Graduate Operating Systems (COP 6611)
Spring 2008

General Information
Class meetings: TR 3:30-4:45 in ENB 113
Professor: Adriana (Anda) Iamnitchi (anda@cse.usf.edu)
Office hours: Wednesdays 2pm-4pm or by appointment (ENB 334)
Course webpage: http://www.cse.usf.edu/~anda/cop6611/

Teaching Assistant: Alex Lagor (alagor@cse.usf.edu)
Office: TBD
Office hours: TBD

Textbook: *Operating Systems Concepts, Silberschatz, Galvin & Gagne, 7th Edition*

1. Course Overview
This course will cover an exciting range of materials from the broad field of operating systems, including basic operating system structure, memory management, file systems and storage, distributed systems, virtual machines, and security. We will examine influential historical systems and important current efforts, extracting lessons both on how to build systems as well as how to evaluate them.

The textbook material will be supplemented with research papers that are made available on the class web page.

2. Course Requirements
The work of this course consists of:
- Reading, analyzing, summarizing, and discussing in class the research papers assigned.
- Five small programming projects.
- A midterm and a final exam (in-class).
- Final project.

2.1. Paper Summaries (15%)
In addition to the material covered in the textbook, we will read and discuss 18-20 research papers. Some of these papers form the basis of the current state-of-art in operating system, while others are hot topics or current trends. These papers will provide both depth and breadth in the subject and will be a gentle introduction in systems research.

You are expected to submit on H2O concise paper summaries (about 1/2 page) before 10am on the Thursday of the corresponding week. These summaries should touch the following and should provide you with nice notes for when studying for exams.

1. **Summary**: 1-2 sentences that give the essence of the paper. Please focus on the solution proposed rather than only on the problem. For example, “This paper describes the design of an operating system” should rather be “This paper describes a multiprogramming operating system with a layered design, where layers are responsible for disjoint sets of functionalities”.

2. **Problem**: a description of the problem the paper attempts to solve. Make sure you identify the assumptions stated in the problem description.
3. **Contributions**: the main ideas of the solution proposed.
4. **Limitations**: what are the limitations of the paper? Examples of questions you might try to answer here are: Are there important scenarios not discussed in paper in which the solution does not perform efficiently? Are there assumptions that later research disproved? Some limitations may be stated by the authors. Go beyond what is stated and make sure you deeply understand the material.
5. **Discussion**: any questions you might want to ask the authors or see discussed in class.

2.2. Small Programming Assignments (25%)

There will be 5 small programming assignments related to the textbook material (each worth 5%):

1. Fibonacci with fork + Fibonacci with threads + comparison of overheads in each case. Small write-up with overhead comparison. On Linux. [due end of 2nd week]
2. Readers/writers synchronization problem. [due end of 4th week]
3. File system performance measurement. [due end of 7th week]
4. Adding a system call to the Linux kernel (page 74 textbook). [due end of 10th week]
5. Socket programming. [due end of 13th week]

All assignments are individual, except for Assignment 4 that can be solved in teams of 2-3 students.

2.3. Final project (25%)

The final project will require you to design, carry out and communicate the results on an operating systems project. The final projects are intended to lead to publication-quality results.

Projects are carried out in teams of two or three people. I will suggest a set of topics, but you will be required to develop your own proposal either from the topics listed or from your own ideas.

There are two grade components for the project: a status report (10%) and a final report (15%). The project proposals (due in the 3rd week) will not be graded. Project groups will meet with me after project proposals and status reports to discuss future directions.

I would like to make these projects reports available to students who take this class in the future. If your group prefers not to do this, please email me.

2.4. Exams (35%)

Two in-class exams will make the rest of the grade: 15% for midterm and 20% for final (comprehensive). The exams will test understanding of the class material (covered in textbook, reading list and class lectures). Samples of exam questions will be provided later in the semester.

3. Lecture Schedule

Week	Topics	Reading
1	Overview	Ch. 1, 2, 23, [1,2]
2	Process Management. Threads. Scheduling	Ch. 3, 4, 5, [3]
3	Process Synchronization. Deadlocks	Ch. 6, 7, [4]
4	Memory Management	Ch. 8, 9, [5]
5	File Systems	Ch. 10, 11, [6,7]
6	Distributed File Systems	Ch. 16, 17, [8,9]
7	Examples of File Systems. File-sharing.	[10]
8	Mass Storage Structures. Network-attached Storage. Midterm exam (Thursday)	Ch. 12, [11]
9	Distributed Storage	[12]
10	Protection and Security	Ch. 14, 15, [13]
11	Security Issues on the Internet	[14]
12	Distributed Systems. Distributed Coordination	Ch. 18, [15]
13	More on Distributed Systems.	[16]
14	Virtual Machines	[17]
15	Trends in OS Design	[18]

Reading List:

1. E. W. Dijkstra. The Structure of the "THE" Multiprogramming System. Communications of the ACM 11(5), 1968.
2. D. M. Ritchie and K. Thompson. The UNIX Timesharing System. Communications of the ACM 17(7), 1974.
3. C. Waldspurger and W. Weihl. Lottery Scheduling: Flexible Proportional-Share Resource Management in Proceedings of 1st USENIX Symposium on Operating System Design and Implementation, 1994.
4. P. J. Courtois, F. Heymans, D. L. Parnas. Concurrent control with “readers” and “writers”, Communications of the ACM, 14(10), 1971
5. C. Waldspurger. Memory Resource Management in VMware ESX Server in Proceedings of the 5th Symposium on Operating Systems Design and Implementation, 2002
6. M. Rosenblum and J. K. Ousterhout. The Design and Implementation of a Log-Structured File System. ACM Trans. on Computer Systems 10(1), 1992, pp. 26-52.
7. W. Vogels. File system usage in Windows NT, Proceedings of the 17th ACM Symposium on Operating Systems Principles, 1999.
8. J. Howard et. al. Scale and Performance in a Distributed File System, ACM Transactions on Computer Systems, 6(1) , 1988.
9. E. Nightingale, P. Chen, J. Flinn. Speculative Execution in a Distributed File System, in Proceedings of the 20th ACM Symposium on Operating Systems Principles, 2005.
10. S. Ghemawat, H. Gobioff and S-T Leung. The Google File System, in Proceedings of the 19th ACM Symposium on Operating Systems Principles, 2003
11. D. A. Patterson, G. Gibson, and R. H. Katz. A Case for Redundant Arrays of Inexpensive Disks (RAID) Proceedings of the 1988 ACM SIGMOD Conference on Management of Data, 1988.

12. Gibson, G., Nagle, D., Amiri, K., Chang, F., Feinberg, E., Gobioff, H., Lee, C., Ozceri, B., Riedel, E., Rochberg, D. and Zelenka, J. File Server Scaling with Network-Attached Secure Disks in Proceedings of Proc. of the ACM International Conference on Measurement and Modeling of Computer Systems (Sigmetrics), 1997.
13. — *(time to catch up breath)*.
14. HotBots '07 conference summaries from *;login:*
15. L. Lamport, Time, Clocks, and the Ordering of Events in a Distributed System, *Communications of the ACM*, July 1978, pages 558-564.
16. K. M. Chandy and L. Lamport, Distributed Snapshots: Determining Global States of Distributed Systems, *ACM Transactions on Computer Systems*, February 1985, pages 63-75.
17. P. Barham, B. Dragovic, et al. Xen and the Art of Virtualization. In the Proceedings of the ACM Symposium on Operating Systems Principles (SOSP), 2003
18. J. Hill et al. System architecture directions for networked sensors**.** SIGPLAN Notices, 35(11) 2000.

4. Deadlines and Late Work

Paper summaries must be submitted by 10am of the Thursday they are due. No late submissions will be accepted.

All programming assignments and project components deadlines are 11:59pm on the day they are due. Late assignment submission is not accepted (will result in a 0). Late project submissions will incur a 10% penalty per late day (weekend days count as "days").

5. Collaboration and Cheating

Collaborations are strongly encouraged on reading list and final projects. You may want to form a reading group to better understand the research papers. In this case, you still have to submit your own paper summaries; in addition, please acknowledge the colleagues in your reading group.

Collaboration outside your project group or reading group on all class components must be acknowledged. If you use any published or unpublished work in your own work, you must give full citation.

Acts of cheating and plagiarism will be reported to the appropriate administrative bodies. You are responsible for knowing, and will be held to, the Honor Code of the University and of the Department.

Every part of this syllabus is subject to adjustment as the semester progresses. Please contact me as soon as possible if you are dissatisfied with the course policies, lectures, assignments, grading, etc.; I will be happy to accommodate reasonable requests for modifications.