Instructional Objectives for ECE 390
Computer Engineering II

Course Goals:

The primary goal of this course is to understand the relationship between the functionality of programming languages such as Java and C++ and the execution of machine-level instructions of a computer. Throughout the course, students develop, analyze, debug, and optimize complex assembly language programs. They learn to use the computer for real-time data acquisition and on-line control of devices.

In Machine Problem 0, the students will be able to do the following:

Locate and modify the values of variables in segmented memory (a)
Analyze the execution of a program from within a code debugger (b, k)
Use library routines to perform common input/output operations (k)

In Homework 1, the students will be able to do the following:

Determine the results of execution of individual machine-level instructions (a)
Analyze the execution of a machine-level program (a)

In Machine Problem 1, the students will be able to do the following:

Implement conditional constructs, case constructs, and loops in assembly language (c)

In Homework 2, the students will be able to do the following:

Determine effective addresses for different addressing modes (a)
Calculate and analyze results of integer computations in modular arithmetic (a)
Use code review to debug a program (e)

In Machine Problem 2, the students will be able to do the following:

Use subroutines to structure programs (c)
Use different methods to transmit arguments to and from subroutines (c)
Use a debugging tool to identify and repair code defects (b, e)

In Homework 3, the students will be able to do the following:

Analyze the behavior of a recursive program (a)

In Machine Problem 3, the students will be able to do the following:

Develop a recursive program to solve a practical problem (c)

In Homework 4, the students will be able to do the following:

Analyze implementations of data structures such as tables and queues (a)
Explain the reasons for implementing critical sections (a)

In Machine Problem 4, the students will be able to do the following:

Design an interrupt service routine to respond to external input from the keyboard, timer, or I/O port (c)

Write a program that uses text-mode video (c)

In Homework 5, the students will be able to do the following:

Calculate the worst-case completion times for tasks in real-time systems under preemptive and nonpreemptive scheduling policies (a)

Determine encodings of floating-point numbers in IEEE-754 format (a)

In Machine Problem 5, the students will be able to do the following:

Design a program to enable computers to communicate asynchronously over a serial port or a local network with interrupt-driven I/O (c)

Benchmark subroutines to characterize the performance of a program (b)

Optimize code to improve the performance of a system (c)

Work effectively in teams, using code inspection (d)

In the Final Project, the students will be able to do the following:

Develop an original application that uses real-time data acquisition and graphics, such as a video game (e)

Find information on techniques needed for the project but not covered in class (i)

Design and develop a large, complex program through modular design, using appropriate data structures (c)

Participate effectively as a member of a team (d)

Document and justify design decisions, specify the program structure, and describe the program operation in writing (g)

On the Final Examination, the students will be able to do the following:

In addition to all objectives listed above, identify ethical issues in the use of computers (f)

For more information about ECE 291, see the Web site

http://www.ece.uiuc.edu/~ece291/

and the following journal article:

M. C. Loui, The case for assembly language programming, *IEEE Transactions on Education*, vol. 31, no. 3, pp. 160-164, August 1988.

Revised Spring 2003