Today’s Agenda

- Administrivia
  - Course overview
  - General course structure
  - The textbook
  - Term work

- Course Objectives

- OS Review
  - Try to make sense of the subject matter
Course Overview

Course web page:

http://www.cs.umanitoba.ca/~comp3430

- Practically, everything you need to know will be on the course web page.
- Except,
  - the assignment submissions and
  - the discussion forum will be on Desire2Learn.
- More on this later.
General Course Structure

- We’ll be following the textbook closely; get a copy
- 75-minute lectures
- 15-minutes Q&A
- Three labs; attendance is (REQUIRED), as we will discuss the next assignment
- Participate in class—ask questions!
Resources

- The textbook
  - Really outstanding written by current experts
  - Allows you to actually figure out how things work
  - Way better (and way less expensive) than any alternative
  - Second edition still has typos; find them and report to me
  - Think of it as helping you to understand, and dig deeper than, the lecture, section, and project material

- Other resources
  - Many online; some of them are essential
Term Work

There will be four programming assignments:

**Assignment 0**  Warm up exercise with C programming language that you have excelled in COMP2160

**Assignment 1**  Adding a system call

**Assignment 2**  Building a thread package

**Assignment 3**  Modifying the file system

You are encouraged to recommend a partner by the first lecture, by emailing your names to comp3430@cs.umanitoba.ca I will assign partners by the end of the first week and randomly assign the remaining partners.
Assignment 0 will be done individually. The rest will be done in 2-person teams.

All the assignments will be done in C programming language. No exceptions. I urge you to re-visit your COMP2160 notes in advance to re-familiarize yourselves with the language.

Late Policy: Unless otherwise indicated, assignments are due at the following lab day (B01 Tuesday) for each. If you hand in an assignment late, we will take off 10% for each day (or portion thereof) until the following Friday, 5:00pm. In short, we will grant up to three late days with a penalty.
Course Objectives

In this class you will learn:

- What are the major components of most OSs?
- How are the components structured?
- What are the most important (most common) interfaces?
- What policies are typically used in an OS?
- What algorithms are used to implement these policies?
Philosophy

- You may not ever build an OS
  - But as a computer scientist or computer engineer you need to understand the foundations
  - Most importantly, operating systems exemplify the sorts of engineering design trade-offs that you’ll need to make throughout your careers: compromises among and within cost, performance, functionality, complexity, schedule, ...

- We want you ...
Philosophy

- You may not ever build an OS
  - But as a computer scientist or computer engineer you need to understand the foundations
  - Most importantly, operating systems exemplify the sorts of engineering design trade-offs that you’ll need to make throughout your careers: compromises among and within cost, performance, functionality, complexity, schedule, ...

- We want you ...
  - to love this course!
Philosophy

You may not ever build an OS
  But as a computer scientist or computer engineer you need to understand the foundations
  Most importantly, operating systems exemplify the sorts of engineering design trade-offs that you’ll need to make throughout your careers: compromises among and within cost, performance, functionality, complexity, schedule, ...

We want you ...
  to love this course!
  to remember it in 5 years as one that paid off!
Any Questions?
Any Questions?

"Harris, when I said 'any questions' I was using only a figure of speech."

http://www.businesscartoons.co.uk/shop/index.php?act=viewProd&productId=300