

Fall 2009

# CEG 434/634-01: Concurrent Software Design

Douglas J. Kelly

*Wright State University - Main Campus*

Follow this and additional works at: [https://corescholar.libraries.wright.edu/cecs\\_syllabi](https://corescholar.libraries.wright.edu/cecs_syllabi)



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

---

## Repository Citation

Kelly, D. J. (2009). CEG 434/634-01: Concurrent Software Design. .  
[https://corescholar.libraries.wright.edu/cecs\\_syllabi/929](https://corescholar.libraries.wright.edu/cecs_syllabi/929)

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact [corescholar@www.libraries.wright.edu](mailto:corescholar@www.libraries.wright.edu), [library-corescholar@wright.edu](mailto:library-corescholar@wright.edu).

CEG 434/634  
**Concurrent Software Design**

## Syllabus

Fall Quarter, 2009

**Course Description:** This course provides an introduction to concurrent program design in the UNIX environment. Classical problems of synchronization, concurrency, and their solutions are examined through course projects, homework, and readings on operating system design.

**Time/Place:** TR 2000 – 2115 (8-9:15PM), Russ Engineering 154

**Professor Info:** Name: Dr. Douglas J. Kelly  
Office: Air Force Research Laboratory (AFRL), WPAFB  
Office Hours: TR 1900-2000, 2130-2200 or by appointment  
Phone(s): 937-656-4391 (w) daytime 0800-1630  
502-542-1646 (c) other-time  
Email: [douglas.kelly@wright.edu](mailto:douglas.kelly@wright.edu)  
Website: [www.cs.wright.edu/~dkelly](http://www.cs.wright.edu/~dkelly)

**GTA:** None.

**Prerequisite:** CS400, CEG433/633, Operating Systems.  
Expected background: discrete mathematics, data structures, C or C++ programming experience in UNIX.

**Text Books:** *Required:* Operating System Concepts, 6<sup>th</sup>, 7<sup>th</sup> or 8<sup>th</sup> Ed., Silberschatz and Galvin, Addison-Wesley, 2002.  
*Required:* Unix Systems Programming: Communication, Concurrency and Threads, Robbins and Robbins, Prentice Hall, 2003.  
*Reference:* Inter-process Communications in Linux: The Nooks and Crannies, John S. Gray, Prentice Hall, 2003.

**Grading:**

Programming assignment*	30%
Homework**	10%
Midterm Exam	30%
Final	30%

\*Programming assignments are an individual effort.

\*\*You may work with others on homework, but you must turn in your own individual work.

**Cheating:** Any form of cheating such as copied homework or programming code will result in a grade of zero for all parties and be reported to the Office of Judicial Affairs. Also, 10% per day for late unexcused work.

## Lectures:

The following tentative schedule defines in greater details what material is covered in the course and when it is covered.

Week	Reading	Contents
1	Robbins Ch. 1 Silberschatz Ch. 1, 3	Introduction, review of process management, process scheduling
2	Robbins Ch. 2, 3, 6 Silberschatz Ch. 5	Scheduling review, UNIX review. Basic UNIX inter-process communication
3	Robbins Ch. 8 Robbins Ch. 18	Asynchronous events – UNIX signals Client server computing
4	Robbins Ch. 18,20 Gray Ch. 10	Inter-process communication with sockets User Datagram Protocol (UDP)
5	Silberschatz Ch. 4	Threads In-class Midterm Exam (Thursday, Oct 8 <sup>th</sup> )
6	Robbins Ch. 12 Silberschatz Ch. 6	POSIX threads Process synchronization
7	Silberschatz Ch. 6 Robbins Ch.13,14 Siberschatz Ch.7	Process synchronization Deadlocks
8	Silberschatz Ch. 18	Distributed Coordination
9	Siberschatz Ch.19	Real-Time Systems
10	Siberschatz Ch.20	Multimedia Systems
11	Tues., Nov 17 <sup>th</sup> , 2009 8 P.M. – 10 P.M.	Final exam