

## COURSE SYLLABUS

### CMPT 332: OPERATING SYSTEMS PRINCIPLES

---

#### Catalogue Description

An introduction to the principles of modern operating systems. The synchronization and communication of cooperating processes. Process Scheduling. Virtual Memory. File System design and organization. Introduction to distributed systems.

<b>Prerequisite(s):</b>	CMPT 280, (CMPT 215 or CME 331)
<b>Class Time and Location:</b>	MWF 9:30 A.M. Thorvaldson 124 Tutorials in SPINKS S320 M: 3:30 p.m., W: 2:30 p.m. You may attend either tutorial, but space is limited.
<b>Website:</b>	bblearn.usask.ca

#### Instructor Information

<b>Instructor</b>	Dwight Makaroff
<b>Contact:</b>	Email: makaroff@cs.usask.ca, Phone: 966-8656
<b>Office Hours:</b>	Location: Thorvaldson 281.3 Hours: By appointment

#### Course Objectives

After completing this course, students should be able to do the following tasks:

- Demonstrate and illustrate how application software accesses computer hardware through the abstractions provided by the operating system and how the operating system shares hardware resources between processes, tasks, threads, and users.
- Utilize system library functions robustly in the implementation of applications that access operating system facilities.
- Correctly abstract operations through the use of Application Programmer Interfaces and Virtual Function Interfaces.
- Design, implement and document system-level software in a small- team environment.
- Demonstrate the operation of well-known theoretical algorithms with respect to deadlock, process and disk scheduling, and memory management.
- Illustrate the separation of policy and mechanism with examples from operating system design and implementation.
- Design algorithms to provide concurrent access to shared resources and implement these algorithms in the following environments in the C programming language: UNIX processes, various UNIX and Windows threads packages.
- Explain time, space and complexity tradeoffs in operating system implementation issues, demonstrating their application with approximate solutions for various resource sharing problems.
- Modify existing system-level source code to add new functionality.
- Compare and evaluate options for filesystem implementation and use as well as alternative mechanisms for concurrency control between processes.

## Student Evaluation

### Grading Scheme

Requirements: There will be 4 equally-weighted assignments (due approximately every 3-4 weeks), a mid-term exam (held in-class), and a final examination during the regular examination period. Exact dates will be announced as the course progresses. The approximate weightings for the assignments and examinations are as follows:

Participation, including Online Discussions	5%
Assignments	28 %
Midterm Exam October 31, 2014	17 %
Final Exam	50%
Total	100%

### Criteria That Must Be Met To Pass

All components of the course must be complete in order to achieve a passing grade in the course. This includes both exams and every assignment. Failure to do so will result in an automatic failure of the course. Submission of assignments must be a credible attempt to solve the problems in the assignment in the judgment of the marker.

### Final Exam Scheduling

The Registrar schedules all final examinations, including deferred and supplemental examinations. Students are advised not to make travel arrangements for the exam period until the official exam schedule has been posted.

**Note: All students must be properly registered in order to attend lectures and receive credit for this course.**

## Textbook Information

### Required Text

- Title: Modern Operating Systems
- Author: Andrew S. Tanenbaum
- Publisher: Pearson/Prentice Hall
- Edition/Year/ISBN: 4th Edition/2014/978-0133591620
- Additional information: This book covers more material than we can do in one course. We will cover the first 5 chapters with some discussion on later chapters if time permits. It is likely that the 3rd edition will be sufficient for most students. Nevertheless, you should ensure that you have access to a 4th edition for possible reference to exercises and specific sections in the new textbook.

### Recommended Texts

- Additional information: optional, but not necessary for anyone. This provides a different point of view.
  - Title: Operating Systems: Internals and Design Principles
  - Author: William Stallings

- Publisher: Pearson/Prentice Hall
- Edition/Year/ISBN: 7th Edition/2012/0-13-230998-1
- Yet a third option.
  - Title: Operating System Concepts
  - Author: Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne
  - Publisher: John Wiley and Sons
  - Edition/Year/ISBN: 9th Edition/2012/978-1-118-06333-0
- Essential for everyone to have for their career. Many should already have this.
  - Title: The C Programming Language
  - Author: Brian Kernighan and Dennis Ritchie
  - Publisher: Prentice Hall
  - Edition/Year/ISBN: 2nd/1988/0-13-110362-8

## Lecture Schedule

Topic	Topic/Subtopics
Introduction	Overview, history of Operating Systems. Introduction to the computing environment for this course. Possible review of C and UNIX/ Linux.  Readings: Chapter 1.1-1.7 (pp. 1-73) - Tanenbaum
Process Description, Concurrency Control and Inter-Process Communication	processes, threads, and address spaces; I/O devices; synchronization; mutual exclusion; semaphores; monitors; Inter-process communication; deadlock; examples.  Duration: 4.5 weeks.  Readings: Chapter 2.1, 2.2, 2.3, 2.5, 5.1, 5.2, 6, Section 8.2.4, 8.2.5, 8.2.6
Process Scheduling	types of scheduling; uniprocessor scheduling; multiprocessor scheduling; examples.  Duration: 2 weeks  Readings: Chapter 2.4, Section 8.1.4
Memory Management	address binding; virtual memory; address translation; management policies for demand paged virtual memory; examples.  Duration: 3 weeks  Readings: Chapter 3
I/O and File Systems	organizing the I/O function; disk architectures, RAID; disk caching; file organization; secondary storage management; protection and security; examples.  Duration: 2 weeks  Readings: Chapter 4 and Sections 5.3 and 5.4

Distributed Systems and Virtualization	client-server paradigm; Remote Procedure Call; clusters; process migration; distributed file systems, virtual machines.  Duration: 2 weeks (If time permits)  Course Notes and Chapter 7 and Chapter 8
--	--

## Course Procedures

- Facilities. Students will be using the Linux/Windows computers in the Spinks Lab to do most of their work, and are expected to be familiar with the use of these facilities.
- This course will be administered with BlackBoard. Those unfamiliar with the use of BlackBoard are strongly encouraged to make use of the resources available to learn BlackBoard. BlackBoard has a Discussion Forum and it will be used to disseminate answers to questions regarding assignments, lecture material, etc. **Many important things may happen there that will not be repeated in class, so daily reading of the bulletin board is required.**
- The course will make regular use of Internet facilities. Page Locations will be given during lectures and on BlackBoard.
- Assignment descriptions will be made available in electronic form only. Submissions will be in electronic form only.
- Part of the evaluation in the course will include participation, including participation in the electronic component of the course through BlackBoard. Your involvement in this area of the course will be electronically tracked on a periodic basis.
- Assignments are to be done in groups of **at most** two students.
- Version control using SVN at [svn.cs.usask.ca](http://svn.cs.usask.ca) is required for every assignment and will be verified by submitting meaningful SVN logs.
- Final Note: The purpose of CMPT 332 is to provide a basic understanding of operating systems principles: the parts of an operating system, how they are structured, the important policies governing their operation, and the implementation issues. Although examples will be drawn from several operating systems throughout the course, it is not the purpose of this course to provide training in any particular operating system. The specific operating system that we will be developing code for is UNIX-like in some rudimentary ways, but is not in any way required to have the same types of functionality. The class will follow the textbook in content, but not necessarily organization. Students will be responsible for reading the text and learning the material in it. As much as possible, lecture time will be used primarily for
  - highlighting specific material in the text,
  - covering supplementary material as required, and
  - answering student questions.

## Policies

### Late Assignments

In general, late assignments are not accepted. If there are extenuating circumstances whereby it is impossible to hand in an assignment on time, please see the instructor for special accommodations. This policy may change for different assignments.

## Missed Assignments

If an assignment is missed, it is not possible to receive a passing grade in the course.

## Missed Examinations

1. "Students who have missed an exam or assignment must contact their instructor as soon as possible. Arrangements to make up the exam may be arranged with the instructor. Missed exams throughout the year are left up to the discretion of the instructor if a student may make up the exam or write at a different time. If a student knows prior to the exam that she/he will not be able to attend, they should let the instructor know before the exam."
2. "Final exams - a student who is absent from a final examination through no fault of his or her own, for medical or other valid reasons, may apply to the College of Arts and Science Dean's office. The application must be made within three days of the missed examination along with supporting documentary evidence. Deferred exams are written during the February mid-term break for Term 1 courses and in early June for Term 2 and full year courses."

(2007/08. <http://www.arts.usask.ca/students/transition/tips.php>)

## Incomplete Course Work and Final Grades

When a student has not completed the required course work, which includes any assignment or examination including the final examination, by the time of submission of the final grades, they may be granted an extension to permit completion of an assignment, or granted a deferred examination in the case of absence from a final examination. Extensions for the completion of assignments must be approved by the Department Head, or Dean in non-departmentalized Colleges, and may exceed thirty days only in unusual circumstances. The student must apply to the instructor for such an extension and furnish satisfactory reasons for the deficiency. Deferred final examinations are granted as per College policy.

**In the interim, the instructor will submit a computed percentage grade for the course which factors in the incomplete course work as a zero, along with a grade comment of INF (Incomplete Failure) if a failing grade. In the case where the instructor has indicated in the course outline that failure to complete the required course work will result in failure in the course, and the student has a computed passing percentage grade, a final grade of 49% will be submitted along with a grade comment of INF (Incomplete Failure).**

If an extension is granted and the required assignment is submitted within the allotted time, or if a deferred examination is granted and written in the case of absence from the final examination, the instructor will submit a revised computed final percentage grade. The grade change will replace the previous grade and any grade comment of INF (Incomplete Failure) will be removed. For provisions governing examinations and grading, students are referred to the University Council Regulations on Examinations subsection of the Calendar.

(2011 University of Saskatchewan Calendar/Academic Courses Policy)

## Academic Honesty

The University of Saskatchewan is committed to the highest standards of academic integrity and honesty. Students are expected to be familiar with these standards regarding academic honesty and to uphold the policies of the University in this respect. Students are particularly urged to familiarize themselves with the provisions of the Student Conduct & Appeals subsection of the University Secretary Website and avoid any behaviour that could potentially result in suspicions of cheating, plagiarism, misrepresentation of facts and/or participation in an offence. Academic dishonesty is a serious offence and can result in suspension or expulsion from the University.

All students should read and be familiar with the Regulations on Academic Student Misconduct, <http://www.usask.ca/secretariat/student-conduct-appeals/StudentAcademicMisconduct.pdf>, as well as the Standard of Student Conduct in Non-Academic Matters and Procedures for Resolution of Complaints and Appeals, <http://www.usask.ca/secretariat/student-conduct-appeals/StudentNon-AcademicMisconduct2012.pdf>.

Academic honesty is also defined and described in the Department of Computer Science Statement on Academic Honesty:

<http://www.cs.usask.ca/undergrad/honesty.php>.

For more information on what academic integrity means for students see the Student Conduct & Appeals subsection of the University Secretary Website at:

<http://www.usask.ca/secretariat/student-conduct-appeals/forms/IntegrityDefined.pdf>

### **Examinations with Disability Services for Students (DSS)**

Students who have disabilities (learning, medical, physical, or mental health) are strongly encouraged to register with Disability Services for Students (DSS) if they have not already done so. Students who suspect they may have disabilities should contact DSS for advice and referrals. In order to access DSS programs and supports, students must follow DSS policy and procedures. For more information, check <http://www.students.usask.ca/disability/>, or contact DSS at 966-7273 or [dss@usask.ca](mailto:dss@usask.ca).

Students registered with DSS may request alternative arrangements for mid-term and final examinations. Students must arrange such accommodations through DSS by the stated deadlines. Instructors shall provide the examinations for students who are being accommodated by the deadlines established by DSS.