EECE 571P: Program Analysis and Optimization

Lecture 0 (“The Orientation”)

Who am I ?

- Assistant Professor of Electrical and Computer Engineering (ECE)– joined UBC in 2010
  - PhD from UIUC, Post-doc with MS Research
  - Research in fault-tolerant and secure computing
  - Use static analysis techniques in our research
- First time I'm teaching this course (EECE571P)
  - Taught grad course in fault-tolerance (EECE 513)
  - Teach undergrad courses on software design/arch.

Who are you ?

- Your name, department, advisor (if applicable)
- What interests you ? What do you want to do after you finish your current degree ?
- What made you choose this class (or did you) ? What do you want out of the course ?

What is this course about ?

- Advanced compilers, static analysis techniques
- Methods to analyze, optimize, transform code
- Applications of static analysis techniques in reliability, security, parallelism, real-time etc.
  - But this course is not about any of these things …
  - I will tailor this aspect of the course to the class

Pre-requisites

- NO compiler experience necessary
  - An undergrad compiler course is not necessary
- But need algorithmic/mathematical maturity
  - We will deal a lot with algorithmic ideas
  - Willingness to learn more important
- Need programming experience with C/C++
  - Needed for hacking LLVM, in assignments/project

Why should you take this course ?

- Your advisor or supervisor asked you to ☺
- You get to learn the conceptual underpinings of static analysis tools and hack lots of cool code
- Compiler engineers are much sought after in industry – and among the hardest to find
- Opportunity to apply static program analysis in your research – can integrate with course project

OK, I'm sold. What should I do ?

- Show up for the lectures – participate in class
- Submit periodic reviews of assigned papers. Lead discussion (likely) in your research area.
- Complete two assignments based on LLVM. Propose and complete a project of your choice applying static analysis to substantial problem.

Evaluation

- Class Participation 10%
- Reviews and discussion leading 20%
- Assignments 20%
- Project 50%

Class Structure

- **Lectures and paper readings (30 %)**
  - I will lecture roughly every 3 out of 4 classes
  - We will discuss papers in other classes (you need to submit paper reviews ahead of time – counts for 15%)
  - Discussion leading counts for 5% of grade
  - Class participation counts for 10 % of grade

- **Assignments (20 %)**
  - Two assignments each counting for 10%
  - Due in late Sept, late Oct respectively

Class Project

- **Major component of course grade (50 %)**
- **To be done in teams of 2 (3 people allowed in a team if warranted – discuss with me first)**
- **You are encouraged to integrate it with your own research, but this is not necessary**
  - But cannot be the same thing you do for research
  - Talk to your advisor first to ensure this is OK

Final project: Milestones

- In a week or two from today (by Sep 22nd)
  - Decide who you want to work with for the project
- By end of September: 2-page proposal (5%)
- By end of October: mid-term report (10%)
- By early December: Presentation (10%)
- By December 15th -> final report + code (25%)
  - Must be in the form of a conference paper (10 pages)
  - Implementation using LLVM infrastructure

Paper Readings

- **We will read papers on applications of static analysis techniques to different areas**
  - Each of you needs to submit a 1-2 page review of the papers by noon the day before the discussion
  - All reviews will be open to everyone after then.
  - Discussion leader summarizes each paper (5 mins), and the points raised by the class along with his/her own points for leading the discussion.
  - **You are allowed to miss one review penalty-free**

Assignments

- Two assignments both based on the LLVM compiler infrastructure (llvm.org)
- **Assignment 0:** Install and compile LLVM on your machine (not graded) – due by Sept 17th
- **Assignment 1:** Write a simple analysis pass in LLVM (10% of grade) – due by end of Sept.
- **Assignment 2:** Write a simple transformation pass in LLVM (10% of grade) – due by 30th Oct

Policies etc.

- All homeworks are due at midnight of due date (submission by email – auto acknowledgement)
- Late submissions will be penalized by 2% for every day they are late up to 3 days maximum
- Please review UBC's policy on plagiarism and academic conduct. Ignorance of the same will not be an acceptable excuse for lapses.
- You are expected to attend all lectures and paper discussions (absences need to be explained)

To do by next class …

- Join Piazza.com and register for this class
  - No email will be sent/No vista or connect either
  - All announcements, assignments, lectures will be uploaded on Piazza
  - Participation on Piazza counts for class participation – both asking and answering questions
- Complete the questionaire that I've given you
  - Allows me to tailor the course to your interests

Final thoughts

- The goal is to have fun while we learn !
  - I certainly hope you have fun in this course …
  - I hope to have fun learning from you too …

- I am always open to suggestions and critical comments on any aspect. **Such critical comments will not impact your grade in any way.** On the other hand, good suggestions may even earn you some extra credit/cookies