Software Testing

Nai-Wei Lin

Department of Computer Science and Software Engineering

National Chung Cheng University

Content

- Instructor information
- Course description
- Reference books and technical papers

Instructor Information

- Nai-Wei Lin, 林迺衛

- Office: EA502

- Phone: 272-0411 ~ 33110

- E-Mail: naiwei@cs.ccu.edu.tw

- HomePage:
  http://www.cs.ccu.edu.tw/~naiwei/

- Office Hour:
  11:30~12:00, Tuesday & Thursday

Course Description

- This course introduces the principles, techniques, and tools of software testing.

- There is no textbook for this course. The course materials are collected from several reference books and technical papers.

- To gain practical experiences, this course will utilize the development environment Eclipse and the testing tools JUnit and JWebUnit to exercise various testing techniques for Java programs.

Reference Books

- D. Galin, Software Quality Assurance, Addison Wesley, 2004.

- P. C. Jorgensen, Software Testing: A Craftsman's Approach, CRC Press, 1995.

- C. Kaner, J. Falk, and H. Q. Nguyen, Testing Computer Software, Wiley, 1999.

- J. Tian, Software Quality Engineering, Wiley & Sons, 2005.

Reference Books

- J. Rumbaugh, I. Jacobson, and G. Booch, The Unified Modeling Language Reference Manual, Second Edition, Addison-Wesley, 2005.

- J. Warmer and A. Kleppe, The Object Constraint Language, Second Edition, Addison-Wesley, 2003.

- K. Apt and M. Wallace, Constraint Logic Programming Using ECLiPSe

Technical Papers

- Eclipse-Overview

- Eclipse-IDE

- Get started With the Eclipse Platform

- Using JUnit With Eclipse IDE

- E. Weyuker and B. Jeng, "Analyzing Partition Testing Strategies," IEEE Transactions on Software Engineering, Vol. 17, No. 7, p. 703-711, July 1991.

Technical Papers

- D. Hamlet and R. Taylor, "Partition Testing Does Not Inspire Confidence," IEEE Transactions on Software Engineering, Vol. 16, No. 12, p. 1402-1411, December 1990.

- Z. Jin and J. Offutt, "Coupling-based Criteria for Integration Testing," The Journal of Software Testing, Verification, and Reliability, Vol 8, No 3, p. 133-154, September 1998.

Technical Papers

- A. Rountev, S. Kagan, and J. Sawin, "Coverage Criteria for Testing of Object Interactions in Sequence Diagrams," in Proceedings of the International Conference on Fundamental Approaches to Software Engineering, p. 289-304, 2005.

Technical Papers

- A. M. Memon, M. L. Soffa, and M. E. Pollack, "Coverage Criteria for GUI Testing," in Proceedings of the 8th European Software Engineering Conference and 9th ACM SIGSOFT International Symposium on the Foundations of Software Engineering, p. 256-267, 2001.

- G. Gheorghiu, "Performance vs. Load vs. Stress Testing," 2005.

Technical Papers

- "Security Testing," WIKIPEDIA, The Free Encyclopedia.
- R. W. Miller and C. T. Collins, "Acceptance Testing," in Proceedings of 2001 XP Agile Universe Conference, 2001.

Resources

- STORM Software Testing Online Resources
- Eclipse: An Open Development Platform
- JUnit: A Unit Testing Tool for Java
- JWebUnit: A Testing Tool for Web Applications
- Emma: A Code Coverage Tool for Java
- EclEmma: An Eclipse Plugin for Emma
- MuJava: A Mutation System for Java
- MuClipse: An Eclipse Plugin for MuJava
- EclipseUML: An Eclipse Plugin for UML 2.1 Editor

Assignments and Exams

- There are 6~8 programming assignments.

- Midterm Exam: EA 204, 08:45~11:30, Tuesday, November 13, 2012.

- Final Exam: EA 204, 08:45~11:30, Tuesday, January 15, 2013.

Grading

- Programming Assignments: 40%
- Midterm Exam: 30%
- Final Exam: 30%

Course Policies

- Without a prior arrangement, a missed examination results in a grade of zero.

- The grade for a late assignment is deducted by 5 for each day delayed. The grade for a late assignment is zero if it is delayed more than 7 days.

- A miss of an examination or an assignment results in a grade of zero for the semester.

Course Policies

- Assignments in this course require individual attention and effort to be of any benefit. Unless otherwise stated on the assignment sheet, all assignment work is expected to be that of each student alone, and not the product of team efforts or collaboration with other authors. Plagiarism or the incorporation of another student's words or ideas constitutes theft of intellectual property; it will result in a grade of zero for the semester.