

CSc 360

Operating Systems

Introduction

Jianping Pan

Summer 2015

About the course

- *Introduction* to **Operating Systems**
 - (A01) MWTh 2:30--3:20pm, ECS124
 - <http://www.csc.uvic.ca/~csc360> on connex*
 - lectures, tutorials, additional resources, etc
 - assignments, gradebook, chat room, wiki, etc
 - prerequisites
 - Data structures (CSc 225 or the new 226)
 - Computer architecture (CSc 230 or CENG 255)
 - System programming (CSc/SENG 265, CENG)

5/4/15

CSc 360

2

Message from Undergrad Advisor

- New email: cscadvisor@csc.uvic.ca
- Do not have the prerequisite course(s)?
 - need to obtain a waiver
 - otherwise, prerequisite drop after the first week
- Taking the course more than twice?
 - need to have a letter from the Chair and the Dean
 - otherwise, being dropped from the class
- Register your preferred email in connex!
 - email reflector: csc360k15@connex.csc.uvic.ca

About the course instructor

- Dr. Jianping Pan
 - pan@uvic.ca
 - always include [csc360] in your email subject line
 - office hours: WTh 1:30--2:20pm
 - or by appointment
 - ECS 566, x5796
 - work experience
 - UVic, industry research labs, UWaterloo postdoc
 - research area
 - computer networks and distributed systems
 - <http://web.uvic.ca/~pan>

About the lab/tutorial instructor



- Dawood Sajjadi
 - email on blackboard
- Tutorials: start next week!
 - (T01) F 2:30--3:20pm, ECS 104
 - tutorial lectures
 - C, libc, sockets, pthreads, ...
 - assignment help
 - spec go-through, common problems, ...
 - practice problems

Course materials

- Required textbook
 - Operating system concepts, 7th or newer editions
 - 6th edition: different chapter schedule
 - online resources
 - <http://codex.cs.yale.edu/avi/os-book/OS7/os7c/>
 - errata, slides, practice exercises and solutions
- Explore further
 - web links @ course web site
 - Google!

What's operating system?

- OS is a *special* program (computer software)
 - run “directly” on computer hardware
 - CPU, memory, I/O, etc
 - support many other programs
 - system programs: shell, compiler, assembler, etc
 - application programs: editor, browser, game, etc
- Examples
 - Linux, Unix, Windows, and many others
 - iPhone iOS, S60, Maemo, Android, etc

What does OS offer?

- Computer systems
 - hardware, system programs (**OS+**), apps, users
- OS: between hardware and other software
 - present a virtual machine to other software
 - hide hardware details, extend hardware features
 - hardware++
 - provide controlled access to hardware
 - restrict hardware access, manage hardware resources
 - hardware--

Why do we need OS?

- C&C: control and coordinate
 - allow a program to use computer properly
 - program execution, error detection, ...
 - allow many programs to use computer properly
 - resource allocation, conflict arbitration, ...
- S&S: share and separate (protect)
 - share btw devices, programs, computers, users
 - protect one from all the others

Why do we study OS?

- How to use OS
 - not as a computer user!
 - point-and-click or copy-and-paste
 - but as a system programmer!
 - programming!!!
- How to design OS
 - or design any *complex*, large-scale software
- How to implement OS
 - or write any effective and efficient code

Course objectives

- “An introduction to the major concepts of modern operating systems and the relationship between the operating system and the architecture of computer systems.”
- Selected topics
 - process: process, thread, scheduling, synch
 - memory: memory management, virtual memory
 - storage: file systems, I/O systems

Your participation

- Lectures
 - essential for doing well in assignments/exams
- Assignments (55% total)
 - 3 programming assignments + 1 written assignment
- Tutorials
 - extra details and hints on assignments
- Exams
 - 3 midterms (15% each on May 28, June 25, July 30)
- See the course outline for schedules

Suggested approach

- Before lectures
 - read textbook; find questions
- Attend lectures
 - take notes; ask questions!
- After lectures
 - read textbook; explore further
 - write assignments (start early!)
 - get help and help others (chat room, forum, wiki)
- Do attend tutorials

Common *mistakes*

- “Slides are already online”
 - Lectures are much more than just browsing slides
 - Pay attention to in-class questions/discussion too!
- “Slides are too brief”
 - Slides are just guidelines to navigate/understand
 - Take notes in class and read the textbook!
- “Start to do assignments on the due date”
 - Simple fact: you cannot finish, or even start, them
 - Start early and let us know if you have questions!

More systems courses

- Computer networks (CSc 361)
 - suggestion: don't take them together
- Advanced computer networks (CSc 466)
- Advanced communication networks (CSc 467)
- Wireless and mobile networks (CSc 463)
- Network management and security (topics/DS)
- Embedded systems (CSc 460)
- Multimedia systems (CSc 461)
- Distributed systems (CSc 462)

Your feedback

- Teaching/learning is interactive
 - two-way communications
- Let me know
 - what you think about lectures, assignments, tutorials, exams, topics, ...
 - what you want to know more or probe further
- You can *always* reach me
 - in class, during office hours, by email/phone

Course Rep

- Please volunteer yourself!
 - Anonymize
 - Aggregate
 - Amplify
- Feedback to the teaching team
 - lecture and lab instructors, markers
 - once a month face-to-face
 - electronically all the time

Course policies

- See official course outline
 - late assignments, mark appeals, etc
 - academic integrity
 - zero tolerance on cheating!
 - accommodation, etc
- No group assignment/project
 - collaboration/participation is encouraged
 - responsibility: your submitted work is yours
 - obligation: give credits to references

Assignment 0

- Due on Friday, May 8th, 2015
- Send an email to pan@uvic.ca
 - From *you@uvic.ca* (or *you@csc.uvic.ca*)
 - Subject: [csc360] A0
 - name, student number, academic program
 - things you already known in OS
 - things you want to know in OS
 - issues with course logistics, willing to be the rep,...
 - a URL to your mug shot
 - let me know you! e.g., reference letters

Need more challenge?

- ACM International Collegiate Programming Contest (ICPC)
 - ↳ UVic has been participating in the last few years
 - ↳ <http://www.csc.uvic.ca/icpc>
- BCNET Broadband Innovation Challenge, or DMC
 - ↳ any applications running over broadband networks
 - ↳ previous winning projects (some from UVic)
 - <http://tinyurl.com/bcnetbic>
- Other student competitions
 - ↳ <http://www.csc.uvic.ca/competitions>

This lecture

- An introduction to the course
 - who, when, where, what
 - course materials
 - course objectives
 - course topics
 - you and the course

Next lecture

- An overview on operating systems
 - read OSC Chapter 1