CS 350: Operating Systems

Tim Brecht (8:30)

Jeremy Barbay (2:30)

Note: this info is available on the course web page

Course Objective

- This course provides an introduction to operating systems; what they do, how they are used, and how they are implemented

Course Modules

- Module 1: Processes
- Module 2: Synchronization
- Module 3: Memory Management
- Module 4: NachOS
- Module 5: Virtual Memory
- Module 6: Scheduling
- Module 7: Input/Output
- Module 8: File Systems
- Module 9: Interprocess Communication
- Module 10: Security

Course Information

- Intended Audience:
  - CS 350 is a required course for all CS majors
- Related Courses:
  - Prerequisites: CS 240, 246, 251 and enrollment in a CS major plan
  - Successors: CS 343 (Concurrency and Parallel Programming) and many 4th year courses
  - Antirequisites: CS 354, E&CE 354, GENE 241

Course Personnel

- Instructional Support Coordinator (ISC)
  - Fenglian Qui
  - Organizes and manages TAs, assignment marking, web page, grades, etc.
- Instructional Apprentice (IA)
  - Claus Spitzer and Qiang Wang
  - Will answer newsgroup questions, NachOS problems
- Other TAs (see course web page)
- All office hours will be posted to newsgroup

Course Documents

- Textbook
  - Silberschatz, Galvin, and Gagne, *Operating System Concepts*, Wiley & Sons.
    - Officially, the 7th edition (with or without the XP update)
    - The 6th edition is fine as well (as is likely the 5th)
- Course Home Page:
  - **`http://www.student.cs.uwaterloo.ca/~cs350`**
  - Includes all notes and slides, which can also be purchased
- Course Newsgroup:
  - **`uw.cs.cs350`**
  - **All students expected to read newsgroup frequently**
  - **Post questions regarding assignments (instead of emailing)**

Administrivia - Grading

- Components:
  - A1, A2, A3: Mark on assignments 1 - 3, as a percentage
  - M: Midterm exam grade, as a percentage
  - F: Final exam grade, as a percentage
- Grade computation:

```
$Normal = (0.1*A1 + 0.15*A2 + 0.1*A3) + 0.2*M + 0.45*F$
$Exams = (0.2*M + 0.45*F) / 0.65$
$Assigns = (0.1*A1 + 0.15*A2 + 0.1*A3) / 0.35$
If ($Exams < 0.5$)
    $Grade = min(Normal, Exams)$
Else $Grade = Normal$
```

Administrivia - Exams

- Midterm exam
  - Tentatively scheduled for the evening of Tuesday, Feb. 27, location TBA
  - Notify your instructor if you have a conflict
- Final exam
  - Details will be announced when available
- Reappraisals
  - Resubmit you entire exam/assignment for remarking (time limit on all reappraisals)

Administrivia - Assignments

- Assignments will be done in groups of up to three students (really a year long project)
  - You can work alone or in smaller groups, though we recommend three
  - Grading is independent of group size
  - Assignment 0: Form your group
    - Try a “Partner wanted” message on the course newsgroup if you are trying to find a partner
    - Partners need not be in the same lecture section
  - Also note the policies for “divorce”

Administrivia - Assignments

- Assignments have a number of *slip days*
  - Submit an assignment late without penalty
  - You have a fixed number of days that you can use
  - Exception: last assignment cannot be submitted after the last day of classes
- Reappraisals
  - Contact the TA who marked it. They can explain the marking.
  - Follow reappraisal procedure on web page.

Administrivia - Assignments

- All assignments will be done with NachOS
  - Simulates an operating system running on a MIPS-based workstation
  - This has been modified for use at Waterloo.
    - Download a new version once available through web site
- NachOS code base is 10,000 lines of C++
  - Large code base to learn and work with, so…

Administrivia - Assignments

START ASSIGNMENTS EARLY!

- Assignment 1 will be available shortly. Put together a group and start soon!
- We’re not going to ease into the term - we’re jumping right in!

Administrivia - Assignments

- NachOS will run on Linux systems, so you can work at home
  - You must ensure that your assignments work in the CSCF environment, since that’s where it will be marked.
- Make sure you read and understand what must be submitted for each assignment
  - Part of assignment is demonstrating that it works. Guidelines for producing good tests will be provided on web site.

Plagiarism and academic offenses

- Don’t think we won’t report problems just because you’ve made it to third year!
- **Minus 100%** on the assignment makes it hard to pass the course
- Nice explanation of plagiarism on-line

  http://arts.uwaterloo.ca/arts/ugrad/academic_responsibilty.html
- **Read this and understand it**
  - **Ignorance is no excuse!**
  - Questions should be brought to one of the instructors
- Plagiarism applies to both text and code
- You are free (even encouraged) to exchange ideas, but ***no sharing code***

Plagiarism (2)

- Common mistakes
  - Excess collaboration with other groups
    - Share ideas, but no design or code!
  - Using code from other sources (like previous offerings of this course, your own code from)
- We will be comparing assignment submissions using cheat detection software
- Possible penalties
  - First offense
    - **-**100% for that part of the course
    - This makes it hard to pass the course
    - You can not drop the course with an academic offense
  - Second offense
    - Expulsion is possible