Graduate Operating Systems
CSE 60641
(Welcome & Administrative Items)

Fall 2020

Course Overview

- Instructor:
  - Christian Poellabauer (cpoellab@nd.edu)
- Course Meetings
  - TR 9:35 – 10:50
  - McCourtney Hall B01 (& Zoom)
- Office Hours
  - Primarily Zoom Office Hours (in-person only by request)
    - Times TBD (Doodle Poll); probably 2-3 slots
  - Zoom# will be on website, Sakai, etc.
  - Office hours will be "public", but you can request 1-1 meetings

Course Overview

- Primary components:
  - Read and discuss papers (homeworks, exams)
  - Work on a mini research project (reports, papers, presentations, software, review)
  - "Grad Student 101"
    - Note that department also offers "Research Methods" now!

Goals of Course

- (Operating) systems research
- Reading, reviewing, critiquing research literature
- Conduct (a little bit) of systems research (including paper writing process)
- Satisfy core requirement & pass the qualifying exam
- Learn about "life as a grad student & researcher"

OS Research Literature

- How has the role of the OS evolved?
- What are key principles for OS?
- How did past influence present?
- What are current trends and what will the future bring?

Reading/Critiquing Papers

- Read many papers
- Discuss papers, methodology, problems they address, solutions they propose, etc.
- Determine what makes a good research paper
- Typically discuss 1-2 papers per lecture

Papers and Discussions

- Classical/important/recent papers
- Papers that demonstrate excellence in research
- Papers that demonstrate how the field is changing
- Be willing to question the paper
- Be willing to take a position
- Be willing to be wrong
- Understand that there is not always a “right” or “wrong” answer

Course Organization

- In-person lectures for students on campus
- Live streaming of lectures via Zoom for students sick, in quarantine, off campus, etc.
- Asynchronous lectures via recorded Zoom for students in off campus in different time zones
  - Zoom office hour attendance mandatory (1x per week minimum)

Course Organization

- For every lecture, read 1-2 papers as shown in the online schedule
- Submit (via Sakai) a brief (1 paragraph) summary of reading assignments PLUS answer potential questions; due by 9.35am EST of the day of the lecture!

Summary Example

Paper 1: First Author, Second Author, and Third Author, "This is the title of the paper", Cool Journal, volume 2, number 3, July 2019.

Summary: This paper proposes a novel scheduling algorithm that adjusts the CPU cycles allocated to a process based on the predicted CPU and I/O loads of all processes in the runqueue. Specifically, the authors use a game-theoretic approach that considers recent utilization history, past process traces, resource availability, and other factors to make these predictions. The paper also proposes a novel mechanism to "donate" resources between processes if the processes have certain dependencies. The evaluation section compares the proposed approach to another recent predictive scheduler that utilizes a neural network based approach, showing that the game theory based approach on average predicts future resource utilization 18% more accurately. The authors identify several shortcomings of their approach and propose to investigate multi-processor systems in their future work. *A strength of the proposed work is that the proposed approach is computationally much more efficient than prior solutions. However, I believe that the authors made several simplifications that make the results somewhat questionable. Specifically, they assume that all processes ….*

Question 1: In this paper, process dependencies are automatically detected by detecting calls to locking mechanisms at run-time.

Question 2: The resource donation process uses a mechanism based on shared memory.

What is **Research**?

- "Creative and systematic work undertaken to increase the stock of knowledge" [Wikipedia]
- Engineering vs Research
  - "I want to build a mouse trap"
    - This is **not** research!
  - **Research requires a question!**

Engineering vs Research

- "I want to build a mouse trap"
  - "Is my new mouse trap better at trapping mice than a conventional mouse trap?" (Why?)
  - "Are there common traits among the mice that are being captured (and the ones not being captured)?"
  - Are there characteristics (materials) that make better traps?
  - "Does habituation occur and how?"
  - If my mousetrap were invisible, would it be better?
  - How can we build an invisible mousetrap?
  - **Research requires a question!**
  - **Research often requires engineering!**

Engineering vs Research

- Engineering helps you answer the question
  - Create a prototype mousetrap
  - Build a framework in which to evaluate the efficacy of mousetraps
  - Designing experiments combines engineering and research
  - Conducting experiments is often engineering
  - Analyzing and interpreting the results is research

Rigor & Reproducibility

- **Rigor:** strict application of the scientific method to ensure unbiased and well-controlled experimental design, methodology, analysis, interpretation and reporting of results
- **Reproducibility:** ability of a study or experiment to be reproduced (by somebody else)

Diligent Research

- Write down everything
- Understand the data
- Question yourself constantly
- Remind yourself of the question you are asking
- Keep (publish) the data
- Avoid bias
  - Introduction of systematic error
- Be careful (ethical) using & interpreting data
- Discuss your work with others (share data, paper drafts, etc.)
- **Know the literature!**

Research

- There are not necessarily any right answers
- No one can tell you with certainty that you are right
- You are never really done
- Understanding (and working with) large systems is difficult

Examples of Research Approaches

- Form a hypothesis
- Measure a real system (trace data)
- Instrument existing systems (and measure again)
- Run simulations
- Analytical investigation of collected data
- Micro vs macro investigations
- Draw conclusions
- Compare results against others' results
- Use results to form new hypotheses

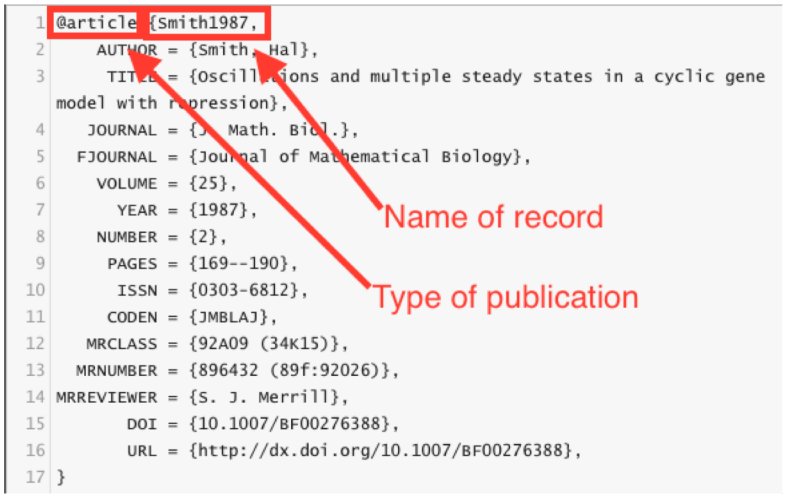
Research Papers

- Big idea papers, unifying themes, small ideas with evaluation, measurements, comparisons, retrospective or experience papers, …
- Keep track of important/relevant/good papers in your field (bibliography, bib file, etc.)

Research Papers: LaTeX

- High-quality typesetting system
- De facto standard for the communication and publication of scientific documents
- www.overleaf.com

Research Papers: bibtex



Research Papers: Critiquing

- Is the problem well described/motivated?
- Does the idea make sense?
- Does the paper make a difference?
- What is being measured/proven/demonstrated?
- Are the measurements (experimental setup) meaningful?
- Are the results meaningful?

Research Papers: Critiquing

- Summarize paper in a few sentences
- Put papers in categories (e.g., classic, important, useless, …)
- Is the paper well-written?
- What did you learn from the paper?
- How would you have conducted the research?
- Does the paper suggest any future work?

Writing a Paper

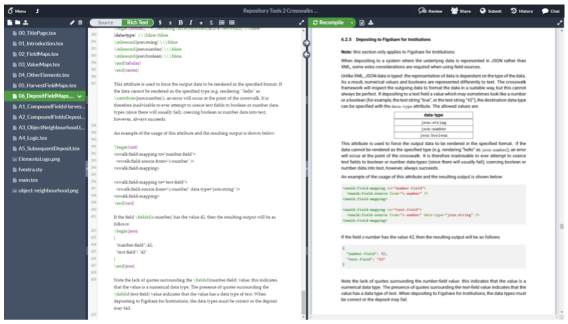
- Abstract: introduce area, state problem, explain approach, summarize conclusions
- Introduction: describe problem, importance, approach and contributions, road map
- Background: anything reader needs to know
- Approach/Solution: what you did
- Results: experimental setup, explain expected results, surprising results
- Related Work: relate your work to prior efforts
- Conclusions (and future work)

Writing a Paper

- SPELL CHECK!!!!!!!!!!!!!!!
- Learn grammar, style, etc., adapt to your field/advisor/community/…
- Read and critique your own work!!! Are you satisfied? If you know there is a problem, a reviewer will find it too
- Write while you do the work; keep track of all you do; safely store data!

Writing a Paper

- LaTeX
  - Recommendation: Overleaf/Sharelatex
- Microsoft Word
  - Recommendation: Google Docs



“Entrance Exam”

- On the following slides, you’ll find the “entrance exam”
- Try to answer questions on your own by next time (**you do not have to submit anything and this is NOT graded!**)
- Based on how you perform:
  - May want to wait a year and take **ugrad OS** first if you have no background whatsoever in operating systems or a related field
  - **Be willing to learn (some) basic OS concepts in parallel** (will require proactiveness, independence, time, …)

"Entrance Exam"

- What is a **multi-threaded** process?
- What is the purpose of **mutual exclusion**?
- What does it mean to say an operation is **atomic**?
- Use a brief example to describe what a **deadlock** is or how it can be caused.
- What is the difference between **deadlock** and **starvation**?

"Entrance Exam"

- What is the purpose of an **interrupt**?
- What is **priority inversion**?
- What does a **page table** do?
- What does **thrashing** mean?
- What is a **symbolic link**?
- What is a **parity bit**?
- What is an **i-node** (or **file control block**)?

“Entrance Exam”

- What does it mean to **fork** a process?
- What is the danger of **caching a write**?
- What is a **page fault**?
- What is the difference between **kernel space** and **user space**?
- What is **disk fragmentation**?
- What is a **critical section**?

“Entrance Exam”

- What is a **runqueue** (or **ready queue**)?
- What is a **binary semaphore**?
- What is the difference between a **direct pointer** and an **indirect pointer** in a file system such as EXT?
- Can you name and very briefly describe a **scheduling algorithm** that would be **fair** to all tasks awaiting execution?

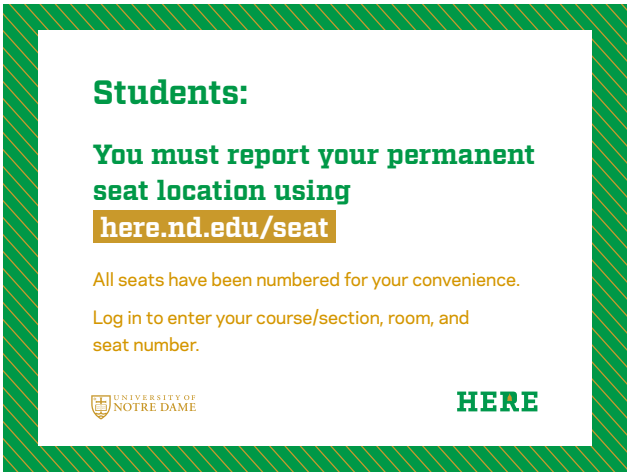
"Entrance Exam"

- Can you name and very briefly describe a **scheduling algorithm** that might be a good choice in a **real-time system**?
- What is a **system call**?
- What does it mean for a system call to **block**?

Next Lecture & Next Week

- Thursday:
  - Introduction
  - Revisit "entrance exam"
- Next week:
  - **OS History and Architecture**
    - **[1]** P. Brinch Hansen, "The Nucleus of a Multiprogramming System", Communications of the ACM, 238-242, April 1970.
    - **[2]** Dennis M. Ritchie and Ken Thompson, "The UNIX Time-Sharing System", Communications of the ACM, volume 17, number 7, July 1974.
    - **[3]** Dawson R. Engler, M. Frans Kaashoek, and James O'Toole Jr., "Exokernel: An Operating System Architecture for Application-Level Resource Management", Proc. of the 15th Symposium on Operating Systems Principles, December 1996.

Reminder



Reminder