Com S 417
Software Testing

Fall 2017 – Week 1, Lecture 1

Introduction to Software Testing

Tuesday & Thursday 12:40-2:00
Horticulture Hall (HORT) 0118

Who am I

- Digital and software engineer, teacher, technical writer
- Ph.D. student at ISU
  - Research interest: Theoretical foundations of Scrum, System Dynamics Simulations of Software Engineering.
- Parent (four children, ages 12 to 41)
- Avocations
  - Flying -- Private pilot (not current)
  - Woodworking (fine furniture)
  - Photography
- Native of Kansas, Displaced mountain resident, Des Moines resident.

Who am I (Formerly? Professionally?)

"Distinguished Advanced Research Software Engineer," Arris Group

"Director Software Engineering," Motorola Mobility

- Led conversion to Scrum
- Directed teams in Sweden, India, Russia, Argentina, Canada, and U.S.

CS lecturer University of Kansas

Assistant Professor, McPherson College, McPherson, KS

CEO and publisher, R & D Publishing

C/C++ Users Journal (Founding Editor)

Window/DOS Developers Journal

SysAdmin

Network Administrator

Author "Debugging C" (Que, 1985)

M.S.C.S. , University of Kansas, 1985

District Magistrate Judge, Kansas 4th Judicial District

Who are you?

What Department?

- Computer Science Major?
- Software Engineering Major?
- Computer Engineering Major?
- Other?

What Objective?

- Just Curious?
- Programmer?
- Research Scientist?

- Prior Programming?
  - Language?
  - Work?
  - School?
- Prerequisites?
  - See me for form!

What is this class about?

Survey of Software Testing as a sub-discipline.

The big Questions: Who, what, why, when, how?

- Why bother testing?
- What is testing … exactly?
- What tests should we perform?
    - How do we select them?
- How do we perform tests?
- When should we start?
- When should we stop?
- Who should do the testing?

Why Test?

To find and fix bugs

- Actually, testing just indicates a bug exists, it doesn’t find it or fix it.

To Improve Quality

- Testing might be part of a quality assurance program, but by itself, it just measures existing quality, doesn’t improve it.

To make certain the program is correct

- We have a proof (the halting theorem) that is impossible.

To save money

- Testing can be very expensive. How do we know it’s worth it?

Why not let the user’s find ‘em?

- Arianne V – loss of craft $500 million + 1 year.
- Therac-25 – Three verified deaths. Three others seriously injured. (See the wikipedia page.)
- Knight Capital Group $400 million trading bug
- AT&T switch upgrade: $60 million
- Intel Pentium FP Bug: $475 million

What kinds of damage?

Damage to Person

- costly decisions
- loss of security
- loss of privacy
- loss of reputation
- loss of time
- loss of life/health

Property Damage

- loss of use
- buildings, tools, vehicles, merchandise, etc.
- cash, financial instruments, insurance
- loss of information

The Unavoidable Dilemmas

- We can’t guarantee correctness.
- The test space is effectively infinite.
- More tests don’t guarantee less risk.
- Delayed release also carries a risk of loss.

Improving Test Cost Effectiveness

- Automation
- Eliminate redundant tests
- Use a mixture of test techniques
- Focus on high risk areas

Testing as Risk Managment

Ultimately, containing test costs means limiting what we test.

In the final analysis,
Testing is about balancing risk against testing cost.

Software Testing is Changing

- Hardware capabilities have made automation much more attractive.
- Agile methods have drawn attention to the benefits of
  - reducing separation between testers and developers.
  - Treating all tests as regression tests.
  - fully integrated devops systems.

Introduction To The Course

The “official” description

COM S 417. Software Testing. (Cross-listed with S E).
(3-0) Cr. 3. S.
Prereq: COM S 309; COM S 230 or CPR E 310; ENGL 250, SP CM 212.

Comprehensive study of software testing, principles, methodologies, management strategies and techniques. Test models, test design techniques (black box and white box testing techniques), test adequacy criteria, integration, regression, system testing methods, and software testing tools.

Staff

Instructor:

Robert Ward

Email: *robertw@iastate.edu*
Phone: TBD

Office: B05 Atanasoff

Office hours: TBD

Note: I live in Des Moines. Normally I will be on campus only Tues, Wed, Thu.

TAs:

Peng Sun

Email: *psun@iastate.edu*

Office: Pearson 145

Office hours: 9am & 11am Thursday/Wed & by appointment

Priyangika R Piyashinghe (Rumesh)

Email: rumesh@iastate.edu
Office: Pearson 145

Office hour: 10am Monday and Friday & by appointment

Slides based on course notes by Dr. Xiaoqiu Huang

Communication

- I prefer email to phone. If posted office hours aren’t convenient, email for an appointment.
- All course materials, project descriptions and other resources (not intentionally limited to classroom availability) will be posted to BlackBoard.

Text Book

- The required textbook is *Foundations of Software Testing (1st Edition),* by Aditya P. Mathur*.* This book is out of print. While there are a limited number of copies still in circulation, to assure that you have suitable access, we recommend you purchase the Kindle version.
- We will also use material from *Introduction to Software Testing*, by Ammann and Offutt, (1st Edition) and various supplemental materials to be posted to BlackBoard.
- For occasional use, you can access digital versions via the library. However, be warned: the library license is session limited. You cannot count on access.

Learning Objectives

- Students to be able to design tests to meet realistic coverage criteria.
- Students to be in a position to apply standard software testing techniques.

Note: This is a 400 level course with written and oral communication requirements.

What you can anticipate

- You will:
  - explore Junit relatively thoroughly. (All exercises will be in Java.)
  - experiment with web testing tools.
  - use code coverage tools to guide test selection.
  - use static analysis tools.
  - learn a *lot* of test-related vocabulary.
  - develop test plans.
  - learn about coverage criteria, input partitioning, and test selection/generation techniques.
  - research a wide range of testing topics and tools.

Assignments & Exams

- 5 Labs
- 2 Exams and 1 Final
- 1 Team Research project/report
- Occasional homework
- Occasional in-class quizes
- Weekly reading assignments

Tentative Schedule

Aug 31 – Lab 1 due. Basic Junit and RIPR.

Sept 14 – Lab 2 due. Measured code coverage, Table-driven tests.

Sept. 21 – Exam 1.

Sept 28 – Lab 3 due. Mocks and Oracles.

Oct 5 – Lab 4 due. Creating a Test Plan.

Oct 10 – Lab 5, part 1 due. Basic HTTP monitoring, Cactus Scripts.

Oct 26 – Exam 2.

Nov 2 – Teams and research project topics finalized.

Nov 9 – Lab 5, part 2 due. Performance testing for web apps.

Nov 16 – Dec 5 Student Research Presentations

Nov 20 – Nov 24. Thanksgiving Break.

Dec 4 – Dec 8. Stop Week.

Dec 11 – Dec 15. Finals Week.

Grading

Exam 1: 10%

Exam 2: 15%

Final Exam: 20%

Labs: 35%

Homework, in-class exercises and quizzes: 10%

Project report and presentation: 10%

Attendance is required during the student presentations.

To encourage regular attendance throughout the semester, homework assignments will be announced in class. To encourage timely completion of reading assignments (and attendance), I will periodically administer unannounced, in-class quizzes.

Note: Some work may be scored pass/fail.

What I expect

- You are here to become a professional.
- You will read assignments *before* the associated lecture.
- You will attend class.
- You will bring questions and observations and contribute to the class experience.
- You will build skills as well as acquire knowledge.
- You will behave with integrity.
- We will all treat one another with respect.

READ THE SYLLABUS

Academic Honesty

Students are expected to read, understand and comply with the ISU Student Conduct Code: “The academic work of all students must comply with all university policies on academic honesty. Examples of academic misconduct are:

A. Attempting to use unauthorized information in the taking of an exam;

B. Submitting as one's own work, themes, reports, drawings, laboratory notes, computer programs or other products prepared by another person;

C. Knowingly assisting another student in obtaining or using unauthorized materials; or

D. Plagiarism.’’

Anyone suspected of academic dishonesty will be reported to the Dean of Students Office.

Special Accomodations

Iowa State University complies with the Americans with Disabilities Act and Section 504 of the Rehabilitation Act. Any student who may require an accommodation under such provisions should contact the instructor no later than the end of the second week of class or as soon as you become aware of it.

Slides based on course notes by Dr. Xiaoqiu Huang

Reading Assignment

- Introduction to Junit (to be posted)
- RIPR Fault Lifecycle (to be posted)
- In Mathur, Chapter 1, sections
  - 1.1, 1.2, 1.3, 1.4.1, 1.4.2, 1.11, 1.12, 1.17 & 1.18

Pre-Requisites

Prerequisites for this course are:

COM S 309; COM S 230 or CPR E 310;
ENGL 250, SP CM 212.

If you have not met these prerequisites and you do not have a waiver on file, you will be automatically dis-enrolled.

Please see me now, here in the classroom.

Slides based on course notes by Dr. Xiaoqiu Huang