Syllabus: INFO I211
Information Infrastructure II

Indiana University East

Spring, 2013

Version 1.7, 2013 Jan 11. OFFICIAL.

Course Web Site: http://mypage.iu.edu/~gdweber/info/i211/

Scheduling Information

Time and location: Sec. 1341: Lecture ARR; Lab ARR.
Sec. 1761: Lecture ARR; Lab Mon Wed 2:00–3:15, Tom Raper 102.
Sec. 1762: Lecture ARR; Lab Mon Wed 5:30–6:45, Tom Raper 102.

Frequency: Every spring semester

Start date: Monday, January 14

Last date to register: Jan 19

Final exam: Tues–Thurs, Apr 30–May 2

Introduction

Course Description

I211 Information Infrastructure II (4 cr.) The systems architecture of distributed applications. Advanced programming, including an introduction to the programming of graphical systems. Credit given for only one of the following: INFO I211, CSCI A202 (IUB), or CSCI C212 (IUB) [or CSCI C202 (IUE)].

Prerequisite

INFO I210. See comments below.
Comments

This course is the second semester of computer programming and covers the Java language from scratch.

The prerequisite course at IU East, INFO I210, covers Python programming, but knowledge of any high-level programming language should be sufficient preparation. Generally, students should be familiar with decision and iteration structures (if, for, while), functions, recursion, objects, and classes. The lecture will assume students are familiar with these prerequisite concepts using some programming language (not necessarily Python). The textbook assumes no prerequisite knowledge. Good students with no programming background may be able to succeed in INFO I211; however, they may find the course intense.

Why, after learning Python, are we now learning Java? An experienced programmer will not only know a variety of programming languages, but will know when to use one and when to use another. Some languages are more suitable for some kinds of tasks. For example, Java is a more efficient language than Python, and therefore more suitable for computationally intensive tasks. On the other hand, Java requires more work on the part of the programmer, increasing the cost of program development. Sometimes it is necessary to do the extra work to make programs run more efficiently.

A good programmer also will usually be able to pick up a working knowledge of a new programming language in a couple of weeks. The first language is typically a serious challenge for students; the second language, a bit less so. The third, fourth, and later languages are usually much easier to learn.

Two phrases in the course description require explanation: “systems architecture of distributed systems” and “programming of graphical systems.” Distributed systems are cooperating programs (typically running on different computers) which send messages to each other through the network to share information and/or coordinate their actions. This course covers the bare basics of distributed systems; you can learn much more in INFO I320, Distributed Systems and Collaborative Computing. Graphical systems could be programs using graphical user interfaces (windows, icons, menus, mouse, etc.) or simply drawing pictures; we will cover both. However, these two topics are not the whole or even the main course content. Rather, this course is a balanced introduction to Java programming, covering those two topics among many others.

This course is required for the BS in Informatics and for the MIS Concentration in the BS in Business.
Learning Objectives

Indiana University East has specified seven learning objectives for undergraduate students.\footnote{See \textit{IU East Catalog}, http://www.iue.edu/catalog/policies/learning_objectives.php.} Within this context, the learning objectives for this course are:

“2. \textbf{Depth} Educated persons should have achieved depth in some field of knowledge. . . .”

- \textit{Understand and be able to apply the fundamentals of designing software systems and developing Java programs in the environment of a Linux/GNU operating system.}

To “understand” a programming concept means to be able to interpret algorithms and programs that use it; to “apply” a programming concept means to be able to design and write algorithms and programs that use it.

- Create and revise Java programs using text input and output, variables, assignment, arithmetic and string operations, classes and objects, inheritance, polymorphism, conditionals, exceptions, loops, arrays, recursion, data collections, graphics, and graphical user interfaces.

- Know the basic Unix commands for working with files and directories, compiling and running Java programs.

“3. \textbf{Expression} Educated persons should be able to express themselves clearly, completely, and accurately. . . .”

- Use good programming and documentation style, which helps us communicate the ideas of our program to other programmers or to ourselves at a later date.

“5. \textbf{Problem Solving and Critical Thinking} Educated persons should have the ability to develop informed opinions, to comprehend, formulate, and critically evaluate ideas, and to identify problems and find solutions to those problems. . . .”

- Programs are written to solve problems. As we design, code, debug, and test programs, we will be exercising and developing our problem solving skills.

- Use top-down and object-oriented design strategies to solve moderately complex problems.

- Debug and test programs effectively, including unit testing.
Learning Resources


Web sites (required) (Please memorize these and/or set bookmarks)

Public site http://mypage.iu.edu/~gdweber/info/i211/
Oncourse https://oncourse.iu.edu/

Software (required) All that you need is provided in the Linux lab (TR 102) and on our merlin server, but what if you want to work outside of the lab?

Primarily, we will be using a text editor and Java compiler or an integrated development environment (IDE) for Java. Java 7 (either OpenJDK or Oracle) is highly recommended; Java ≥ 5 is required. You need the JDK, not just the JRE.

Use any text editor or IDE that you like. Unix text editors include Emacs (my personal favorite), vi, GEdit; there are many more. Windows users will find Notepad++ a much nicer tool than Microsoft Notepad.

Simple IDEs include DrJava (recommended), BlueJ (okay), and jGRASP (not recommended). More complex IDEs include Eclipse and NetBeans; they are more powerful, but harder to learn.

Students will need access to a Unix-like shell. This is provided in the Linux Lab (TR 102) and by remote login to the merlin server using secure shell (ssh) or virtual network computing (VNC).

Macintosh OS X is Unix based and provides a shell.

Students wanting a Unix-like shell on their Microsoft Windows PCs may consider installing MinGW (including MSYS) or Cygwin.

Students will need a web browser that adequately implements the standards (“recommendations”) of the World-Wide Web Consortium (W3C). The most recent versions of popular browsers—Mozilla Firefox, Chromium or Google Chrome, Opera, Apple Safari, and Microsoft Internet Explorer—are doing a mostly commendable job of trying to keep up with the standards (which keep changing). If you have an older web browser (especially Internet Explorer < 9), you should upgrade it to the latest stable version. No matter which “brand” of browser you use, if it’s below the latest stable version, you are open to security hazards as well as not having the latest and greatest browsing experience, so upgrade today!

2http://www.w3c.org/
Learning Activities and Measurement of Learning

Reading

Each student is expected to read the assigned chapters of the textbook. It is most beneficial if you read the chapters before attending the corresponding lectures. Read carefully. As you read, take notes. Write down questions about anything you do not understand, and ask them in class or through Oncourse Forums.

Programming (Labs, Homework)

There will be a new programming (“lab”) assignment approximately every two weeks. Programming can be a very time-consuming activity; programming classes usually require more time for homework than other subjects. An average student should allocate at least 6 hours per week for programming activities outside of class, in addition to time for reading the textbook, reviewing notes, and attending class.

Program development is an activity that is difficult to schedule precisely, even for experienced programmers. How much more so for beginners! Students are advised to plan on finishing their programs at least one class day before they are due, because of the likelihood of unexpected problems.

Discussion Forums

There may or may not be some assigned discussion topics (“discussion assignments”) in the Oncourse Discussion Forums. If there are, you will be expected to respond in a timely manner; late posts are worth little or nothing.

Regardless of whether there are discussion assignments, the discussion forum is always open for questions. (But if you have a need for privacy, for example, your question is of a personal nature or involves revealing more than a little of your code, send me an Oncourse Message instead.) If you can answer someone else’s question, please do.

Please check forums regularly for new posts. When you have read a post or thread, you should mark it as read or mark all as read. This will enable Oncourse to give you an accurate count of unread messages, in “Message Center Notifications”; and it will enable the rest of us to know how many people have read our posts.

Exams

The midterm exam will cover chapters 1–7. The final exam will cover the entire course, but will emphasize the last half of the course, chapters 8–13.

These exams will be closed book, closed notes, given in class, and answered by using paper and pencil; the exams are not given through or using a computer. Students in the online section(s)
<table>
<thead>
<tr>
<th>Component</th>
<th>Points</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab assignments (7 to 8 assigned, 50–60 points each)</td>
<td>400</td>
<td>57–51%</td>
</tr>
<tr>
<td>Discussion assignments (0 to 9 assigned, 10 points each)</td>
<td>0–90</td>
<td>0–11%</td>
</tr>
<tr>
<td>Midterm exam</td>
<td>140</td>
<td>20–18%</td>
</tr>
<tr>
<td>Final exam</td>
<td>160</td>
<td>23–20%</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>700–790</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

Table 1: Estimated Point Distribution

<table>
<thead>
<tr>
<th>Percent</th>
<th>Grade</th>
<th>Percent</th>
<th>Grade</th>
<th>Percent</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>96.667–100.000</td>
<td>A+</td>
<td>93.333–96.666</td>
<td>A</td>
<td>90.000–93.332</td>
<td>A−</td>
</tr>
<tr>
<td>86.667–89.999</td>
<td>B+</td>
<td>83.333–86.666</td>
<td>B</td>
<td>80.000–83.332</td>
<td>B−</td>
</tr>
<tr>
<td>76.667–79.999</td>
<td>C+</td>
<td>73.333–76.666</td>
<td>C</td>
<td>70.000–73.332</td>
<td>C−</td>
</tr>
<tr>
<td>66.667–69.999</td>
<td>D+</td>
<td>63.333–66.666</td>
<td>D</td>
<td>60.000–63.332</td>
<td>D−</td>
</tr>
<tr>
<td>0.000–59.999</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>F</td>
</tr>
</tbody>
</table>

Table 2: Grading Scale

will take their exams either at the IU East Testing Center or with a proctor approved by the instructor.

No late or makeup exams will be allowed without prior arrangement and documented and serious reasons. “Prior arrangement” means, if at all possible, contact the instructor before the exam is given. In case of emergencies, the instructor should be notified at least by the end of the day. (So notify your families that they may be obliged to do this for you.)

Activities Summary: Determination of Course Grade

Table 1 shows an estimate of the number of pieces of graded work of each type and how they contribute to the course grade. Table 2 shows the grading scale.

Lab Facilities

Students using a secure shell client may log in remotely to the Linux server, merlin.iue.edu. With VNC you can have a graphical remote login, but this works poorly with a slow Internet connection. **Note: merlin.iue.edu is not a web server, and you cannot simply “open it” in a web browser. You must use software such as secure shell, PuTTY, or VNC to access the merlin server!**

PuTTY and a VNC client should be installed on the Windows computers in Tom Raper Hall 102 and, I believe, in all campus computer labs. PuTTY is the recommended secure shell client for Windows users (see next paragraph). SSVNC is the recommended VNC client for Windows users (see next paragraph). There are many other VNC clients, such as TigerVNC and TightVNC.
However, your VNC session needs to be encrypted through a secure shell tunnel, and SSVNC makes it easiest to do this. If your web browser has a Java plugin, you may also be able to start a VNC session through http://www.iue.edu/it/vnc

Please contact the IT Help Desk (Hayes Hall 169, telephone 765-973-8375) for any problems of lab access including locked doors, login and password. The lab should be open (unlocked) on the same schedule as other campus computer labs: Monday–Thursday 8 a.m.–9 p.m. and Friday 8 a.m.–5 p.m.

Please see the Linux Lab Manual\textsuperscript{3} for further information.

\section*{Notices}

\subsection*{Late Homework}

One characteristic of good software is that it is delivered “on time,” and the same is true of homework, such as labs. Work that is turned in on time may earn full credit. Work turned in late will earn partial credit, up to 90\% for one week late, up to 80\% for two weeks late. \textbf{Unless a student is entitled to a grade of “Incomplete,” no assignment will be accepted after the last day of class before the final exam.}

I do not normally accept assignments that are more than two weeks late; however, in case of extraordinary circumstances, discuss the situation with me (earlier is better) and we may work something out.

\subsection*{Attendance}

IU requires students to attend class.

\textit{Merely not attending does not} withdraw a student from a course; see “Withdrawal” below.

\textbf{Students who fail the course due to non-attendance will receive a grade of FN or FNN; these grades can result in termination and reversal (paying back) of financial aid.}

\subsection*{Attendance in an Online Class}

IU requires students to attend class. What does that mean in an online course? For this course, every participation by a student in a class-related activity that becomes known to the instructor, including the discussion forum, taking a quiz, turning in a lab assignment, and taking an exam, will be considered a day of attendance. If a student does not participate in any activities after a certain date, the student is considered to have stopped attending after that date.

\textit{Merely not attending does not} withdraw a student from a course; see “Withdrawal” below.

\textsuperscript{3}http://mypage.iu.edu/~gdweber/doc/labman/xhtml/index.html
Students who fail the course due to non-attendance will receive a grade of FN or FNN; these grades can result in termination and reversal (paying back) of financial aid.

Lab Attendance

Lab attendance is ordinarily required. However, you may be excused from lab if you have turned in all current assignments, and if there is no quiz or presentation of new material. Check with the instructor to be sure.

Withdrawal

Withdrawal from a course requires a withdrawal form. Normally, withdrawal must take place on or before the “last date for withdrawal with an automatic W”, which this semester is (Mar 15). Withdrawal after that date requires the instructor to determine the grade as W, meaning the student was passing at the time of withdrawal, or WF, meaning the student was failing.

Helping, Cheating, and Academic Honesty

See the IU East Standards of Student Conduct Policy\(^4\) and the Indiana University Code of Student Rights, Responsibilities and Conduct,\(^5\) especially the section “Academic Misconduct.”\(^6\)

Strict academic honesty is expected of all students. The IU code of student conduct provides serious penalties for cheating. All work turned in for credit must be *substantially* the work of the student (or students, if teamwork is authorized) turning it in. Other students needing help, except for simple questions, should be referred to the instructor.

It is part of my responsibility as instructor to help students who are having difficulty with their assignments. Don’t be ashamed or embarrassed to ask me for help! I *want* to help! (By the way: usually, I will try to help a student to think through the solution rather than directly provide an answer.)

Copying another person’s work is cheating, and so is providing the original work to another student for copying. In such cases, *both* students are equally guilty and will be equally punished. Do not share your work with other students or leave it lying around for anyone to pick up. There are usually many different ways to solve a problem; therefore, identical or very similar solutions are *prima facie* evidence of cheating.

\(^4\)http://www.iue.edu/organizations/senate/documents/Policies/studentaffairs/StudentConductPolicy090407.pdf
\(^5\)http://www.iu.edu/~code/
\(^6\)http://www.iu.edu/~code/code/responsibilities/academic/index.shtml
Restrictions on Copying and Distribution of Class Materials

All of the materials that I post that are my original material, including recorded lectures, lecture notes, etc., are materials that you may use freely for the purposes of your study within this course and other academic courses. You may not copy or distribute them, electronically or otherwise, for any other purpose without my permission. Similarly, photographing and audio or video recording of our classes (and posting said pictures/files to YouTube or another site) is not permitted, except as authorized for the purpose of accommodating a documented disability.

Disabilities

Students with disabilities (including physical, mental, sensory, and learning impairment) and wishing to receive auxiliary aids and services (“accommodations”) should fill out a Disability Accommodations Request Form and send it to the Student Support Services Office, early in the semester, and inform their instructor if appropriate. See the Student Support Services site or contact Student Support Services for additional information.

Religious Observances

Students needing an accommodation for a religious observance (for example, rescheduling an exam from a date on which work is prohibited) should communicate their needs to the instructor, in accordance with the IU East “Religious Observances Policy”.

Student Athletes

Student athletes who are unable to attend class(es) because of any IU East athletic events must inform the instructor, during the first week of class, about the conflicting dates. The instructor will inform students if accommodations can be made, the nature of the accommodations, and the accommodations will be written and signed by instructor and student, with a copy available for the coach. For details, see the IU East “Policy for the Approved Absence of Students Participating in Athletic Events”.

7http://www.iue.edu/support/
Contacting the Instructor

**Name:** Gregory D. Weber

**Office location:** HY 255T

**Office hours:** See [http://mypage.iu.edu/~gdweber/contact/schedule.html](http://mypage.iu.edu/~gdweber/contact/schedule.html) for my (possibly updated) normal schedule. Additional hours by appointment.

**Telephone:** (765) 973-8420 (voice); (765) 973-8550 (FAX).

**Messages:** Please use Oncourse Messages instead of email for class-related communications. For other purposes, such as advising, see contact information at [http://mypage.iu.edu/~gdweber/contact/info.html](http://mypage.iu.edu/~gdweber/contact/info.html).

**Other channels:** (by appointment only)
- IU’s “Connect” Adobe Breeze service or Google+ Hangout or Google Talk
- Jabber/XMPP chat: magister.informaticae@jabber.org
- SIP phone: magister.informaticae@ekiga.net

**Personal home page:** [http://mypage.iu.edu/~gdweber/](http://mypage.iu.edu/~gdweber/) (this is *not* the course web site!)
Schedule

Dates of assignments will be announced.

<table>
<thead>
<tr>
<th>Dates</th>
<th>Topics and Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 14–19</td>
<td>Chapter 1. Introduction.</td>
</tr>
<tr>
<td>Jan 21</td>
<td>Martin Luther King Day, no class.</td>
</tr>
<tr>
<td>Jan 22–26</td>
<td>Chapter 2. Data and Expressions.</td>
</tr>
<tr>
<td>Feb 4–9</td>
<td>Chapter 4. Writing Classes.</td>
</tr>
<tr>
<td>Feb 11–16</td>
<td>Chapter 5. Conditionals and Loops.</td>
</tr>
<tr>
<td>Mar 4–9</td>
<td>Midterm review and exam, chapters 1–7.</td>
</tr>
<tr>
<td></td>
<td>Hybrid sections: test in the classroom Mar 6 at lab times.</td>
</tr>
<tr>
<td>Mar 11–16</td>
<td>Chapter 8. Arrays.</td>
</tr>
<tr>
<td>(Mar 15)</td>
<td>Last date to withdraw with an automatic W.</td>
</tr>
<tr>
<td>(March 18–23)</td>
<td>Spring break; no class</td>
</tr>
<tr>
<td>Apr 1–6</td>
<td>Chapter 10. Polymorphism.</td>
</tr>
<tr>
<td>Apr 8–13</td>
<td>Chapter 11. Exceptions and I/O.</td>
</tr>
<tr>
<td>Apr 29</td>
<td>Last day of class; no assignments accepted after this.</td>
</tr>
<tr>
<td>Apr 29–May 4</td>
<td>Final review and exam, all chapters, emphasizing 8–13.</td>
</tr>
<tr>
<td></td>
<td>Hybrid sections: test in the classroom May 1 at lab times.</td>
</tr>
<tr>
<td></td>
<td>Online section: test in the Testing Center, Apr 30–May 2.</td>
</tr>
</tbody>
</table>