CMPT 300 D100
Operating System I
(Course Introduction)

Dr. Hazra Imran

Summer 2022

Today's agenda

- What will you do in this course?
- What is an operating system and why do we need one?
- Why study operating systems?
- What will be our course policies?

Teaching team

Instructor

Dr. Hazra Imran
himran@sfu.ca

TAs

AmirMohammad Deilami ada135@sfu.ca

Mugilan Mariappan mmariapp@sfu.ca

Dhruv Patel dnp5@sfu.ca

What is an Operating System?

A program that acts as an intermediary between a “user” of a computer and the computer hardware

…a ***User*** can really be a person, an application program or another computer

“Operating systems are among the most complex pieces of software yet developed”, **William Stallings, 1994**

OS is everywhere

Credit : Google Images

History of OS

Early Operating Systems

- One application at a time
- Had complete control of hardware
- OS was runtime library

Batch systems

- Keep CPU busy by having a queue of jobs
- OS would load next job while current one runs
- Users would submit jobs, and wait, and wait…

History of OS

Time sharing OS

- Multiple users on computer at same time
- Multiprogramming: run multiple programs at same time
- Try to complete everyone's tasks quickly

Nowadays Operating Systems

- Smartphones
- Embedded systems
- Laptops/ Tablets
- Virtual machines

Tomorrow's Operating Systems

- Giant-scale data centers
- Increasing numbers of processors per computer
- Very large scale storage

Why CMPT 300 is important for you?

- Integral part of computer science
- OS is the point where hardware, software, programming languages, data structures, and algorithms all come together
- Useful to know how computers operate in order to design efficient and complex systems
- Increasing need for specialized operating systems.
  - e.g. embedded operating systems for devices - cell phones, sensors and controllers
- Worth 3 credits

Topics Covered

Compute:

- Processes and Threads
- CPU Scheduling
- Synchronization
- Deadlocks

Memory:

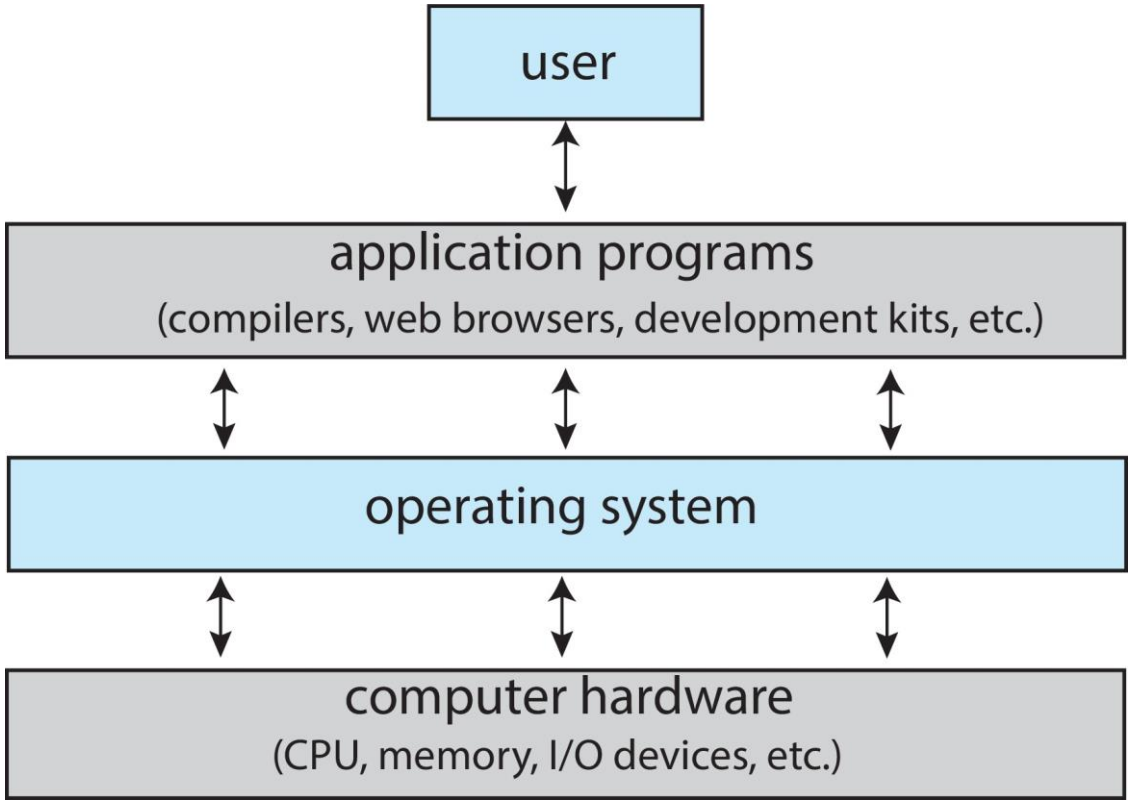
- Main Memory
- Virtual Memory

Storage:

- File Systems

Might be adjusted as we go

Big Picture: Computer Systems



Course Learning Goals

By the end of the course, you should be able to:

- understand the functions and structure of operating systems
- simulate and compare common scheduling algorithms used by operating systems
- apply synchronization and deadlock avoidance methods to writing robust concurrent software
- implement simple Linux kernel extensions (modules)
- describe techniques and common technologies used for virtualization
- manipulate files and directories within the operating system

Course Details

- Lectures:
  - Cover basic notions and concepts known to be hard
  - Most lectures will include a set of learning goals
  - Pre-lecture notes available on the course website before the lecture
  - Note that pre-lecture slides are subject to change and are only posted to help you to take notes during the lecture. Use **post lecture** slides for studying
  - Some lectures will include Clicker questions or in-class exercises.

- We'll use **Canvas** (assignment, content submissions and grades) and **Piazza** (discussion board).

Piazza

- We’ ll be using Piazza for posting announcements and additional course-related material.

- Check course outline for Piazza link and access code.

Grading Scheme (Assessments)

- 50% for the Assignments (4)
  - 2% for 04 reflection surveys
  - 5% Assignment 1
  - 15% Assignment 2
  - 18% Assignment 3
  - 10% Assignment 4
- 15% for Quizzes (9)
- 4% In-class (take home) activities (5-6)
- 16% for Midterm Exams (2)
- 15% for the Final Exam

To pass the course, you must pass the final exam and achieve an overall grade of 50% or better.

Assignments

- 04 Programming Assignments
- The assignments are designed to exercise your algorithm and low-level programming skills.
  - Can be done individually or in team of two.
  - You should **not** simply divide the assignment; this is meant to be a learning exercise.
- Assignments are mostly back-to-back – start early
- late days: Maximum of **four** late/slip days.
  - No credit for late submissions if you are out of slip days.
  - If you don't claim late days, **you lose 20% per day**
  - You may use up to two of your late days on any assignment
  - Use with care!

Clicker

What kind of C programmer are you?

(a) An experienced (> 250 lines of code, pointers, malloc, multiple modules)

(b) Competent (50-250 lines of code, makefiles, 1-2 modules)

(c) Novice (10-50 line program)

(d) What in the world is C programming?

Quizzes/ In-class activities/Take-home activities

There will be quizzes and In class activities taken from the lecture and textbook. There will be **no make-up** for these activities.

In class activities/ Take-home

- marks will be based on participation only.
- There is **NO** set schedule for these activities (Please no emails asking for the schedule)

- **Two Midterm exams** – June 20 and July 25

- **Final exam - TBA**

How to get help?

Piazza Discussion Board (CHECK IT OFTEN)

- Post questions on course material
- Answer others' questions if you know the answer
- Learn from others' questions and answers
- Please **search before you post**

Go to office hours (Piazza is NOT a good substitute for this)

- General purpose OH
- Programming OH
- OH starts from week 2
  - https://canvas.sfu.ca/courses/70193/quizzes/210916?module_item_id=2510056

A few useful tips

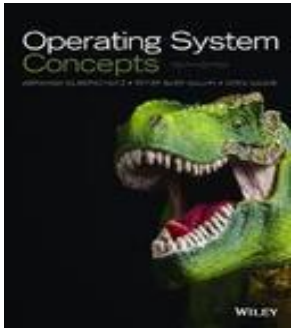
- I like answers that are: **correct**, **easy to read**, and **as short as possible while including important information**.

- I value **conceptual understanding** over memorization.
  - If you prioritize understanding over grades, then you increase the likelihood of getting both.
  - If you prioritize grades over understanding, then you increase the likelihood of getting neither.

Textbook

Silberschatz, Galvin and Gagne (2018) Operating Systems Concepts, Tenth Edition, ISBN-13: 9781119320913

- You are responsible for all the material in the assigned readings, regardless of whether it has been explicitly covered in class.
- You are also responsible for all the material covered in class, whether or not it is included in the readings/available online.
- It is strongly recommended that you read the assigned readings before each class. It will help you understand the material better when I lecture.



Inclusive learning environment

- Create an environment where everyone can learn and thrive
- Always feel free to ask a question!
- Create a climate where we treat everyone with respect

The Collaboration/Cheating Policy:

In general, you can collaborate as much as you want with whomever you want with two restrictions:

- You must acknowledge everyone you collaborated with.
- You may not take any record away from the collaboration.

The exceptions are:

- On the assignments you may collaborate with your partner.
- You may *not* collaborate on the midterms unless explicitly stated.

For code

- Use private repos (no solution sharing!!!!)

Academic Misconduct

- **When you are in doubt whether the line is crossed:**
  - Talk to me or the TAs

- See **SFU official regulations** on what constitutes plagiarism (pointer in course syllabus)

- Ignorance of the rules will not be a sufficient excuse for breaking them

- Any unjustified cases will be reported to the department (in addition to penalties on the course syllabus)

Time Commitment

- Time spent in the course $\propto$ grade
- Start early. You don't know how long it will take
- Plan before coding
- Should rethink taking this class while taking another heavy class

Note: It's a heavy load course

To be successful in CMPT 300

- Attend lectures
- Be ready to hunt down materials on your own
- Ask questions, post comments on Piazza
- Do assignments
- Start early and be honest
- Study for exams

The course is quite straightforward – If you do the work, you will do well.

Note: It's a heavy load course

From Spring 2022 students …

Be sure to start the assignments early. They are quite doable and generally interesting; however they can be very time consuming.

Don't be like me and give up reading the textbook 2 weeks into the course. The textbook, although lengthy, has a lot of useful information. Start early on all readings and assignments and you will do well!

Hazra is a very understanding and helpful teacher, so are the TA's. So if you feel stuck, just ask for help!

start the assignments early on, especially Assignment 3.do not miss any of the quizzes and in class activity, they can prove to be a potential grade booster in the end.

Assignments are really important, do them carefully. Remember to handle corner cases. Understand terms and concepts instead of memorizing. Explain those terms to yourself after a certain amount of time. Also ask yourself what is the differences between this one and that one, and what is the similarities.

Set an alarm or calendar entry for quiz deadline so that you don't forget in the event of really busy circumstances

Make sure you understand how the navigate the C documentation and the Linux man pages. It will save you a lot of time during the lab components and further your understanding of the language.

What to do next?

1. Decide whether this is the right course for you

2. Join the Piazza class

3. Check out the syllabus, and tentative schedule on Canvas

4. Complete the surveys available on Canvas
   - Getting to know you better
   - Office hour availabilities

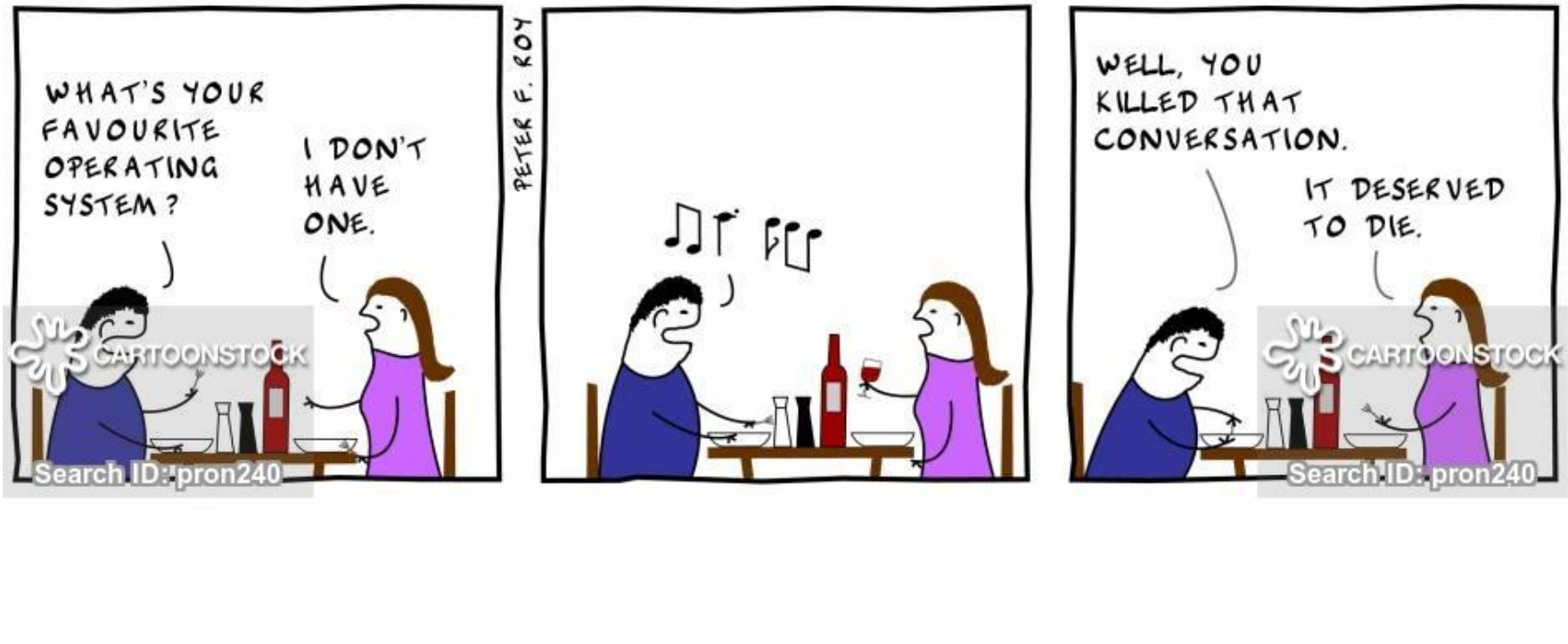
Due: May 12, 11:59 PM

5. Tutorial 1: Set up the environment (Detailed instructions are available on Canvas)

And Now . . .

That's It! Enjoy the course!

Any questions?



Credit: https://www.cartoonstock.com/directory/o/operating_system.asp