

# UCF

## School of Electrical Engineering & Computer Science

### COP 4600: Operating Systems

#### Fall 2004

## Syllabus

**Instructor:** Euripides Montagne    Tele.: 823-2684    email:[eurip@cs.ucf.edu](mailto:eurip@cs.ucf.edu)

**Lecture meetings:**

MW 12:00 (Noon) – 1:15 p.m.(ENG 427 )

**Office hours:**

(CSB 239) MW 3:00 p.m. – 5:00 p.m. and TR 2:00 p.m. to 4:00 p.m.

**TA:** Cheng Hao

email: [haocheng@cs.ucf.edu](mailto:haocheng@cs.ucf.edu)

**Office hours:** TBA

CCI 202:

**Course Outline:** The goal of the course is to teach fundamentals concepts and design principles of operating systems.

**Course Topics:** Operating systems structure. Process management. Process scheduling. Memory Management. Virtual memory. I/O system. Performance Evaluation.

**Prerequisites:**

- COP 3530C – Computer Science III.
- COP 3402C – System Concepts/Programming.
- Proficiency in C and Familiarity with UNIX.

If you have not satisfied **all** of the above prerequisites, you **do not** belong in this class and have little chance of passing.

**Reference Guide:**

The textbook for the course is: H. M. Deitel, P. Deitel, and D. Choffnes, “ Operating Systems ” 3<sup>rd</sup> Edition, Prentice Hall, 2004. We will cover Chapters 1-13. You are responsible for the material contained in all of those chapters, even if it is not discussed in class. Time permitting we may cover parts of Chapter 19.

**Style of Class Meetings:**

Class meetings will not consist of traditional lectures, with the instructor doing most of the talking and the student doing most of the listening. Rather, meetings will consist of discussions on each topic and the instructor will help guide the discussion by asking questions.

**Grading Policy:**

- (20%) **Exam #1** – closed book, closed notes exam given in class.
- (20%) **Exam #2** – closed book, closed notes exam given in class.
- (25%) **Final Exam** – closed book, closed notes comprehensive exam given during final exam week.  
**Note:** You must score at least 60% on this exam to pass the course.
- (25%) Programming project – a large, multi-part simulation of a multiprogramming operating system. Written in C on a UNIX system, this project is not easy, but can be done in the time allotted. The grade for this project will be divided between your C code, one or more demonstrations of your project, your documentation and quizzes given on selected topics from the project.
- (10%) Concurrent programming assignment.

**Letter grades:** 90-100: A, 80-89: B, 70-79: C, 50-69: D, Below 50: F.

**Note:** Any academic dishonesty (including, but not limited to, Cheating, copying and/or plagiarism) with respect to any exam or assignment in this class will result in a grade of **F**, following by the usual procedures for dealing with such behavior, as describe in the *UCF Golden Rule : a handbook for students*.

**The Semester Plan:** Tentative.

Aug. 26 -Operating Systems Fundamentals.

Aug. 31 -Computer System Structure.

Sept. 02-Interrupt Handling.

Sept. 07-Interrupt Handling

Sept. 09-Operating System Structure.

Sept. 14-Processes and Threads.

Sept. 16- Process Synchronization.

Sept. 21- Process Synchronization.

Sept. 23- Process Scheduling.

Sept. 28 - Review

Sept. 30- **First Midterm Exam.**

Oct. 05-Memory Management

Oct. 07- Memory Management

Oct. 12-Virtual Memory

Oct. 14-Virtual Memory

Oct. 19- I/O structure

Oct. 21- Disk Scheduling

Oct. 26- Review

Oct. 28- **Second Midterm Exam.**

Nov. 02- File System

Nov. 04- Resource Allocation and Deadlock

Nov. 09- Resource Allocation and Deadlock

Nov. 16 – System Performance Evaluation

Nov. 18 – System Performance Evaluation

Nov. 30 - Review

Dec. 02 -**Final Exam**

### **COP 4600 Programming Project ( Fall 2004)**

This project is divided into 4 parts to make it more manageable. Details will be given out well before the due dates for each part(the parts of the project are called *objectives*). This project must be written in C(**not** C++) on a UNIX system. The standard for this class will be the Sun Sparc system in the main computer lab. called Olympus. You are welcome to write and test code on some other system, if you wish, but it will be graded on Olympus and if it does not work there, it does not work. You will be given an Olympus account and, once the project has begun, should check your e-mail regularly for updates.

To pass this course, you **must** successfully complete objectives 1, 2 and 3, and the concurrent program. No exceptions.

Each objective will have a due date and points will be subtracted for submission after that date. Also, after each due date some evaluations of you progress will be made. This may include a walk through of your code with the instructor or grader, a quiz on the objectives that was just completed (including questions about code, data structures and/or algorithms) or a short, written description of the purpose and implementation of the objective. Details will be handed out with each objective and I reserve the right to change the method of evaluation at any time.

In general, this project will give you a better understanding of the data structures and control flow of a multiprogramming operating system and also provide you with experience in developing and debugging a complex software project.

**Lets make this clear:** when working on the project, you are allowed to talk to other students about programming concepts, C syntax and general solutions to problems (algorithms or questions about the project instructions), but you are not allowed to share, exchange or copy code. Both the source and the recipient of any exchange of code are equally at fault.

**Important Dates:**

- **Classes Begin August 23<sup>rd</sup>.**
- **Withdrawal Deadline is October 22<sup>nd</sup>.**
- **Classes End December 4<sup>th</sup>.**
- **Summer Holidays are:**
  - **Labor Day Sept. 6.**
  - **Veteran's November 11<sup>th</sup>.**
  - **Thanksgiving Nov. 25 – 27.**