SYLLABUS

Course: Operating Systems, CS 430, (Undergraduate)  18 Jan 05
Instructor: Paul A. Courchene
Contact:  paul.cour1@verizon.net  preferred method of contact
Tel:  (603) 891-4964 (Home)  Leave Message and Telephone Number

Prerequisites: C/C++ Programming  (or ability to write and compile a simple C program), Computer Architecture.

Exams format: Closed book, Multiple Choice, Mix and Match, Short Essay.


Course Description:
An Operating System (OS) is a complex software program that is used to manage computing resources. An OS typically has four major components: Process management, I/O Device management, Memory management and File management [2]. In another context, an OS may be viewed as an extended machine or a virtual machine that allows many users access to these resources on an interactive or timely basis.

This course will study the allocation and management of those resources and how they are structured (monolith, layered, virtual machine and client-server models). The emphasis of discussion will be on general OS building blocks, the Unix OS, and Linux OS as well. Examples of Linux and Unix as used on the Internet and Web, will show practical implementation as a machine capable of performing complex computing tasks. While we are not attempting to certify you as a Linux technician, examples of Linux will serve to teach you how boiler-plate computing concepts are implemented in today’s OSes.

Objectives:

To learn about the evolution of a generic OS
To review the layout of a computing machine and its hardware components.
To learn about the basic building blocks of a generic Operating System.
To learn about the concept of User space versus Kernel space.
To acquire skill in accessing the components of the OS, as a User Interface
To learn about the principle functions of an OS (Scheduling, Memory Management, File Systems ).
To learn about efficiency and performance issues in scheduling tasks and processes.
To be able to define and apply key terms as listed in the attached Schedule.
To design and compile basic programs using the C Programming language and GNU Compiler, thus creating, managing and deleting of threads and processes.

Schedule and Reading Assignments: attached.
Homework:

- **Project 1** 5 points  Construct a layout dwg. of a complex computing machine
- **Project 2** 10 points  Linux Process (creation & deletion) Project
- **Project 3** 10 points  Linux Threads Project

**Classroom Behavior**

Respect for others. Please offer the respect to others that you would expect to receive as well. Disruptive behavior will not be tolerated.

**Attendance:**

Attendance is required. Prior Email notification of intended absence is encouraged.

**Evaluation:**

Undergraduate students will be evaluated and assessed as a group of peers.

Graduate students will be evaluated as a separate group.

Presentations will be graded by all members of the Class. A score will be assessed by averaging individual member inputs.

**Grades and Evaluation:**

You will be graded as follows:

- Mid-Term (closed book)  25 %
- Final-Exam (closed book)  25 %
- Homework Projects  20 %
- Project with paper  10 %
- Presentation of Findings  10 %
- Attendance  10 %
- Total for Course  100 %
**Personal Ethics**

There will be a number of Instructor lead presentations where a lot of personal interaction will be encouraged. It is a well known fact that students who do interact, learn more by this interaction. However, personal interaction is not acceptable on Homework assignments, nor Mid-term or Final Exams. If you are observed looking at someone else’s paper or “work” during an Exam, I will simply pickup your paper and you will be asked to report to the Dean of the College, for cheating. I will therefore issue a grade of “F”, on your behalf.

I invite you to review the policy and tenets of “Academic Honesty” as detailed in the Rivier College Undergraduate and Graduate Catalogs.

**Biblio-Citations:**
Paper of Investigation

Project and paper:

You will be assigned as a member of a team, and will investigate “new ideas” or advances in Operating Systems, today. Your task as a member of the team will be to investigate, describe or summarize and document new OS ideas either implemented today, or ideas being discussed by Industry or Academia. These ideas will enhance the management of computer resources for improved throughput, efficiency, scalability, scheduling, process control, preemptibility, filesystems, data storage issues, etc.

Names of contributing team members will also be listed on the Front Cover.

Team Member(s) will be required to give a short (< 10 min.) presentation on your findings, to the Class.

Paper and presentation will be graded on originality, academic excellence and good research & problem solving practices.

A short paper of 2 – 4 (?) pages will be prepared, to document your research. This paper will have as a minimum, the following sections (you may delete a section if you feel that it is not relevant for your specific case.

1. Problem Statement. This section should contain a precise definition of the problem being investigated. Some focus on defining the problem will assist you in understanding what needs to be researched. Occasionally, a paper of this nature may have a few key questions following the problem statement, to help focus on what needs to be done or what needs to be researched.

2. Background. Is there some background that you can gather, to help understand the problem?

3. Discussion. Have there been some previous attempts to work this problem or solve this problem.

4. Analysis. What evidence or data do you have to support any conclusions or outcomes?

5. Summary. How can you summarize your findings?

6. Conclusion. What conclusions may be drawn as a result of this investigation or research?

7. Bibliographic Citations. What was your source of material? Is there other literature that supports the investigation or research of this problem, or similar problems? Will you supply at least one copy for the review of members of the Class?

18 Jan 05