

**Carleton University**  
**Department of Systems and Computer Engineering**  
**SYSC 5105      Software Quality Engineering and Management      Winter 2012**  

---

**Course Outline**

**Instructor:** Yvan Labiche  
Room 7078 MC

520-2600 ext. 5583  
labiche@sce.carleton.ca

**Web site:**

Course material will be available on WebCT: look for the WebCT link from Carleton's web site.

**Prerequisites:** Undergraduate studies in software engineering, computer systems engineering, computer science, or related field.

You should be familiar with at least one programming language, and preferably an object-oriented language such as C++ or Java.

Knowledge of UML is preferable.

**Lectures:** Tuesdays and Thursdays from 19h30 to 21h00 in Paterson Hall 236, from Jan. 05 (first class) to Apr 05 (last class).

**Textbook:**

Title: Introduction to Software Testing

Authors: Paul Ammann, Jeff Offutt

Publisher: Cambridge University Press

ISBN: 978-0-521-88038-1

Note that no one text book currently available can cover the whole spectrum of software testing. Course notes are therefore equally important to understand the material. Additionally a bibliography of (possible) reference books/articles that are particularly relevant to this course will be posted on the course web site.

**References:** References for SYSC-5105 will be available on the course website.

**Office hours:** By appointment only.

**Grading Scheme:**

The course format depends, to some extent, on the number of students registered and is to be discussed during the first class. My current intent is to give equal weight to a final exam and a course project.

**Students from the University of Ottawa:**

Students from the University of Ottawa can request to have access to WebCT: please see <http://www5.carleton.ca/fgpa/student-forms/>.

**Students with Disabilities:**

Students with disabilities requiring academic accommodations in this course are encouraged to contact a coordinator at the Paul Menton Centre for Students with Disabilities to complete the necessary letters of accommodation. After registering with the PMC, make an appointment to meet and discuss your needs with me at least two weeks prior to the final exam. This is necessary in order to ensure sufficient time to make the necessary arrangements. Please note the following deadlines for submitting completed forms to the Paul Menton Centre: **March 7, 2012** for the Winter Term.

**Plagiarism:**

Plagiarism and cheating at the graduate level are viewed as being particularly serious and the sanctions imposed are accordingly severe. Students are expected to familiarize themselves with and follow the Carleton University Student Academic Integrity Policy (see the policies from the web site of the Faculty of Graduate and Postdoctoral affairs: <http://www5.carleton.ca/fgpa/student-forms/>). The Policy is strictly enforced and is binding on all students. Plagiarism and cheating – presenting another’s ideas, arguments, words or images as your own, using unauthorized material, misrepresentation, fabricating or misrepresenting research data, unauthorized co-operation or collaboration or completing work for another student – weaken the quality of the graduate degree. Academic dishonesty in any form will not be tolerated. Students who infringe the Policy may be subject to one of several penalties including: expulsion; suspension from all studies at Carleton; suspension from full-time studies; and/or a reprimand; a refusal of permission to continue or to register in a specific degree program; academic probation; or a grade of Failure in the course.

**Tentative Outline**

Week 1: Introduction & overview

- Definition of software quality engineering and management, and testing
- Where is it needed?
- Requirements for a real engineering discipline for software development
- The role of quantitative methods
- Testing fundamentals

Weeks 2&3: Testing from Graphs

- Control flow testing
- Data flow testing
- Integration testing
- Black box and White box testing

Weeks 4&5: Testing from textual descriptions

- Equivalence partitioning
- Category-partition testing
- Combinatorial techniques, logic function testing

Week 6: Testing from Boolean specifications

Week 7: Testing from textual structures

- Mutation testing

Weeks 8&9: Testing Object-Oriented systems

- Issues when testing OO systems
- Class and class cluster testing
- Integration testing

Week 10: Regression Testing

- Regression Testing Selection
- Regression Test Minimization
- Regression Test Prioritization

Week 11: Software reviews and inspections

- Fundamentals
- Inspection process and activities
- Inspection techniques
- Empirical studies

Weeks 12: Defect Analysis and Modeling

- Preventing defects: Defect root cause analysis
- Assessing V&V effectiveness: Orthogonal defect Classification
- Defect content models
- Measuring and predicting software reliability
- Certifying software reliability