CS201, Software Development Methods, Fall 2007
Beginning of Course Memo

Class Website: http://www.cs.virginia.edu/cs201

Instructors:
Prof. Tom Horton  Office: Olsson 228B. Phone: 982-2217  Email: horton [at] cs.virginia.edu
Office hours: MW 11:00 am-12:00 pm; TTh 3:30-4:00 pm; F 11:00-11:30 am
Prof. Mark Sherriff  Office: Olsson 228A. Phone: 982-2688  Email: sherriff [at] cs.virginia.edu
Office hours: Monday-Friday 2:00-3:00 pm

Lab Coordinators / Teaching Assistants: See the class website.

Email:
• cs201@cs.virginia.edu for all questions about assignments, lab exercises, assignment grades, corrections, etc.
• Email instructors for appointments, emergencies, absences, etc.

Prerequisites: CS 101 with grade of C- or higher. (Or equivalent, such as AP or placement exam. Talk to instructor if you have any questions.)

Description from the Undergraduate Record:
A continuation of CS 101, emphasizing modern software development methods. An introduction to the software development life cycle and processes. Topics include requirements analysis, specification, design, implementation, and verification. Emphasizes the role of the individual programmer in large software development projects.

(This will soon be updated to be a better description, as follows.)
A second course in computing and software development, with an emphasis both on modern software development and on principles central to computer science. Topics related to software engineering include the development lifecycle, requirements analysis, specification, design, implementation, and verification. Also addresses core computing principles such as abstraction, encapsulation, information hiding and recursion. More object-oriented programming techniques including inheritance, polymorphism, event-based programming.

Textbooks Required and Readings:

Required text: Modern Software Development Using Java. 2nd edition. (We’ll refer to this as “MSD”.) Authors: Paul Tymann and G. Schneider.

Also: Students may wish to own (or have good access to) a good Java “reference” book. Recommendations:
• A reasonable (not huge) book with good explanation, I recommend: Just Java 2, 6th edition. Author: Peter van der Linden. (About $35.)
• Free! Bruck Eckel’s Thinking in Java, 3/e: http://www.mindview.net/Books/TIJ
A few other readings may be assigned, either through handouts, on the web, or PDFs on-line.

Also required: a “clicker” device: a ResponseCard RF, made by Turning Technologies

In-class Exercises and Student Response System (AKA “Clickers”):
This semester CS201 will use the “clicker” device bundled with your textbook to make lectures more interactive and engaging. During most classes, we will ask you to respond to a number of questions using the RF Response Card. We will use your responses to gauge the overall comprehension of the subject, to clear up any misunderstandings, and to steer the discussion in the direction that most interests students. You will receive 2 points for a correct answer and 1 point for an incorrect answer. Once you have earned 100 points, you will not earn any additional credit. We do not allow students to make up missed questions. Clickers may also be used for other in-class activities that do not contribute to this score.
Grading Information and Criteria:

- **Labs and Homeworks**: 42%
  - See info below on lab activities. Details will be announced in class and on website.
  - Participation: up to a 5% penalty on the overall class grade may be assessed for not attending lab sessions. Also,
- **Tests and Final Exam**: 57%. (Held in class. Dates are subject to change.)
  - Test 1, 13%. Fri., Sept. 28
  - Test 2, 13%. Fri., Oct. 26
  - Test 3, 13%. Mon., Nov. 19 (note: Monday of Thanksgiving week)
  - Final Exam, 18%. Tue., Dec. 11, 9am-noon (probably just a 120-minute exam).
  - Note: we may schedule a Java-oriented exam to be held in a lab session. If so, it will be included in this part of your grade and the weights of the 3 exams will be reduced.
- **In-class Exercises**: 3% (see below)
- **Participation in Lab Sessions**: Up to a 3% penalty on the final grade total. Also, following our Dean’s office recommendation, not filling out the end-of-term class evaluation may result in a penalty.
- **Grading Scale**:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>100</td>
</tr>
<tr>
<td>A</td>
<td>97</td>
</tr>
<tr>
<td>A-</td>
<td>92</td>
</tr>
<tr>
<td>B+</td>
<td>89</td>
</tr>
<tr>
<td>B</td>
<td>86</td>
</tr>
<tr>
<td>B-</td>
<td>82</td>
</tr>
<tr>
<td>C+</td>
<td>79</td>
</tr>
<tr>
<td>C</td>
<td>76</td>
</tr>
<tr>
<td>C-</td>
<td>72</td>
</tr>
<tr>
<td>D+</td>
<td>69</td>
</tr>
<tr>
<td>D</td>
<td>66</td>
</tr>
<tr>
<td>D-</td>
<td>62</td>
</tr>
<tr>
<td>F</td>
<td>59</td>
</tr>
</tbody>
</table>

**Rounding**: Grade averages not falling as integers will be rounded up or down with consideration to class attendance and participation.

Lab Sections and Homeworks:
Each student must be registered in a lab section. A graduate teaching assistant will be the Head TA for each lab section, and he or she will be the point of contact all issues related to the grading of lab-exercises, programming assignments, and lab-participation. For questions on such matters, contact the Head TA (cs201@cs.virginia.edu) and not the instructor. The Head TA will track your homework, test, and overall grade. If you have issues with your Head TA, do not hesitate to contact the course instructor.

Labs will be held weekly in Olsson 001. Students will be expected to attend their scheduled section. A policy and procedure for attending a different section on occasion if necessary will be published on the course web-site. For some group-work labs, you must attend your own section.

Our lab activities will vary. Normally:
- Sometimes there will be a pre-lab activity that you must complete before coming to lab. This may be simply reading, or it may involve some a small bit of coding.
- During lab, you will do some kind of lab exercise. This may be coding on an assignment, doing something in groups, working through an exercise, taking a quiz, etc. You’ll get a participation grade for this each week, and if you attend and make a good effort, you will get full points. If you do not complete the activity, you do not lose points but you must work through this on your own after lab. You are responsible for understanding all that’s covered in the lab.
- Programming homeworks will normally be done outside of scheduled lab meetings. Usually you’ll be allowed to work in pairs, if you agree to follow certain rules. Sometimes lab-exercises will be designed to get you started on your assignment.
- One homework will be a larger course-project. You’ll be required to work in teams of three or four.
- For grading purposes, a few programming assignments will have a weight of “5”, while most will have a standard weight of “10”, and projects might be weighted as much as “20”. You can anticipate about 5-6 assignments to be given during the term.
Course Objectives:
Upon successful completion of this course, students will:

1. Comprehend more advanced principles of object-oriented programming and how a programming language supports these, and apply these by developing larger and more complex programs than in their first programming course. (Topics include polymorphism, inheritance, collection classes, generics, etc.)
2. Comprehend and apply principles of design at the class and object level. These principles include abstraction, encapsulation, and information hiding. This also includes the ability to define and evaluate class interfaces to solve specified design problems, as well as the ability to understand, apply, and evaluate the use of reusable components to solve such problems.
3. Comprehend and analyze problems and programming issues such as dynamic memory management, indirect object references, and recursion. Also, be able to apply this knowledge by implementing software that includes these features.
4. Apply knowledge of software development practice to effectively use strategies, tools and environments such as interactive development environments, debuggers, testing frameworks, etc.
5. Comprehend important basic concepts of software engineering and the development of large software systems, including the software lifecycle, requirements, design, and software quality. In their development activities, students will be able to apply basic unit testing and carry out a software inspection.
6. Comprehend the basic principles of the architecture of larger software systems, in particular object-oriented frameworks. Students will be able to apply this knowledge by developing a GUI using a framework.

Expectations:
We will try to be very clear about what we expect of each other in this course. For each major section of the course, we will post a list of topics that we expect you to understand along with a set of problems that represent what we would expect you to be able to do. At certain points in the semester, we may announce what we expect you to do before coming to lecture (read something, work a problem, etc.) and the class meeting will reflect this. Several times in the course, we will ask you to read many pages in the textbook as background before lecture so that we don't have to lecture on basic material or things that are review from earlier courses.

You have the right to expect us and the TAs to be fair and efficient in managing the course, and when we're not please let us know. (Perhaps through anonymous email feedback.) You have the right to know how you're doing in the course (as far as we can determine) as we progress through the term. You should be able to expect us to not waste too much time on material you've already had before (and we may thus reasonably expect you to review such material outside of lecture).

Honor Policy:
The School of Engineering and Applied Science relies upon and cherishes its community of trust. We firmly endorse, uphold, and embrace the University's Honor principle that students will not lie, cheat, or steal, nor shall they tolerate those who do. We recognize that even one honor infraction can destroy an exemplary reputation that has taken years to build. Acting in a manner consistent with the principles of honor will benefit every member of the community both while enrolled in the Engineering School and in the future.

Students are expected to be familiar with the university honor code, including the section on academic fraud (http://www.student.virginia.edu/~honor/proc/fraud.html).

Each assignment will describe allowed collaborations, and deviations from these will be considered Honor violations. If you have questions on what is allowable, ask! Unless otherwise noted, exams and individual assignments will be considered pledged that you have neither given nor received help. (Among other things, this means that you are not allowed to describe problems on an exam to a student who has not taken it yet. You are not allowed to show exam papers to another student or view another student's exam papers while working on an exam.) Send, receiving or otherwise copying electronic files that are part of course assignments are not allowed collaborations (except for those explicitly allowed in assignment instructions).

Assignments or exams where honor infractions or prohibited collaborations occur will receive a zero grade for that entire assignment or exam. Such infractions will also be submitted to the Honor Committee if that is appropriate. Students who have had prohibited collaborations may not be allowed to work with partners on remaining homeworks.

LNEC and Other Special Circumstances:
If you have been identified as an LNEC student, please let the Center know you are taking this class. If you suspect you should be an LNEC student, please schedule an appointment with them for an evaluation. I happily and discretely provide the recommended accommodations for those students identified by the LNEC. Please contact me one week before an exam so we can make accommodations. Website: http://www.virginia.edu/studenthealth/lnec.html

If you have other special circumstances (athletics, other university-related activities, etc.) please contact your instructor and/or Head TA as soon as you know these may affect you in class.