CSE 542: Operating Systems

http://cse.nd.edu/courses/cse542/www/

- Instructor: **Surendar** Chandra (surendar@nd.edu)
  - Office Hours: Thu: 2:00-3:00 Fri: 3:00-4:00 (other times, by email appt)
  - Email is the best way to reach me, I am also logged into AIM: surendar (Apple iChat)

- TA: William Acosta
- Listserv: cse542-01-fa04@listserv.nd.edu
  - Send questions to the list so that everyone can follow along

Introductions

- Tell us your name, academic background, research interests
- What do you hope to achieve in this class
- Machines that you own, OS you have worked with
- Prior OS experience

Outline for today

- Course policies:
  - Course organization and expectation
  - Grading policy, late policy, reevaluation policy
  - Academic honesty
- Chapter review: 1,2,3

Course goals

- Introduce fundamental graduate level topics in OS
  - Course text for reviews + read and discuss "seminal" papers
  - Not designed to introduce latest OS research - papers may be old
    - Good venues: ACM: SOSP, ASPLOS, TOCS, USENIX: OSDI, Systems seminar (Wed 3:30pm)
  - Course project designed to give you hands on experience
    - Identify a problem and motivate others why everyone should care about this particular problem
    - Develop your solution strategy and argue why it would be practical to implement
    - Perform experiments to verify your hypothesis
    - Argue how your experiments prove your point

Course Organization

I expect that you have some prior knowledge of operating system/using operating systems. The text book should help review basics.

You will read the assigned paper/book chapters before class

I **encourage open discussion about the technologies**

The course expects significant programming to achieve pedantic goals. However, this is not a course to test your programming skills. To the extent possible, I am language/operating systems/project details agnostic.

Systems philosophy

1. Results from your own implementation

2. Results from your own simulation

3. Implementation results from a paper

4. Simulation results from a paper

5. Everything else (hearsay, rumors, “I think so”, “I think that it is how it should work” etc)

Grade distribution

- Class participation and paper evaluation–mandatory
  - Each paper will have a shepard (to summarize), one proponent and opponent. Each give their view in ~5 minutes. I will pick these people before class. No futzing. You are allowed to skip two of these.
- Midterm and Finals – 30% (open book, notes)
- Two take home quarter-term assignments – 20%
- Course project – groups of two - 50%
  - Option 1: 6 small home work project (week+2 days) (24%) + course project (26%)
  - Option 2: course project (50%)

Midterm and Final

- Open book, open notes, in class exam

Home work assignment

- Written, take home assignment to help you prepare for the exams
- Individual effort.
- May have to write code to answer some questions.

Option 1.1: Small homework projects

- You will be assigned six programming projects
- Project groups of up to two - do not partner with your “clone”.
- These projects are designed to quickly bring you upto speed (more in the senior level)

- Suggested variations of OS: FreeBSD or other modern OSs

Course Project

- Teams of 2 students. Start thinking about what you want to do/partners etc. now.
- Individual grades influenced by the final oral examination and presentation.
- I will provide a list of project ideas. I prefer projects that make sense to you (research, other course etc.)
- You will explore a substantial topic and show its validity with an implementation
- At the end of the semester, there will be a mini-symposium where you will publicly present your work. Best paper and presentation will be recognized.
- There will be an one-on-one oral interview

Project report

- Project report should be electronically turned in with a succinct and anonymous report for double blind reviews (6 readable pages - you can use ACM, USENIX or IEEE style) on your implementation strategy and what you learned
- I will assign the project reports to your peers for review. You will review the papers in a similar format to a regular conference. You will be graded on the quality of your reviews.
- Me and the TA will provide another review. We will grade you on the quality of the work
- You do not need to submit the code for the project unless there is some dispute about the validity of the results.

Resources

- Cushing 208 lab
  - 6 Pentium desktops with 512 MB main memory
  - Miscellaneous: Cerfcube, iPAQ, laptops etc.
  - HP Grant:
    - 1 - 2x1.5 GHz Itanium2 with 8 GB of memory ($40K)
    - 1 - 4x1.5 GHz Itanium2 with 8 GB of memory ($80K)
  - All these machines are exclusively for this class
  - Itaniums are waiting for power supply (they use a funny 208V connector)
  - Try out whatever you want to one these machines, it is okay to break the software on these machines (but not the hardware, don't drop the machines!!)

Reevaluation policy

- Arithmetic errors, missed grading will be reevaluated.
- I encourage you to discuss concerns with your solution with me
- I discourage re-evaluation of partial credits:
  - Football penalty policy:

    If you think you deserve a better partial grade, write down the reason why you think that you deserve a better grade and how many extra points you think you deserve. If I agree, you could get up to this many extra points. If I disagree, you will lose this much points.

Late policy

- None – Projects/homework/critiques are due at 11:00 am (right before the beginning of class). **I do not accept late submissions** (not even a second)

- For home work submissions, I will use the clock in wizard.cse.nd.edu for reference

- Please contact me regarding unforeseen emergencies

Academic Honesty

- Freedom of information rule:
  - Collaboration is acceptable
  - To assure that all collaboration is on the level, **you must always write the name(s) of your collaborators on your assignment.** Failure to adequately acknowledge your contributors is at best a lapse of professional etiquette, and at worst it is plagiarism. Plagiarism is a form of cheating.

Academic Honesty – Gilligans Island Rule

- This rule says that you are free to meet with fellow students(s) and discuss assignments with them. Writing on a board or shared piece of paper is acceptable during the meeting; however, you may **not take any written (electronic or otherwise) record away from the meeting**. This applies when the assignment is supposed to be an individual effort. After the meeting, engage in half hour of mind-numbing activity (like watching an episode of Gilligan's Island), before starting to work on the assignment. This will assure that you are able to reconstruct what you learned from the meeting, by yourself, using your own brain.

How to Read a Research Paper

- Typical paper
  - Abstract
  - Introduction
  - Motivation, problem description
  - Research questions that are being addressed by this paper
  - Experiment Setup
  - Results
  - Conclusions and Future work

Why do you read a paper?

- Understand and learn new contributions

- However
  - Not all papers are “good”
  - Not all papers are “interesting”
  - Not all papers are “worthwhile” for you

- You have to learn to identify a good paper and spend your time wisely
  1. Breadth
  2. Depth
  3. React

How to Read a Research Paper

- Ask yourself, what is this paper about? (breadth)
  - Read the title and the abstract

    If you still don't know what this paper is about, then this is a bad paper.
  - Read the conclusion

    Are you now sure you know what this paper is about? If not it is a BAD paper. We will try not to read such papers in this course
- Read the introduction
- Read the section headings
- Read tables and graphs and captions. See what they say

How to read a paper (cont)

- See who wrote it, where it was published, when was it written (credibility)
- Skim bibliography to see if the authors are aware of relevant related work. See if you know the relevant work. See if you know a relevant work that they didn't refer

How to read a paper - depth

- Approach with scientific skepticism
- Examine the assumptions
  - Do their results rely on any assumptions about trends in environments?
  - Are these assumptions reasonable?
    - E.g. “Lets assume that there are billions of powerful computers, connected by a high speed network, spread across the world, our system will …”
    - E.g. “Our system can enable you to run Windows 98 on a 33Mhz Intel 386 with 640K main memory”

How to read a paper - depth

- Examine the methods:
  - Did they measure what they claim?
  - Can they explain what they observed?
    - It is easy to dump your experimental results on the paper. As a reader you want an analysis of why the system behaves a certain way, not the raw data
      - This is why your final reports will be 10 pages long
  - Did they have adequate controls
  - Were tests carried out in a standard way? Were the performance metrics standard? If not, do they explain their metrics clearly?

How to read a paper - depth

- Examine the statistics: (there is truth, lies and then there is statistics!!)
  - Were appropriate statistical tests applied properly?
  - Did they do proper error analysis?
  - Are the results statistically significant?
    - Common mistake: “We performed our experiment once at 4 am and noticed a ten fold improvement. Thus we conclude that our system is better”
  - Be very careful with percentages
    - Method A: 0.01 seconds, our Method: 0.005 seconds
    - Our method shows 100% improvement over method A!!

How to read a paper - depth

- Examine the conclusions:
  - Do the conclusions follow logically from the experiments
    - We performed our experiments with 8 palm pilots and saw a 10 fold improvement. Hence we conclude that our system will scale to millions of palm pilots
  - What other explanations are there for the observed effects
  - What other conclusions or correlations are there in the data that they did not point out
    - Earlier work performed experiments using a 2 Mbit wireless network. Our system (incidentally) used a 11 Mbit network and saw a 5 fold improvement. So our technique works!!

How to read a paper - react

- Take notes
- Highlight major points
- React to the points in the paper
  - Place this work with your own experience
  - If you doubt a statement, note your objection
  - If you find a pleasing quotation, write it down
- Construct your own example
- Summarize what you read

- Maintain your own bibliography of all papers that you ever read

Sample bibliography - bibtex

```
@Book{stevens98,
        author = {W. Richard Stevens},
        title = {UNIX Network Programming:
                Networking APIs: Sockets and XTI},
        publisher = {Prentice Hall},
        year = 1998,
        volume = 1,
        series = {ISBN 0-13-490012-X},
       note = {Sample code from this book is available at

  ¥url{http://www.kohala.com/start/unpv12e.html}},
        edition = 2,
}
```

…. You can refer to the Computer Network books by W. Richard Stevens ¥cite{stevens98} for sample ….

How to Write a Research Paper

- Write it such that anyone who reads it using the method we discussed understands your ideas.
- Clearly explain what problem you are solving, why it is interesting and how your solution solves this interesting problem
- Be crisp. Explain what your contributions are, what your ideas are and what are others' ideas