ECE 199JL: Introduction to Computer Engineering **Fall 2012**

Course Overview

Instructors: Douglas L. Jones (dl-jones@illinois.edu)
113 Coordinated Sciences Laboratory
Office hours: TBD

Steven S. Lumetta (lumetta@illinois.edu)
209 Coordinated Sciences Laboratory
Th 1-3 upstairs at miaZa's
(west of Wright on south side of Green)

Discussion Leader: Geoffrey Herman (glherman@illinois.edu)
Teaching Assistants:
Sai Ma (saima1@illinois.edu)
Seyed Ahmadyan (ahmadya2@illinois.edu)
Office hours for will be announced on the web site.

Together with ECE199KL, this course gives an introduction to the design and programming of computing systems. We start the course by motivating our objectives and connecting them with students future ECE studies and career paths. The philosophy of our approach is quite different than the typical introduction to programming course: after a brief illustration of our goals and objectives with a quick introduction to C, we approach programming from the bottom upwards. In particular, we begin by describing the architecture of a computer, including logic gates, datapaths, registers, and memories. Throughout the course, we will make connections between hardware and software and explore the engineering tradeoffs in using each to develop computing systems.

The second course in the sequence (ECE199KL) will focus on C programming, where each new C concept will be related to the fundamental concepts described in ECE199JL. We will cover basic programming concepts, functions, arrays, pointers, I/O, recursion, simple data structures, and concepts in object-oriented programming. A bottom-up understanding of computing systems has proven more successful in helping students to understand advanced concepts in computing that follow in the ECE curriculum.

This course has no prerequisites.

Objectives of the course:

After completing this course, students should

- understand the role and importance of abstraction in computing systems,
- recognize and be able to make use of standard digital components in designing simple combinational logic,
- be able to design and implement a simple finite state machine (FSM) as a clock-synchronous sequential circuit using digital logic,
- be able to design and implement a simple FSM using assembly language, and
- be able to write simple programs in both C and assembly language.

Since this is a freshman course, we also have explicit objectives for providing students with an understanding of the profession. In particular, students who have completed this course should

- understand the expectations of the engineering discipline in terms of effort, quality, and objectivity,
- recognize that self-motivation and lifelong learning are necessary to success in engineering,
- be able to articulate the importance of understanding tradeoffs, and
- be able to recognize and identify basic design tradeoffs.

Textbook:

Yale N. Patt and Sanjay J. Patel, *Introduction to Computing Systems: from bits and gates to C and beyond*, 2nd Edition, McGraw-Hill, 2003.

This course is mostly based on the textbook listed above, but we will provide different examples, additional reading materials, and a somewhat different viewpoint in the lectures.

What you should expect:

There will be **weekly homework assignments**, most of which will include some computer work. They will be posted on the webpage for the course and due in class on Fridays.

There will be **five laboratory assignments**. The first will introduce you to the EWS laboratory environment and the use of command-line interfaces that you will use for your homework and programming assignments. The next two will give you hands-on experience with digital logic implementation. The last two will be programming assignments and will involve assembly-level programming using the LC-3 simulator. The second assignment will be fairly substantial. Some of the lab assignments will have intermediate checkpoints that you will turn in.

There will be **three midterm exams** and **one final exam**. The midterm exams will be from 8 to 10 p.m. on the evenings of Tuesday 18 September, Tuesday 16 October, and Tuesday 13 November. The final exam is on Tuesday 18 December from 1:30 to 4:30 p.m. Any conflict that you have with any of the exams **must be reported** to either Professor Jones or Professor Lumetta **at least one week before the exam**, but please report such conflicts as early as possible. Conflict exams for the midterms will be held just before the regular time, from 6 to 8 p.m. on the same day.

Grading mechanics:

Homeworks:	15%
Lab Assignments:	15%
Exams:	15% for each of three midterms; 25% for the final

Website and web board:

The website (http://courses.engr.illinois.edu/ece199/) will contain important announcements, lecture notes, handouts, and other material helpful for succeeding in this course.

The ECE199JL "web board" is available through the web boards link at http://my.ece.illinois.edu. Note that this is NOT the same as the campus' web board project. The web board serves as a forum for students to post and answer questions, discuss issues, warn of pitfalls, etc. You should read the board at least once a day. The TAs and I will read and post to the web board to focus discussions and to provide more definitive answers to posted questions.

Final thoughts:

We want to meet with each of you early in the semester, so we have integrated a short survey and meeting as part of your first laboratory assignment.

Challenge assumptions: Computer Science and Engineering deals with man-made artifacts, and you may be able to invent better ways to make them. Innovation requires that someone challenge the current way of doing things.

You are encouraged to study in groups, and to come to office hours in groups. Studying in groups usually will result in all of you understanding the material better. You, working with other members of your study group, can often unravel concepts to the benefit of all members of the group much better than one can person can, working alone.

Although we encourage you to study together, all work products of this course (homeworks, programming assignments, examinations) must be your own individual work. Do not, for example, exchange code with others. We will use code comparison tools to identify violations. If you cheat, you violate the soul of the University, which we take very seriously, and will not compromise. First offense will, in the least, result in a 0 on the assignment or exam. The policy for the course is based on Article 1.4 of the *Student Code* (available at http://www.admin.uiuc.edu/policy/code/).