Software Analysis

Lab & Discussion
April 19, 2011

Prof. Dr. Klaus Ostermann
Tillmann Rendel, MSc
Information

- Tooldemo: FAUmachine?
- Lecture on software product lines: moved to Monday, 14:15 to 15:45 and 16:15 to 17:45

In this discussion session:
- Work in small groups.
- Discuss.
- Present your results to the whole group.
How Failures Come to Be

Relate the following statements to the terms defect, infection, propagation, and failure. Discuss how they (possibly) came to be, and how they (possibly) relate to the output:

- A program throws a null pointer exception.
- A print statement `printf("Helo World")` has a typo.
- A constant $\pi = 31.4$ is declared, but all but one test case pass.
- Variable $z$ has the value 15.
- A bug is removed by fixing three files.
- A clocks shows Greenwich mean time rather than the local time zone.

exercise 1.1 from [WPF]
Proving correctness?

„Beware of bugs in the above code; I have only proved it correct, not tried it.“

Donald Knuth

If we can prove a program correct, we have no need for testing or debugging.

• Argue for and against this assertion.
• Use at least three arguments in either case.

exercise 1.5 from [WPF]
Reporting a Bug

- Write a bug report for the sample problem.
- Justify the amount of information you gave.

exercise 2.1 from [WPF]
Bugzilla

Visit the Mozilla problem-tracking site at http://bugzilla.mozilla.org/, and answer the following questions.

- How many problems have been entered as „new“ into Bugzilla in the past three days?
- How many of these are critical (or even blocking?)
- How many of these are invalid? Why?
- How many unresolved or unconfirmed problems are there in the currently released version?
- Which is the worst currently unresolved problem?
- According to the problem priority, which problem would you adress first as a programmer?

exercise 2.2 from [WPF]
Tracking problems

- What else (in addition to problem-tracking software) could be used to manage software problems?
- What else (in addition to software problems) could be managed using a problem-tracking system?

Adapted from exercise 2.3 and 2.4 from [WPF]
Testing

• Discuss the difference between testing for debugging and testing for validation.

• Is testing at the presentation layer of a command-line tool the same as functionality testing? Discuss similarities and differences.

• JUnit works fine to discover defects at the unit level, but fails if a failure is caused by multiple units. Discuss.

Adapted from exercise 3.1, 3.2 and 3.9 from [WPF]
Feedback

• What do you expect from an lab or discussion session?
• Do you like what we did today?