CS590

Advanced Software Testing and Debugging

Xiangyu Zhang

Why Testing and Debugging?

- One of the most prominent challenges for IT.
  - Software bugs cost the U.S. economy about $59.5 billion each year (0.6% of the GDP) [NIST 02].
  - Companies.
- Security is becoming a necessity.
  - The worldwide economic loss caused by all forms of overt attacks is $226 billion. [CRS 03].
- Software defects make programming so painful.
  - 50% of development efforts are on testing and debugging.
- Automation is still a dream
  - Automated testing?
  - Automated debugging?
- Opportunities

Course Objectives and Styles

- Get to know the state of the art
- Get familiar with various tools
- Get experience on research
  - Projects
  - Presentations
  - Essays and reports.
- Generate papers.
  - Last time I offered, 4 first-tier conference papers were produced.
- Style
  - Challenging, informative and fun.

How to Make it Challenging, Informative and Fun

- No exams and no textbook assignments.
- Projects
  - Project one: given an open source program, generate test cases to achieve the best coverage (competition)
  - Project two: (three options)
    - Option 1: given a real world buggy program, its correct binary, and a test suite, identify and fix the bug.
    - Option 2: given a real world program with a vulnerability, please find the vulnerability and craft a test case to expose it.
    - Option 3: given a real world program with a known concurrency bug, (and possibly the failure core dump), deterministically reproduce the failure.
  - Final Project.

How to Make it Challenging, Informative and Fun

- Training on paper writing
  - A writing competition.
  - Closely harness the project proposal and report writing.
- Getting familiar with academic life.
  - Simulating a mini-PC meeting.
    - Paper review, writing review, and pc discussion.
- Real research challenges
  - Show your creativity.
- Any other ideas to make it more fun?

Grading

- Project: 60%
  - The first small project is a testing competition. Prizes to be decided. (10%)
  - The second small project is about debugging. (10%)
  - Final project proposal (5%)
  - Final project (30%)
  - Final project report (5%)
    - Students will work independently on small projects. Working in groups for the final project is allowed.
- Two paper presentations: (20%)
- Problem solving and paper writing exercise: (20%)
  - Program committee simulation, including paper review and discussion. (10%)
  - Problem solving challenges will be given regularly (10%)
    - One writing competition, Prizes to be decided.

Schedule

- Lectures for the first five weeks.
  - Concepts and principles.
- Student presentations
  - One presentation per class, the presenter needs to prepare one hour presentation, and lead 15 minutes discussion.
  - I reserve the right of asking questions regarding the papers.
  - Possible guest presentation.

Lecture Topics

- Program Representations
- Tracing and Profiling
  - Reading: Efficient Path Profiling
- Slicing (static, dynamic, and relevant slicing)
- Testing
  - Topics in testing
  - Combinatorial test generation
  - Dynamic test generation
  - Concurrency testing
- Debugging
  - Statistical debugging
  - Delta debugging
  - Comparison-based debugging
  - Debugging concurrent programs

Pre-requisite

- Interest
- Familiar with Linux
  - Install and use tools
  - Programming
- Some concepts in compilers