Course Description

The purpose of this course is to provide an overview of software testing and maintenance and how these activities fit into the Software Engineering Life-Cycle. Many examples used in the lectures are derived from analysis of various NASA systems. Topics include various forms of testing such as Functional Testing, Combinatorial Testing, Structural Testing, Model-Based Testing, Security-oriented testing as well as Software Architecture's role in testability & maintainability, Regression Testing, Automated Testing, Testing Coverage including MC/DC coverage and testing standards. In addition, topics related to software change, how to estimate, measure and manage software change as well as software change impact analysis will be covered. Assignments are hands-on and include working with testing tools such as junit, selenium, guice, jmock and ant.

Required technology: The assignments require access to a computer that runs Java, preferably MS Windows 7/8. For two in-class assignments students should bring laptop to class. More information will provided during the first lecture.

Prerequisites: Assignments require some familiarity with Java or similar programming language.

Method for communication outside classroom: email.

First class:1/29/2015
Due Dates: Midterm: 3/12/2015 Final: tentative 5/14/2015.
Due dates for Assignments will be provided during the first lecture and throughout the course.
No class:3/19/2015 (spring break)

Course attendance policy: Students must attend the following classes: Midterm, Final, Classes (two in-class assignments), Student presentation(s).

Grading procedures: Grading is calculated based on a sum of the weighted scores from (tentative): Midterm (20%), Final (30%), Assignments (40%), Student Presentation (10%). More information will be provided during the first lecture.

Required/Recommended Textbooks

No textbook. We will use a selected set of papers and articles. In addition to the assignments, students will read the selected papers and articles and derive suitable questions and answers that will be shared with the other students as study help. More information will provided during the first lecture.

Course Outline

The following topics are planned to be covered, however changes may occur:

• Introduction to Software Testing and Maintenance
• Testing and Maintenance in the Software Engineering Life-Cycle
• Software Architecture and testability & maintainability
- Functional Testing
- Combinatorial Testing
- Structural Testing
- Model-Based Testing
- Random-Based testing
- Security-oriented testing
- Usability-oriented testing
- Testing Object-Oriented Software
- Fault-based Testing
- GUI Testing
- System Testing
- Acceptance Testing
- Regression Testing
- Automated Testing
- Testing Coverage including MC/DC coverage and testing standards
- Test Execution
- Inspections
- Symbolic execution
- Reverse Engineering
- Recovering Architecture and Design information statically and dynamically
- Analyzing Programs, Systems of Systems and other Distributed Systems
- Impact Analysis and change effort prediction
- Measurement for Software Change
- Software Change Analysis