Instructor: Dr. Issa Traore  
Office Hours: Days: Monday, Thursday  
Phone: 250-721-8697  
E-mail: itraore@ece.uvic.ca  
Time: 2:30-4:00pm  
Location: EOW 415

Lectures: A-Section(s): A01 / CRN 30722  
A02 / CRN 30723  
Days: Monday, Thursday  
Time: 13:00-14:20am  
Location: CLE A206

Labs: No lab  
Location: B-Section(s): B01 / CRN 30724  
B02 / CRN 30725  
Days: Monday  
Time: 2:30-5:20pm  
Location: ELW B220

Required Text:  
Title: Software Testing and Quality Assurance  
Author: Kshirasagar Naik and Priyadarshi Tripathy  
Publisher: Wiley  
Year: 2008

Optional Text:  
Title: 
Author: 
Publisher: 
Year: 

References:  
1. Software Quality Engineering, Jeff Tian, Wiley, 2005  
6. Lectures Notes/Slides: on Moodle

Assessment:  
Attendance/Class participation: 4%  
Project: 36% (Part I: 6%; Part II: 8%; Part III: 12%; Part IV: 10%)  
Mid-term Exam: 20%  
Date: June 26/2014  
Final: 40%

Note: Failure to complete all laboratory requirements will result in a grade of N being awarded for the course.
Due dates for assignments:
1. Project Part 1: due May 26, 2014
2. Project Part 2: due June 16, 2014

The final grade obtained from the above marking scheme will be based on the following percentage-to-grade point conversion:

<table>
<thead>
<tr>
<th>Passing Grades</th>
<th>Grade Point Value</th>
<th>Percentage for Instructor Use Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>9</td>
<td>90 – 100</td>
</tr>
<tr>
<td>A</td>
<td>8</td>
<td>85 – 89</td>
</tr>
<tr>
<td>A-</td>
<td>7</td>
<td>80 – 84</td>
</tr>
<tr>
<td>B+</td>
<td>6</td>
<td>77 – 79</td>
</tr>
<tr>
<td>B</td>
<td>5</td>
<td>73 – 76</td>
</tr>
<tr>
<td>B-</td>
<td>4</td>
<td>70 – 72</td>
</tr>
<tr>
<td>C+</td>
<td>3</td>
<td>65 – 69</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>60 – 64</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>50 – 59</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Failing Grades</th>
<th>Grade Point Value</th>
<th>Percentage for Instructor Use Only</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td>0</td>
<td>35 – 49</td>
<td>Fail, conditional supplemental exam. (For undergraduate courses only)</td>
</tr>
<tr>
<td>F</td>
<td>0</td>
<td>0 – 49</td>
<td>Fail, no supplemental.</td>
</tr>
<tr>
<td>N</td>
<td>0</td>
<td>0 – 49</td>
<td>Did not write examination, Lab or otherwise complete course requirements by the end of term or session; no supplemental exam.</td>
</tr>
</tbody>
</table>

The rules for supplemental examinations are found on page 80 of the current 2013/14 Undergraduate Calendar.

<table>
<thead>
<tr>
<th>Term in which E Grade Was Obtained</th>
<th>Application Deadline for Supplemental Exam</th>
<th>Supplemental Exam Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>First term of Winter Session (Sept – Dec)</td>
<td>February 28 in the following term</td>
<td>First week of following May</td>
</tr>
<tr>
<td>Second term of Winter Session (Jan – Apr)</td>
<td>June 30 in the following term</td>
<td>First week of following September</td>
</tr>
<tr>
<td>Summer Session (May – Aug)</td>
<td>October 31 in the following term</td>
<td>First week of following January</td>
</tr>
</tbody>
</table>

Deferred exams will normally be written at the start of the student's next academic term; i.e., approximately 4 months following the deferral of the exam.

Course Description

Course Objectives
The purpose of the course is to introduce fundamental notions of software quality and the techniques used to build and check quality in software systems. A particular emphasis is placed on quantitative assessment of software quality and quality control using software testing techniques.
**Learning Outcomes**

By the end of this course, students should have a good grasp of software quality metrics and models, software testing techniques, and software reliability analysis models and techniques.

**Syllabus**

The following syllabus is subject to the time available and may change during the term. Some of the topics may not be covered.

**Unit 1. Software Quality: Overview and Basics**

Introduce software quality standards and processes, quality planning and control, quality attributes and specification. Specifically, the following issues will be covered:

- Quality Assurance and Standards
- Quality Specification
- Quality Control

**Unit 2. Software Inspection**

Overview of different types of software review and focus on quality review through (formal) inspection. Introduce inspection process, documents and metrics.

**Unit 3. Quality Models and Measurements**

Presentation of quality management models; Use of quality models and data for in-process quality management and to guide software testing. Introduction of a number of techniques to quantify, classify and analyze discovered defects.

**Unit 4. Software Reliability Models**

Notions of software reliability and reliability growth. Overview of software reliability growth models (SGRM).

**Unit 5. Testing: Concepts and Management**

Presentation of testing dimensions, concepts, terminologies and processes. Introduction to lifecycle testing and to model-based testing. Discussion of underlying issues and approaches to test management. Introduction to test planning, test status and defect reporting.

**Unit 6. Domain Testing**

Presentation of selected test models and testing strategies: domain test model.

**Unit 7. Test generation from Finite-State Machines**

Presentation of selected test models and testing strategies: state-based test model.

**Unit 8. Control Flow and Data Flow Testing**

Notion of test adequacy; test coverage criteria and metrics; the basis-path test model; control flow testing.

**Unit 9. Combinational Testing**

Presentation of selected test models and testing strategies: combinational test model.
Unit 10. System Integration Testing

Presentation of selected test models and testing strategies: test integration.

Unit 11. Software Reliability Engineering

Notions of software reliability and availability; comparison between hardware and software reliability; Software Reliability modeling and metrics. Reliability block diagrams; concurrent systems (series/parallel) reliability. Application of reliability concepts and models within a disciplined and systematic software engineering process. Reliability validation and demonstration.

Note to Students:
Students who have issues with the conduct of the course should discuss them with the instructor first. If these discussions do not resolve the issue, then students should feel free to contact the ECE Chair by email or the ECE Chair's secretary to set up an appointment.
Accommodation of Religious Observance
See http://web.uvic.ca/calendar2013/GI/GUPo.html

Policy on Inclusivity and Diversity
See http://web.uvic.ca/calendar2013/GI/GUPo.html

Standards of Professional Behaviour
You are advised to read the Faculty of Engineering document Standards for Professional Behaviour at http://www.uvic.ca/engineering/current/undergrad/index.php#section0-25 which contains important information regarding conduct in courses, labs, and in the general use of facilities.

Cheating, plagiarism and other forms of academic fraud are taken very seriously by both the University and the Department. You should consult http://web.uvic.ca/calendar2013/FACS/UnIn/UARe/PoAcI.html for the UVic policy on academic integrity.

Plagiarism detection software may be used to aid the instructor and/or TA's in the review and grading of some or all of the work you submit.