

# COSC 416: Operating Systems and Computer Architecture

Messiah College

Spring 2007 Syllabus

## Catalog Description, Adapted

Overview of the parts of a modern operating system: process scheduling and synchronization, deadlocks, virtual memory management, file systems, protection and security, distributed systems. Windows XP and Linux as case studies. Examples coded using Java. Prerequisite: COSC 317, Computer Hardware Organization.

Course home page: [www.messiah.edu/~chase/csc/416](http://www.messiah.edu/~chase/csc/416)

## Instructor

Dr. Gene B. Chase. <chase AT messiah DOT edu> Office, Frey 123, extension 2770; home, 766-7904 (before 10 pm). Office hours: I have a signup sheet on my office door for drop-in appointments. Other times are available by asking me in advance. The course home has a link to my semester schedule.

## Goals and Objectives

My goals for this course are as follows. • You should be able to argue that a program that does parallel processing satisfies its design postconditions and side effects. • You should be able to determine when a computer resource should be replaced (faster processor, larger swap area, faster memory). • You should be able to choose Linux or Windows based on a knowledge of the characteristics of a good operating system and whether those two families of operating systems are good.

The course has the following objectives. At the end of the course, you will be able to—

1. discuss various aspects of modern operating systems, including
  - a. process and processor management;
  - b. memory and disk management;
  - c. monitoring and optimizing system performance;
  - d. network, security, and accounting considerations;
  - e. two case studies: Linux and WinXP;
2. use the Windows and Linux operating systems to illustrate some of the subjects discussed in the course; and,
3. investigate ideas about operating systems on your own that go beyond the material presented in class, and present them in written and oral form.

## Outcomes

In brackets I mention for each outcome the objectives above which that outcome meets.

1. You will write a Java program to use competing processes to simulate a problem, to show that you understand process management. [1a]
2. You will pass three exams that test your understanding of the underlying model that a modern operating system uses to handle multiple users. [1a–e, 2, 3]
3. You will complete a Linux laboratory to demonstrate that you can use Linux from a command line prompt and that you understand the different philosophies of Linux & Windows. [1e, 2]
4. You will complete a Windows “stress test” lab to demonstrate that you understand which operating system parameters affect an operating system's performance in which ways. [1c]
5. You will write a paper on an approved operating systems topic of your choice that involves some experimentation or computer coding and reading of the professional literature. [3]

## Text

*Operating System Concepts with Java*, Seventh Edition, by Abraham Silberschatz, Peter Galvin, and Greg Gagne. NY: John Wiley, 2007. ISBN: 0-471-76907-X. The textbook's web page is linked from our course web page.

## Bibliography

For your final written report, you will need to read two articles for background from two computer journals. See your text's bibliography beginning on p. 899, Murray Library's open stacks, and the on-line journal resource Academic Search Premier available from the library home page, [www.messiah.edu/murraylibrary](http://www.messiah.edu/murraylibrary) where you can find articles in academic journals, trade journals, newspapers, and magazines.

## Grading Policy and Course Requirements

45% Three exams, not cumulative, at 15% each.

15% Hands-on project. Milestones and other guidelines will be posted on the course website.

40% Four homework projects: File Systems (Oral report), Linux introduction, Synchronization in Java, and a WindowsXP stress test.

The standards for letter grades as voted on by the Messiah College faculty are located at the following URL, linked from our course web site: [www.messiah.edu/departments/mathsci/gradeChart.htm](http://www.messiah.edu/departments/mathsci/gradeChart.htm)

**Class participation** can modify the final grade up (bonus for exceptional contribution) or down (penalty for unexcused absences). Classes will discuss the readings for the day. Everyone is expected to participate. Attendance is required. Assignments are due at 8 a.m. on the day that they are to be submitted. **Late assignments** are penalized at 10% per day or fraction, a weekend counting as a single day. Late assignments are not accepted after the assignment has been returned to the class. If your assignment is late because you are ill, bring a written note from the Engle Health Center as evidence that you are well enough to return to class, and late penalty will be waived.

## Legal Matters

Q **Academic integrity** guidelines are in your Student Handbook, which is linked from the course website and are incorporated into this syllabus by reference. You are not properly enrolled in this course unless you have read, understood, and agreed to abide by the guidelines posted.

## Americans with Disabilities Act

Any student whose disability falls within Americans with Disabilities (ADA) guidelines should inform me at the beginning of the semester about any special accommodations or equipment needs necessary to complete the requirements for this course. You must register that need with the Office of Disability Services in Hoffman Hall. If you have questions, call Dr. Keith Drahn at ext. 5358.

## Course Overview

An operating system is your computer's resource manager. This course focuses on the following three examples of how operating systems improve efficiency while making the computer easier to use:

1. Multiprogramming makes computers more efficient by simultaneously having several programs in memory so there is less time wasted when parts of the computer are idle. Operating systems may do this in hardware using multiple processors or in software using multiple processes. A single process can itself use multiple threads of execution. Multiprogramming creates a complex scheduling problem, the possibility of deadlock, 8), and a complex memory management problem.
2. Virtual memory increases efficiency and enhances multiprogramming by allowing more processes to be in memory simultaneously. This is achieved not by adding actual memory but by making it seem to the programmer that memory is larger than it actually is. The operating system swaps data in and out of memory to create this illusion.
3. Sharing data or programs in memory is another way to gain efficiency, but it creates an interprocess communication problem. Similar gains are made by sharing a file system, and a whole

new set of problems arise. Distributed systems extend this idea to sharing files over a network.

### Acknowledgments

I gratefully appreciate four folks who provided me with valuable insights into teaching operating systems. **Richard Dent**—who first taught Operating Systems at Messiah College in the 1980s—showed me how it should be done. **Dr. Paul van Arragon**—who taught this course 1992–1999 at Messiah College—generously provided me with his private notes and public handouts. Alumnus **Joel Worrall**—who did a second semester of Operating Systems with me as an independent study—revised some of my labs and handouts based on the things we learned together in his course. **Brian Coogan**—who taught Unix System Administration at Messiah College in January, 2002 and 2006—gave me some practical pointers about Unix that went beyond what I taught myself when I was on a 1987–1988 leave of absence from Messiah College to work with Wycliffe Bible Translators. That is when I first used Unix professionally.

### Tentative Schedule

Visit the course web for the readings and assignments to be prepared for quizzes and discussion. This schedule contains only the fixed days.

<i>Date</i>	<i>Assignment</i>
-------------	-------------------

---

Q Feb 5, 7, 9

Q Feb 12, 14, 16

Friday, Feb. 16 begins **Presentation #1**

Q Feb 19, 21, 23

Friday, Feb. 23 **Lab #1: Linux** (in Frey 366)

Q Feb 26, 28, Mar 2

Friday, Mar 2 **final paper topic is due**: title, paragraph description, and five references no more than three of which can be from the internet (unless they are also published in print)

Q Mar 5, 7, 9

Monday, Mar 5, **Linux Lab #1 is due**

Friday, March 9 **Exam 1**; Dr. Chase will be at the annual SIGCSE computer conference

Q Mar 12, 14, 16

—**Saturday, March 17–Sunday, March 25: Spring Break**—

Q Mar 26, 28, 30

Friday, March 30, no class. Work on final paper, Exam 2, and synchronization homework; Dr. Chase will be at his son John's wedding

Q Apr 2, 4

Monday, April 2 **Exam 2**, again proctored by a colleague; Dr. Chase will be flying home

—**Easter Break, Friday, April 6 through Monday, April 9**—

Q Apr 11, 13

Wednesday, April 11: **Synchronization homework due**

Wednesday, Apr 11: **Lab #2: WinXP loading experiment** (Frey 366)

Q Apr 16, 18, 20

Q Apr 23, 25, 27

**Friday, April 23: Lab #2 due: Win2K loading experiment**

Q Apr 30, May 2

Q May 4 Class cancelled for School of MEB Scholarship Day: attendance required except when you have classes in other Schools.

Q May 7

Final paper due 8 a.m. Review of course.

Q May 10

**Exam 3**, not comprehensive, 8:00–10:00 a.m.