © 2006 D.L. Bailey, except as noted

Instructor

- D.L. Bailey, Room 3042 MC, bailey@sce.carleton.ca
 - Office Hours: TBA on the course Web site

Course Outline

- Distributed in class and available on the course Web site
- Questions? Ask me - the sooner, the better

Lecture and Lab Schedule

- Lectures: Mon. and Wed., 10:00 - 11:30, 417 SP
- Lab: Tues., 15:30 - 16:30, 507 AA
 - I've reserved the lab for an extra hour on Tuesdays, from 14:30 - 15:30 (most of you have a "free" hour between MATH 1104 and the scheduled SYSC 1101 lab?)
- You can use any of the undergrad SCE computer labs (4233 ME, 4324 ME, 507 AA, 508 AA, 6050 MC, 6055 MC) whenever ME and MC are open, except when the labs are reserved for specific courses

Course Web Site

<http://www.sce.carleton.ca/courses/sysc-1101>

- Parts of the site are password protected
 - User name: supplied in class
 - Password: supplied in class
- I will not be using WebCT in this course

Computer Accounts

- Verify that you can log on to your SCE undergrad lab account from last term (the one you created for SYSC 1100)
 - The password should be unchanged from last term
- Report any difficulties/problems ASAP
- Don't wait until next week's lab!
- If you don't have an SCE lab account, you'll need to drop by one of SCE labs (4233 ME, 4324 ME, 507 AA, 508 AA, 6050 MC, 6055 MC), power up a computer, and follow the instructions to create your account

Contacting Me via Email

- Students must activate and monitor their Connect email accounts (see *Academic Regulations of the University*, Section 3.5)
- Many of Carleton's administrative offices/academic advisers will only respond to student email sent from Connect accounts
- For this course, I will accept mail from accounts originating at Carleton
 - Connect accounts
 - Accounts from other Carleton domains; e.g., sce (Systems), doe (Electronics), scs (Computer Science), etc.
 - Engsoc accounts
- Mail with "SYSC 1101" at the start of the subject line is routed to its own folder and receives the highest priority
- All mail from free email services (hotmail, yahoo, gmail) and most ISPs is discarded without being read
 - A student who attempts to arrange a meeting by sending a message with the subject line "Wanna meet u" from zippyd@yahoo.com will wait a long time for a reply...
- Special cases: if there's a good reason for you to email me using a corporate email account (e.g., you're a part-time student working at Nortel or Cognos), provide me with your email address so that I can add it to my email filter rules

Getting Advice & Assistance

- Instructor office hours will be posted on the Web site
- Meetings can be arranged for times other than the posted office hours, subject to my schedule
- If my office door is open, you are more than welcome to interrupt me to ask questions
- If I'm not in whenever you drop by, please don't wait until the end of term and then complain bitterly that I'm never around
 - *Speak to me after class or send me an email*
- Questions and comments of general interest (e.g., items related to the lectures and assignments) can be emailed to me
 - *These will normally be answered in the next class, not via email*
- "Private" questions (e.g., PMC accommodations, marks, illness, etc.) can be dealt with via email or by booking an appointment with me

Pedagogy: a Cautionary Note...

"27 October, 2024, 10 A.M.

Lanie watched as the youngsters, from preschoolers through high-schoolers, got off the bus. They looked frail and frightened.

"Learning" was being reevaluated, and the tech schools represented a new direction in education. Their primary subject was Wristpad 101. It taught children how to manipulate the computer net through their pads, how to access absolutely anything they'd ever need or want to know. The proliferation of voice lines on the pad even precluded the need for reading and writing. The power of the pad was the power of absolute knowledge. But what about discipline? What about memory storage and retrieval? Stealing one last glance at the line of twenty children, Lanie followed Newcombe into HQ. Tech kids - they had a poor ability to synthesize and react to physical demands and emotional situations. They lived in the pad *They thought it gave*

them everything, all the answers. The problem was, they didn't know the questions."

Richter 10, Chapter 9, Arthur C. Clarke and Mike McQuay, 1996, Bantam Books, © Clarke & McQuay

Pedagogy: Learning by Experimentation

"Our aim is to produce practitioners who not only understand the basic ideas but also have the confidence and ability to perform complex programming tasks even when the tasks appear, on the surface, to be beyond their capabilities.

We subscribe to the notion that learning is a series of experiments in which negative results are just as important as positive results. The task of the teacher is to lead the experiments and summarize what has been learned rather than simply present the conclusions to be accepted without question. Indeed, the idea is to foster an environment where questioning is the modus operandi. We learn because we doubt or mistrust the seemingly obvious...

[We adopt] a discovery approach to learning; i.e., together, we hypothesize, experiment, and discover as many of the important concepts as we can. This discovery process can at times be relatively lengthy, but it has the side-effect of providing deeper and more detailed understanding...

[We learn] to be curious, to experiment, to ask questions, and to attempt to answer them - prerequisites for any successful learning."

Discovering Smalltalk, Wilf Lalonde, © 1994 Benjamin/Cummings

More Pedagogy

- Problem-based learning vs. courses organized around programming language constructs
- Assignments drive lectures vs. lectures drive assignments
- Apprentice-based learning

Programming Language

- Sun Microsystems currently supports two versions of Java
 - Java 1.4 (current release is 1.4.2_10)
 - Java 5.0 (current release is 5.0 Update 6)
 - Java 5 provides increased compile-time type-checking as well as several other new features (new syntax)
 - The compiler will issue warnings when compiling many Java 1.1-1.4 classes
 - Usually these warnings can be ignored, although developers are encouraged to rewrite their code to compile without warnings under Java 5
 - We will use Java 5.0 in this course for the first time
 - Both Java 5.0 Update 4 and Java 1.4.2_09 are installed on our lab computers, so we have the older version to fall back on if there are problems

Programming Tools - BlueJ

- BlueJ is an excellent, free, IDE for teaching object-oriented programming to students

- It's written in Java, so it runs on any platform that supports Java 2, Standard Edition (Windows, MacOS X, Linux...)
- I've used the "Educational" or "Student" editions of several IDEs (for various languages), and I doubt if the developers of most of these IDEs ever stood in front of a classroom before specifying the feature list
- BlueJ was developed at several universities by people who teach OOP, and clearly understand what tools novice programmers need (and don't need)
- Don't be fooled by the simple user interface (compared to commercial IDEs):
 - BlueJ supports the complete Java language (it uses the Java development tools that are available from java.sun.com)

Other Programming Tools - JCreator LE

- A free IDE that has a "look-and-feel" that is similar to many commercial IDEs
- Installed on our lab computers, used in SYSC 1101 last year, but you don't need to use it this year
- JCreator uses the Java development tools that are available from java.sun.com
 - As shipped, it can run `javac` and `java` (the Java compiler and interpreter)
 - Configuring JCreator to run `javadoc`, `jar` and `junit` can be a bit painful for novices
- The major drawback (for this course) is that the LE version lacks a debugger

Other Programming Tools - Eclipse

- *"Eclipse is an open source community whose projects are focused on providing an extensible development platform and application frameworks for building software. Eclipse provides extensible tools and frameworks that span the software development lifecycle, including support for modeling, language development environments for Java, C/C++ and others, testing and performance, business intelligence, rich client applications and embedded development."* - www.eclipse.org
- The latest release of the Eclipse SDK is Release 3.1.1
- Release 2.1 is installed on our lab computers (for reasons that are out of my control, we can't upgrade to Release 3.1.1 this term)
- If you want to use this course as an opportunity to learn Eclipse, you are free to do so
 - I haven't used Eclipse much (learning it is on my to-do list), so neither I or (probably) the TAs will be able to help you with Eclipse-related problems

Other Programming Tools - JBuilder 3 Educational

- Don't use it
- It's based on Java 1.2.something, it lacks many of the class libraries that are part of Java 1.4 and Java 5, its online help is atrociously slow, it doesn't support many of the things you'll need to do (e.g., run `Javadoc` to create documentation pages for your classes, build `JUnit` test suites, create `JAR` files, etc.)
- Don't use it
- Even if you used it in SYSC 1100, don't use it
- If you drop by my office to tell me that you're having problems with JBuilder, guess what I'm going to tell you...

Textbook

- *Objects First With Java: A Practical Introduction Using BlueJ, Second Edition*

- One of the authors is one of the key developers of BlueJ
- This book is used for a two-course sequence in First Year Computer Science at the University of Kent
 - Chapters 1-6 are used in the first course (somewhat similar to SYSC 1100, except no inheritance)
 - Chapter 7 and Part 2 are used in the second course

Textbook

- *Objects First* is the best textbook I've found for introducing object-oriented programming to 1st and 2nd-year students
 - See the *Preface to the Instructor* (available online at <http://www.bluej.org/objects-first/evaluation>)
 - Chapters 1, 2 and 7 are available from the same page - try before you buy
- The book uses Java 1.4, but the authors have prepared a supplement (available online) that discusses Java 5
 - They have also posted Java 5 versions of many of the book's projects
- This textbook was used in the Fall 2005 offering of SYSC 2004, so used copies may be available
- My approach to teaching the course material will be very similar to that approach used in the book (see the Preface)
- With some exceptions, I won't be lecturing directly from the textbook (I see no point in repeating examples that have already been explained by the authors)
 - Currently, I plan to present the material from Chapters 7 and 11 pretty much verbatim
- I will present most topics in pretty much the same order as the book, although I may teach some topics from later chapters fairly early in the term, after we've covered the prerequisite chapters/sections
- Given that you have SYSC 1100 under your belts, we will move through Chapters 1-6 very quickly
- Most of the course will be spent on topics presented in Part 2

Lecture Materials

- I will post the source code for the classes that will be discussed during lectures
- I will post summaries of the lectures (including cross-references to the relevant sections in the textbook)
- I will not be posting complete sets of lecture notes
 - I expect students to attend lectures and take notes
- Don't think you can get by simply by downloading and reading the slides from previous offerings of the course
- This year:
 - The ordering of the topics has changed
 - Almost all the examples are new
 - Some topics have been removed or deemphasized
 - Increased emphasis on software engineering concepts (e.g., unit testing, introduction to coupling, cohesion, & refactoring)

Resources

- The course Web site will have links to useful Web sites (Sun Developer Network/Java Technology, BlueJ, etc.)

- To save you some time, I've produced a CD containing the software tools you'll need
- Arrangements for you to obtain a copy will be discussed in class (maybe burn CDs in the first lab session?)

Some Observations about Assignments and Final Grades

- An increasing number of students submit fewer than 50% of their assignments for grading (some submit no assignments!)
 - not just this course, but across the board in Years 1, 2 & 3
- Some students submit the posted solutions from previous years
 - even in courses where the assignments are marked but contribute 0% to the final grade!
- What grades do you think these students get?
- **Do the assignments**
 - The practice is invaluable preparation for subsequent assignments and the exams (see next slide)

Favourite Kinds of Exam Questions

- Take a class (or classes) developed in an assignment, design and code additional methods
- Take a class (or classes) developed in an assignment, change the fields (instance and class variables), reimplement some of the methods to use the revised fields
- Design and code a class that you haven't seen before, given the documentation for its interface
- You can't expect to answer these kinds of questions if all you've done is memorize other people's assignment solutions

Version 1.00, revised January 4, 2006