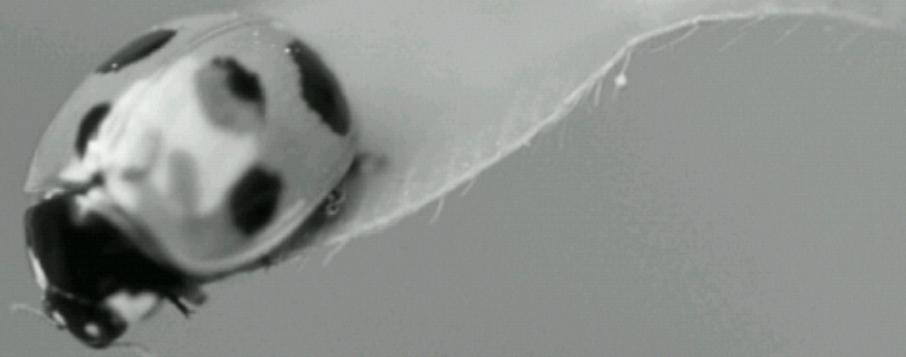


ECE453/SE465/ECE653/CS447/CS647: Software Testing, Quality Assurance, and Maintenance



Lin Tan

January 3, 2016

What this course is and is not

- A class to teach you
 - how to test as a developer & become a better developer
 - how to measure testing coverage, and
 - how to use & write automated testing tools.
- Not a QA course (only 1-2 lectures on QA)

“The Testing Course”



ECE451/SE4653:
Requirements

ECE452/SE464:
Design

ECE453/SE465/CS447/
ECE653/CS647:
Testing

Software Testing

Instructor: Lin Tan

3

Poll

- Do you use testing tools such as JUnit?
- Do you use dynamic tools such as Valgrind?
- Do you use fully automated testing tools such as Coverity, FindBugs, etc?

- Does industry use
 - NC
 - EC
 - PPC?

- Why aren't they used (more widely)?
- Possible Solutions?

Windows

A fatal exception 0E has occurred at 0028:C0011E36 in UXD UMM(01) + 00010E36. The current application will be terminated.

- * Press any key to terminate the current application.
- * Press CTRL+ALT+DEL again to restart your computer. You will lose any unsaved information in all applications.

Press any key to continue _

Why Software Testing?

- Software failures are costly!
 - One hour downtime costs >\$6M on average [Gartner'98].
 - Bugs account for 30% of system failures [Marcus'00]
 - Contribute to 55% of most severe vulnerabilities [CERT]
 - Cost billions annually [NIST'02, myitVendor'02]
 - \$22.2B annual potential cost reduction from feasible infrastructure improvements [NIST'02]
- Testing is an effective way of avoiding software failures.

Top 2 Requests from Companies

- “..., we spoke to representatives from a half-dozen leading software companies.
- We were struck by the unanimity of the number-one request from each company: that students learn **how to enhance sparsely documented legacy code**. In priority order, other requests were **making testing a first-class citizen**, working with nontechnical customers, performing design reviews, and working in teams.” [CACM May 2012]

Testing is Important

- Testing accounts for over 50% of software development costs.

Program Managers often say:
“Testing is too expensive.”

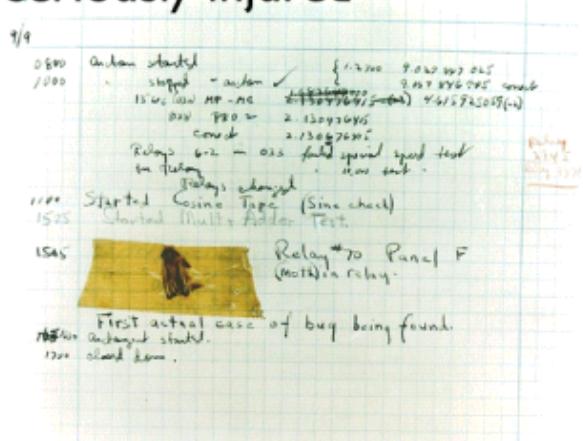
Not testing is even more expensive

How Software Fails?

- Crash
- Hang
- Data Corruption
- Incorrect Functionality
- Performance Degradation
- ...

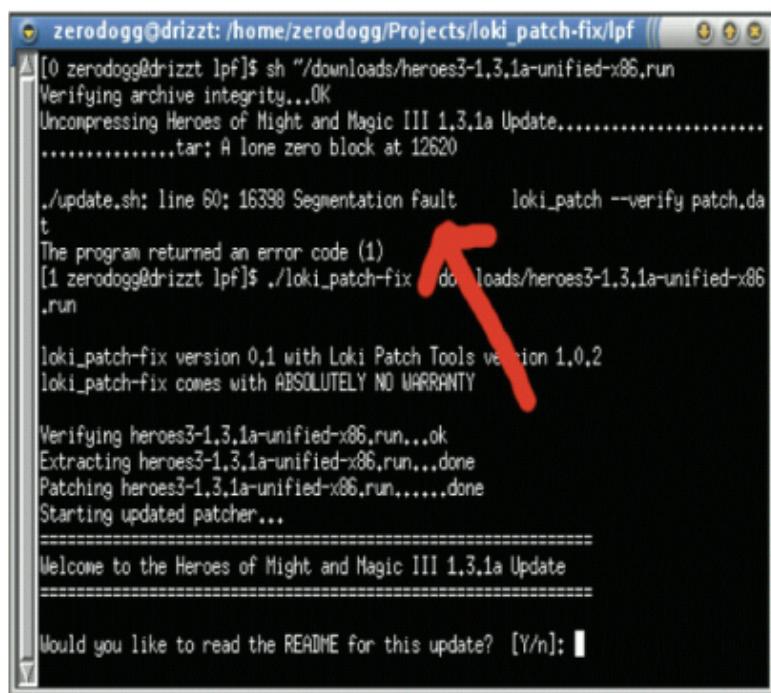
“Famous” Software Bugs

- Therac-25 medical accident, 1985-1987
 - At least 5 patients die; several seriously injured
- Northeast Blackout, 2003
 - 50M people out of power;
 - \$6B financial losses
- Ariane 5 Flight 501 Crash, 1996
 - a loss of \$370M
 - <http://www.youtube.com/watch?v=kYUrqdUyEpl>
- Morris Worm, 1988
 - ~6000 computers affected; \$10M-100M losses estimated



<http://www.wired.com/software/coolapps/news/2005/11/69355>

Why Does Software Fail? (I)



A screenshot of a terminal window titled "zerodogg@drizzt: /home/zerodogg/Projects/loki_patch-fix/lpf". The terminal shows the following command and its output:

```
[0 zerodogg@drizzt lpf]$ sh "/downloads/heroes3-1.3.1a-unified-x86.run"
Verifying archive integrity...OK
Uncompressing Heroes of Might and Magic III 1.3.1a Update.....
.....tar; A lone zero block at 12620

./update.sh: line 60: 16398 Segmentation fault      loki_patch --verify patch.dat
The program returned an error code (1)
[1 zerodogg@drizzt lpf]$ ./loki_patch-fix /downloads/heroes3-1.3.1a-unified-x86
.run

loki_patch-fix version 0.1 with Loki Patch Tools version 1.0.2
loki_patch-fix comes with ABSOLUTELY NO WARRANTY

Verifying heroes3-1.3.1a-unified-x86.run...ok
Extracting heroes3-1.3.1a-unified-x86.run...done
Patching heroes3-1.3.1a-unified-x86.run.....done
Starting updated patcher...
=====
Welcome to the Heroes of Might and Magic III 1.3.1a Update
=====

Would you like to read the README for this update? [Y/n]:
```

A red arrow points to the line containing the Segmentation Fault error message.

- Segfaults - incorrect memory usage

This course will use slides and notes adapted from Prof. Patrick Lam's.

Software Testing

Instructor: Lin Tan

13

Why Does Software Fail? (2)



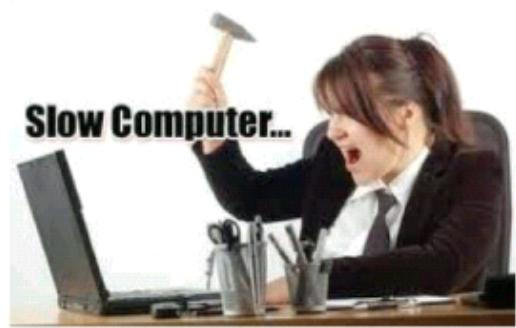
- Deadlocks

Why Does Software Fail? (3)

```
public int add(int x, int y) {  
    return x - y;  
}
```

- Wrong implementation

Why Does Software Fail? (4)



- Memory Leaks
 - Exist in safe languages such as Java as well

Why Does Software Fail? (5)

- Regression bugs

Some “Bugging” Software Failure

- Suspected leaks:
 - Firefox or Chrome becomes a memory hog after days
- Your own “favourite” example?

Avoiding Software Failures

- Test the software (in-house, externally)
- Code review
- Better design (“write better code!”)
- Include fewer features
- Better programming languages & IDEs
- Defensive programming (esp. w.r.t. plugins)

Mitigation: Failure is Inevitable

- Software (in reality) never completely works!
- Aim: Produce software that is good enough!

Coping with An Imperfect World (I)

- Disclaim liability

25. LIMITATION ON AND EXCLUSION OF DAMAGES. You can recover from Microsoft and its suppliers only direct damages up to the amount you paid for the software. You cannot recover any other damages, including consequential, lost profits, special, indirect or incidental damages.

(Vista license)

Coping with An Imperfect World (2)

- Release patches
- Backup user data
- Failure recovery

Ways of Testing Software

- Compile it
- Run it on one input
- Run it on many inputs
- Run it on a representative set of inputs
- Run it on all inputs (static analysis)

Other Testing Concerns

- Integration testing
- Nonfunctional properties
- Regression tests

Key Concepts: Coverage

- Idea: find a reduced space and cover it with tests.
- Possible spaces: graphs, logic, input space, syntax.

In This Course

- Learn about
 - Graph Coverage
 - Logic Coverage
 - Syntax-based Coverage
 - Input-space Based Coverage
 - Testing in Practice
 - State-of-the-art Techniques
 - Use & Build Tools
 - Concurrency
 - ...
- View this course as a *window* to the larger world ...

Tools

- JUnit
- Clang/LLVM
- Valgrind
- Daikon
- Randoop
- Java Path Finder
- cppcheck
- FindBugs
- Splint
- Coverity
- Visual Studio (SAL)
- iComment
- ...

Openings

- Our research lab has [multiple openings](#)
 - [ECE/SE499, USRA, URA, Co-op](#)
 - [Master's, PhD](#)
- Research areas & topics (incomplete):
 - Automated bug detection & fixing
 - Apply Natural Language Processing (NLP), Machine Learning, Program Analysis to improve software reliability
 - Comment & document analysis
 - Code dependency analysis
 - ...