EECS 111 SYSTEM SOFTWARE
(Required for CpE)

Catalog Data: **EECS 111 System Software (Credit Units: 4)** Multiprogramming, interrupt, processes, kernel, parallelism, critical sections, deadlocks, communication, multiprocessing, multilevel memory management, binding, name management, file systems, protection, resource allocation, scheduling. Experience with concurrent programming, synchronization mechanisms, interprocess communication. Prerequisites: EECS112; ICS 23 or EECS114. Only one course from EECS111 and CSE104/Computer Science 143A may be taken for credit. Formerly ECE142. (Design units: 2)

Textbook: Silberschatz, Galvin, and Gagne, *Operating System Concepts,* 7th edition, John Wiley and Sons, Inc. -or-
Deitel, Deitel, and Choffnes, *Operating Systems,* 3rd edition, Pearson

References: Other references witll be available from the E3 website or the Engineering Copy Center

Coordinator: Steven Jenks

Course Objectives:
Introduce the concepts of multiprogramming operating systems, concurrent programs, and processes.
Introduce the techniques for scheduling CPUs and other resources for execution of multiple processes and threads.
Introduce the basic principles and techniques for concurrent programming.
Introduce the techniques for logical organization and management of memories to support concurrent processes and threads.

Course Outcomes:
Students will:
Structure concurrent programs composed of processes and threads.
Describe basic CPU scheduling techniques.
Describe the principles and techniques for designing and analyzing concurrent processes capable of correct synchronization among themselves.
Describe the principles and techniques for designing and analyzing concurrent processes capable of avoiding or recovering from deadlocks.
Describe the principles and techniques for designing and analyzing memory management mechanisms including virtual memory.

Prerequisites By Topic:
Understanding of:
Building blocks and organization of digital computers.
Arithmetic, control, and memory units, input/out devices and interfaces.
Microprogramming and microprocessors.
Fundamental data structures and related algorithms and their implementation and mathematical analysis techniques.

Lecture Topics:
Course overview
Processes

Threads
CPU Scheduling
Process Synchronization
Deadlocks
Memory Management
Virtual Memory
File Systems
I/O Systems
Mass-Storage Structure

Class Schedule: Meets for 3 hours of lecture and 1 hour of discussion each week for 10 weeks.

Computer Usage: Personal computers or workstations running widely used operating systems such as Windows, Linux, etc.

Laboratory Projects: 3 – 4 project assignments are given in this course.

Professional Component: Contributes toward the Computer Engineering Topics Courses and Major Design experience.

Relationship to Program Outcomes: This course relates to Program Outcomes a, b, c, d, e, k, l, m, and n as stated at:
http://undergraduate.eng.uci.edu/degreeprograms/computer/mission

Design Content Description

Approach: Concurrent programming activities are essentially design activities. Lectures emphasize relevant design principles and applicable measures and constraints. Laboratory projects provide hands-on design experiences to the students.
Lectures: 60%
Laboratory Portion: 40%

Grading Criteria:

Homework:	10%
Lab assignments:	30%
Midterm exam:	20%
Final exam:	40%
	100%

Estimated ABET Category Content:

Mathematics and Basic Science: 0 credit units or 0%
Engineering Science: 2 credit units or 50%
Engineering Design: 2 credit units or 50%

Prepared by: Steven Jenks **Date:** July 2007

CEP Approved: Fall 2004