<table>
<thead>
<tr>
<th><strong>Location</strong></th>
<th>110 West Village H</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Time</strong></td>
<td>Mondays, Wednesdays, Thursdays 10:30AM–11:35AM</td>
</tr>
<tr>
<td><strong>Web site</strong></td>
<td><a href="http://www.ccs.neu.edu/~amislove/teaching/cs3600/spring12">http://www.ccs.neu.edu/~amislove/teaching/cs3600/spring12</a></td>
</tr>
<tr>
<td><strong>Forum</strong></td>
<td><a href="http://piazza.com/northeastern/spring2012/cs3600">http://piazza.com/northeastern/spring2012/cs3600</a></td>
</tr>
<tr>
<td><strong>Instructor</strong></td>
<td>Prof. Alan Mislove</td>
</tr>
<tr>
<td><strong>Contact</strong></td>
<td><a href="mailto:amislove@ccs.neu.edu">amislove@ccs.neu.edu</a> (put “[cs3600]” in the subject line)</td>
</tr>
<tr>
<td><strong>Office hours</strong></td>
<td>Mondays, 3:00pm–5:00pm, 250 West Village H</td>
</tr>
<tr>
<td><strong>Teaching assistant</strong></td>
<td>Yabing Liu</td>
</tr>
<tr>
<td><strong>Contact</strong></td>
<td><a href="mailto:cs3600sp12-staff@ccs.neu.edu">cs3600sp12-staff@ccs.neu.edu</a> (put “[cs3600]” in the subject line)</td>
</tr>
<tr>
<td><strong>Office hours</strong></td>
<td>Thursdays, 1:00pm–3:00pm, 266 West Village H</td>
</tr>
<tr>
<td><strong>Teaching assistant</strong></td>
<td>Ashutosh Warikoo</td>
</tr>
<tr>
<td><strong>Contact</strong></td>
<td><a href="mailto:cs3600sp12-staff@ccs.neu.edu">cs3600sp12-staff@ccs.neu.edu</a> (put “[cs3600]” in the subject line)</td>
</tr>
<tr>
<td><strong>Lab hours</strong></td>
<td>Wednesdays, 4:00pm–6:00pm, 102 West Village H</td>
</tr>
</tbody>
</table>

**DESCRIPTION** (from the Registrar)
Introduces the basic concepts underlying computer operating systems and computer networks and provides hands-on experience with their implementation. Covers the basic structure of an operating system: application interfaces, processes, threads, synchronization, interprocess communication, processor allocation, deadlocks, memory management, file systems, and input/output control. Also introduces network architectures, network topologies, network protocols, layering concepts (for example, iso/osi, tcp/ip reference models), communication paradigms (point-to-point vs. multicast/broadcast, connectionless vs. connection oriented), and networking API’s (sockets). Uses examples from many real operating systems and networks (UNIX, MS-DOS, Windows, TCP/IP, Ethernet, ATM, and token rings) to reinforce concepts.

**DESCRIPTION** (from me)
We’ll be studying how computers and the Internet really work. The course will be hands-on and participation is strongly encouraged. This course will not be easy.

**LOGISTICS**
The class will thrice per week for 65-minute sessions, plus a two-hour final exam. The midterm exam will be administered during a regular class session. I am not going to require you to attend class; I am not your mother. However, special requests will be treated with particular suspicion if I do not recognize you from lecture.

**PREREQUISITES**
The official prerequisite for this course is cs2600. This course will be project-centric, and all students will complete in projects in groups of two (or possibly three, if necessary). Thus, to succeed in this course, you must be able to work in a group. I will allow you to form your own groups, and the course staff will serve as a matching service if necessary. As you are free to choose
your partner(s), I will not be sympathetic to complaints at the end of the semester about how your group-mates did not do any work.

It is also highly recommended that you become familiar with using a debugger, as this will greatly aid you in completing the projects. At a high level, you should be motivated, eager to learn, willing to work hard, and make up, on your own, any prerequisite deficiencies you may have.

**GOALS**

By the end of this course, I expect you to understand the following concepts:

- Operating system basics — context switching and program loading
- Virtual memory at the hardware and operating system level
- Basic operating system security mechanisms
- Basic networking protocols, including protocol layering
- Networking principles including routing, addressing, congestion control
- The benefits and limitations of the current Internet and its service model
- The causes of network congestion, and the basic methods for alleviating congestion

**TEXTBOOK**

The textbooks for the course are


Yes, there are two books. I'm sorry. The course is Systems and Networks. If its any consolation, you'll use the operating systems book in cs5600 and the networking book in cs4700/cs5700. For the C bootcamp (see below), you are also encouraged (but not required) to buy the book


**C BOOTCAMP**

Part of the goal of this class is to teach you some systems programming skills. To do so, most of the projects will be required to be completed in C. As many of you are not familiar with C, we will spend the first four lectures doing a C “bootcamp.” Each lecture will cover an aspect of C, and will be followed by short homework problems that make sure you understand what we did in lecture.

*The C bootcamp homeworks are due at the beginning of the next lecture.*

**HOMEWORK ASSIGNMENTS**

This course will have ten homework assignments reviewing concepts in class. Homework assignments are to be done by each student individually.

*Homework assignments are due at the beginning of lecture on the specified date.*
PROGRAMMING PROJECTS
The goal of this course is to teach both the fundamentals of systems and networks, as well as how to write programs which use both. As such, there will be four programming projects throughout the semester.

You will form groups of two people to do the programming projects. To collaborate effectively, you should both be involved in all of the major design decisions. You should also determine a partitioning of responsibilities so that you can both work effectively in parallel. For example, one might be responsible for generating all the test code while the other is responsible for the main code. You may switch groups between programming projects.

Important: You alone are responsible for finding a partner. The class forum (newsgroup) located on Blackboard is a particularly good resource for this. Breaks during lecture are also a good time to look for partners.

For the projects in this course, we will use Java to complete the programming projects. The submitted code must compile and run without any special requirements on the CCIS Linux machines. If you have any questions about the use of Java for the programming assignments, please email the instructor. If you have any questions about the support code, Java error messages, and so forth, please post to the class forum.

The TA has been instructed to grade in part on design and implementation style and to be increasingly strict about this as the semester proceeds. In other words, it is not enough to get a working solution; you must implement the solution in a clean way that would simplify making further enhancements. It will benefit you in the long run to work on your software engineering skills.

Programming projects are due at 11:59:59pm on the specified date.

FORUM
We will provide a Web forum (the URL is provided at the beginning of this document) that can be used by students to ask questions and exchange wisdom while completing the homeworks and projects in this course. Please use the forum to post questions and answers that may be useful to others. Specifically, questions of the form “How do I link foo?, “Does such-and-such option in the compiler work for you?, or “What is the precise interpretation of homework question III, part b? should be posted on the forum first. If you mail me (or other course staff) these questions, we might not be able to answer them in time.

EXAMS
There will be one midterm and one final. All exams will be closed book and closed notes, and computers are not allowed nor is any access to the Internet via any device. The exams will cover material from lectures, readings, and the projects. They will cover the material discussed during the first and second halves of the class, respectively (i.e., they are not cumulative).

GRADING
The breakdown of the grades in this course is

40% Projects (4 at 10% each)
30% Exams (15% midterm and 15% final)
20% Homeworks (10 at 2% each)

1 If necessary, one group of three will be allowed.
5% C Bootcamp Projects
5% Participation

Each project and homework will include a breakdown and description of how it will be graded. Any requests for grade changes or regrading must be made within 7 days of when the work was returned. To ask for a regrade, attach to your work a page that specifies (a) the problem or problems you want to be regraded, and (b) for each of these problems, why do you think the problem was misgraded.

**LATE POLICY**
Written homework assignments have strict deadlines. Homework handed in late will be marked off 20% per day. Homework more than 2 days late will not be accepted. Late C bootcamp homeworks will not be accepted. Extensions will not be granted.

For programming projects, we will use flexible slip dates. Each student is given an automatic extension of 4 calendar days for the semester. You can use the extension on any project during the semester in increments of a day. For instance, you can hand in one project 4 days late, or one project 2 days late and two projects 1 day late. The slip time will be deducted from each group member’s remaining slip time. This should let you schedule due dates around the due dates for other courses. After you have used up your slip time, any project handed in late will be marked off 20% per day. Projects more than 2 days late will not be accepted. Extensions will not be granted.

**ON CHEATING**
It’s ok to ask someone about the concepts, algorithms, or approaches needed to do the assignments. We encourage you to do so; both giving and taking advice will help you to learn. However, what you turn in must be your own, or for projects, your group's own work; looking at or copying other people’s code, solution sets, or from any other sources is strictly prohibited. In particular, looking at other solutions (e.g., someone else's solution to a similar project) is a direct violation. The project assignments must be entirely the work of the students turning them in. If you have any questions about using a particular resource, ask the course staff first.

All students are subject to the Northeastern University Academic Integrity Policy, available at [http://www.northeastern.edu/osccr/academichonesty.html](http://www.northeastern.edu/osccr/academichonesty.html). All cases of suspected plagiarism or other academic dishonesty will be referred to the Office of Student Conduct and Conflict Resolution (osccr).

**ADVICE**
As the course is difficult, students are recommended to start early on homeworks and projects. The projects require substantial design, implementation, and testing effort, especially for students who are unfamiliar with network and systems programming. Students are encouraged to drop by the instructor’s or teaching assistant’s office hours (or set up a separate appointment via email) if they run into problems while completing the assignments.

Finally, computer use during class is allowed but expected to be for class purposes (e.g. note-taking, reference) only. Please avoid non-class-related computer use so that I do not have to revisit this policy.

---

2 A “day” refers to 24 hours. Thus, a homework turned in 28 hours late will count as two days late.

3 Note on slip days: Slip days can only be used if all group members have at least one remaining slip day. If not, the 20% off per day policy stated above will be used for all group members.