CISC / CMPE 327 Software Quality Assurance
PRELIMINARY Course Outline - Fall 2019

Dr. Steven Ding, Goodwin Hall 536, ding@cs.queensu.ca
Office hours: Tue. 13:00–14:00 + Fri. 13:00–14:00 + by email appointment

Lectures:
Tue 15:30 (**STIRLING RM C**) / Thu 14:30 (**STIRLING RM C**) / Fri 16:30 (**STIRLING RM C**)

First lecture is Thursday, September 5th, 2019

TAs: TBA

In this course the primary role of the TAs will be to advise you on coordination and evaluation of your course project. The TAs will provide you with advice and assistance in working on the project.

TA advising: time and place TBA

Website: https://l1nna.com/course/cisc327/

Textbook: The primary source for course material will be lectures, augmented by additional secondary sources in the course notes available at the bookstore:

> Cordy, *CISC 327 Course Reader 2018*
> (or the 2014, 2015, 2016 or 2017 versions, which are identical)

Lectures and additional web materials will be provided online on the course website as the course progresses.

Course Project

Since this is a course on the practical aspects of software quality, the course project will provide you with an opportunity to put what you learn into practice. The expectation is that you will hand in work of the highest quality standard, and professionally presented.

The project will be carried out using the principles and philosophy of the Extreme Programming methodology. You will form small (three-person) software "companies" who will contract to provide a software system roughly specified by a set of user desires, which you will hone into requirements, specifications, design quality plans, tests, and test plans as you develop solutions. Details of the project will be handed out in the second week of the course.

Aims of the Course

The primary purpose of this course is to introduce you to the concepts, theory and practice of software quality assurance through process, testing, inspection and measurement. It is not a course in the broader aspects of software engineering, except insofar as they relate to quality control.

Integrity, Accommodation, Extenuating Circumstances, and Copyright

http://www.cs.queensu.ca/students/undergraduate/syllabus/year2018-19.php

Additional Remark on Integrity

See the exception under "Rule: One-Time Forgiveness".

Material Covered

Introduction (1 week)

- Quality - what is it, how is it measured, how is it achieved

Software Process (2 weeks)

- Software process models: plans for achieving and improving software quality
- eXtreme Programming: a controversial modern software process

Software Testing (3 weeks)

- Systematic Testing - what is it, levels of testing, designing for test
- Black Box Testing - functional, input, output, partitioning and gray box testing
- White Box Testing - coverage, path, decision and mutation testing
- Continuous Testing - regression, defect testing
- Test Automation - test maintenance and analysis, harnesses, tracking, tools

Software Inspection (3 weeks)

- Systematic Inspection - what is it, levels of inspection, inspection process, formal reviews
- Inspection in the Software Process - requirements, design, process and code inspections
- Code Inspection - techniques, practices, continuous inspection, refactoring
- Automated Inspection - static analysis, design recovery, clone detection, model checking

Software Measurement (2 weeks)

- Software Metrics - measurement basics, assessment and prediction
- Product Quality Metrics - external, internal, faults, failures and defects
- Code and Structure Metrics - size, complexity and functionality metrics
- Process Metrics - predicting cost and effort, regression analysis, function points

Software Dependability and Security (1 week)

- Software Security - issues and methods, web application security

Overflow & Review (1 week)

Marking

There is no final examination in this course. Instead, your exam mark consists of four 45-minute in-class "mini-exams" on the four main sections of the course. Each mini-exam is at the same level as the corresponding part of a final examination would be for that section of the course.

       4 in-class mini-examinations @ ~12.5% each (may vary)   = 50%

       6 course project assignments @ ~8% each (varies)   = 50%

Because half of the marks in the course are project marks, there is an additional requirement: your final mark in the course is bounded by your personal combined mark in the four mini-exams.

That is, in order to obtain a final A grade in the course, you must have at least a B grade in the combined mini-exams. To get a B grade, you must have at least a C, and so on.

Finally, you must obtain a passing mark (at least 50%) in the combined mini-exams in order to pass the course.

Rule: One-Time Forgiveness (mini-exams)
In general, I will drop your lowest mini-exam mark. This includes situations where you missed the mini-exam entirely (for any reason) and received a zero: zero is a very low mark, so it would be dropped. **Exception:** If a student receives a reduced mini-exam mark (such as zero) as a consequence of a departure from academic integrity on that mini-exam, that mini-exam mark will **not** be dropped—instead, all mini-exams will be counted when calculating the course grade (just as if this One-Time Forgiveness rule did not exist).

Preliminary Schedule

Subject to change—see the course website for the up-to-date schedule.

Mini-Examinations

E1: Introduction and process   Tuesday September 24
E2: Black & white box testing   Tuesday October 22
E3: Regression and inspection   Tuesday November 12
E4: Analysis, metrics, security   Friday November 29

Project Assignments

A0: Choose teams   Tuesday September 17
A1: Front end: requirements tests   Tuesday October 1
A2: Front end: rapid prototype   Tuesday October 15
A3: Front end: requirements testing   Tuesday October 29
A4: Back end: rapid prototype   Tuesday November 5
A5: Back end: unit testing   Tuesday November 19
A6: Integration and delivery   Tuesday December 3