Carleton University School of Computer Science

COMP 4004A: Software Quality Assurance Offered online for Fall 2024

Last Updated August 25th, 2024

1 CLASS LOGISTICS

Lectures: Most lectures are offered synchronously on Mondays and

Wednesdays from 1:05 PM to 2:25 PM via Zoom.

Recording of these lectures is planned but is **NOT** guaranteed. You are expected to attend the lectures and to have a closed

microphone.

Several lectures will be offered asynchronously (i.e., posted for

you to listen to in the subsequent days).

Fall Break: week of October 21st (no office/TA hours that week)

Withdrawal by November 15th 2024

'Office Hours' by email appointment only and held via Zoom

2| INSTRUCTOR INFO

Jean-Pierre Corriveau email: jeanpier AT scs.carleton.ca

No textbook is used in this course

Material for all lectures (including the recordings of lectures and links to youtube videos) will be posted to Brightspace.

Brightspace will also be used for all announcements (including the Zoom link for lectures) and for all student assignment submissions.

All students are responsible for immediately gaining access to Brightspace.

TAs: names, emails, git credentials and office hours

will be announced in Brightspace early September

3] LEARNING OBJECTIVES:

Theory: (verified via the final exam)

- Understand the basics of software testing (in particular, equivalence partitioning) and of software metrics.
- Understand the principles and limitations of test-driven development (TDD) in the context of Agile Development, as well as issues pertaining to refactoring.
- Understand behavior-driven development (BDD) for acceptance testing
- Understand the principles and limitations of model-based testing (MBT).
- Understand some seminal papers and advanced issues pertaining to s/w testing.

Practice: (verified via 3 assignments)

Gain experience at learning and using testing tools BY YOURSELF:

- Gain experience with using JUnit for TDD.
- Gain experience with BDD and scenario-based testing using Cucumber.
- Gain experience with testing a web application using Selenium.

4| DETAILED COURSE DESCRIPTION

It is widely acknowledged that *software quality* is of the utmost importance. Yet, despite recent advancements in advanced techniques (such as program verification, automatic debugging, assertion deduction and model-based testing), Ralph Johnson (of Gang of Four design patterns fame) and many others **still** view software verification as a "catastrophic computer science failure".

In this course, we first and foremost explore the issue of software **testing**, that is, the **execution** of software in order to find errors in software. A first pervasive concern will be *test automation*, which is necessary if testing is to be objective, systematic, and scalable. A second pervasive concern will be *traceability* between tests and application code.

We will first briefly review software development processes in order to understand the central role of software testing in an agile approach to software development. Before going any further, we will then overview the foundations and inherent limitations of software testing and introduce the fundamental distinction between *validation* and *verification*.

Next we will consider test driven development (TDD), its strengths and limitations, as well as those of **JUnit** and *unit testing*. Such a code-oriented approach, carried out by developers, corresponds to current practice in most of industry.

Mastering *equivalence partitioning* is a key aspect for making any software testing approach efficient. Consequently, we will study it at length. Refactoring, which is inherent to TDD, is another technique that has been proposed for improving tests. We will discuss it in the context of application code, as well as for code used in tests. We also need to understand the notion of *coverage*, which is widely used for assessing how much testing is required.

Understanding the shortcomings of a purely bottom-up code-oriented approach to software development and testing, we will then introduce acceptance testing and the use of **Cucumber** for it. We will emphasize the systematic *testing of scenarios* (e.g., as captured in use cases or user stories) for acceptance testing. This will lead us to consider *model-based techniques* for software testing.

In the latter portion of the course we will study some seminal papers in software testing, as well as address miscellaneous topics such software metrics, the rise and fall of Spec Explorer at Microsoft, etc.

5| PREREQUISITES

A student registering in COMP 4004 is assumed to have a **strong** background in object-oriented design **and** programming. More specifically, a student should have taken COMP 2401, 2406 *and* 3004, or their equivalent.

You are expected to work individually and autonomously (i.e. learning by yourself). This course involves a considerable amount of programming as well as extensive testing.

6| EVALUATION SCHEME

3 INDIVIDUAL Testing Assignments 18% + 15% + 17% = 50%

- The first 2 assignments address the first *iteration* of a simple game. This first iteration is strictly text-based.
- The 1st assignment involves the use of TDD. It's due October 15th 2024.
- The 2nd assignment builds on top of the first. It involves scenario modeling and testing using Cucumber. It's due November 6th 2024.
- The 3rd assignment corresponds to the next iteration of the game (which entails changes to the code of the first iteration and thus regression testing). It builds on top of the two previous assignments and involves coding a web interface and testing it using Selenium. It's due December 6th 2024.
- All 3 assignments involve coding, extensive traceability documentation and the creation and submission of videos demonstrating your code's behavior and your testing. All of this is considerably time-consuming (including uploading and verifying your videos).
- Please be aware that no solution to any of the assignments will be posted.

1 CLOSED-BOOK FINAL EXAM 50%

• The specific topics addressed by this exam will be discussed in class later in the term.

7] SOFTWARE

For the assignments, students will be using a **PRIVATE** repository (Github), a dependency manager (Maven), an IDE (IntelliJ) and Java, as well as Junit, Cucumber and Selenium. Exact details, including specific versions to use, are to follow in Brightspace. No substitution will be allowed for any of these tools.

Most importantly, students are expected to learn by themselves all tools used in this course. Lectures will NOT be discussing the technical aspects of the installation and use of the tools required by the assignments. Questions on these issues should be directed to the TAs. It must be emphasized that it is a key learning objective to have students tackle mostly on their own such issues (especially given that, in the past, most installation problems stemmed from specificities of the computer of a student, NOT the tool to install). Questions about the game and the requirements of the assignments should be sent to the instructor.

There are several tutorials available online for the tools we use; in particular for JUnit (http://www.vogella.com/tutorials/JUnit/article.html) and Cucumber.

8 SUGGESTED OPTIONAL READINGS

1) Robert Binder (good introduction to OO Testing)

Object-oriented Testing, Addison-Wesley

2) K. Naik and P. Tripathy (pricey but excellent reference for s/w testing) Software Testing and Quality Assurance, Wiley, 2008

9 COURSE LOGISTICS

Please be sure to have your microphone closed at all times during lectures. The chat of Zoom will be used to ask me questions.

All students in the class, the instructor, and any guests should be treated with respect during all interactions.

Wrt assignments, late submissions WILL NOT BE ACCEPTED and will get a mark of 0. Assignment submission is handled electronically and there is no "grace period" with respect to a deadline. It is crucial you start early your assignments and leave plenty of time to upload your submissions. Last minute technical issues will not lead to more time to submit your assignment. Also, if we cannot access your repository or open the file(s) of your submission, you will get a mark of 0 for that assignment.

Do not expect extensions for the assignments.

Furthermore:

- 1) There will be NO supplemental or grade raising exam in this course.
- 2) No mark or extra work can be substituted for another mark.
- 3) About ChatGPT: Please be aware that many companies ban its use. You may want to try it to experience its limitations wrt testing an unknown game... Using it defeats the learning objectives of 4004, namely to learn to create and run tests.
- 4) Should you become physically or mentally sick during an assignment, please immediately **email** me. You *may* get a new deadline for the current assignment BUT please be aware that:
 - Such a new deadline will **NOT** grant you more time for the subsequent assignments
 - I will look at how much code and tests run at the time of this request
 - Long-term incapacitation (i.e., 5 days or more) is handled with the help of the Registrar's office
- 5) The Fall term ends December 6th and no submission is accepted after that date.
- 6) Your mark for *each* assignment *and* for the final exam must be greater than 35 out of 100 in order to pass the course.
- 7) For each assignment, you will be submitting one or more files. These files must be given the correct filename and be provided in the specified format. Incorrectly named items or in the incorrect format will be penalized and may receive a mark of zero.
- 8) Assignments are to be done **individually** *without any sharing*. Collaboration between students is *not* permitted: all alleged plagiarism will be reported to the office of the dean of Science (ODS). Penalties for such offences can be found on the ODS webpage: https://science.carleton.ca/academic-integrity/. In particular, posting *any* work put in a public location (e.g., a GitHub public repository, or Chegg, CourseHero, OneClass) constitutes a form of sharing that enables plagiarism. As such, if your work is plagiarized from a public posting of yours, you will be considered to have participated in this offence!

All material created for this course (including exams and assignments) remains the exclusive intellectual property of their author(s). They are intended for personal use and may not be reproduced or redistributed without prior written consent of the author(s). In particular, no recorded lecture of mine should be posted elsewhere than where I post it.

10| OTHER USEFUL INFORMATION

Underdergraduate Academic Advisor

The Undergraduate Advisors for the School of Computer Science are available in Room 5302HP; or by email at scs.ug.advisor@cunet.carleton.ca. The undergraduate advisors can assist with information about prerequisites and preclusions, course substitutions/equivalencies, understanding your academic audit and the remaining requirements for graduation. The undergraduate advisors will also refer students to appropriate resources such as the Science Student Success Centre, Learning Support Services and Writing Tutorial Services.

Science Student Success Centre (SSSC)

The Science Student Success Centre is a central advising unit for students in Science courses. We help students achieve their goals by providing access to resources, workshops and activities that enhance their academic and study skills, and help them make key connections with their peers. Mentors can help you customize an individual study plan which includes weekly and semester work or study schedules, and also help when you need information on developing a new study strategy, obtaining summer job opportunities, or clarifying ideas and concepts to better understand and cope with new course content. Science mentors can help you learn how to learn what you need to learn for your classes.

Drop by the Science Student Success Centre at 3431 Herzberg Laboratories or visit http://sssc.carleton.ca/

SCS Computer Laboratory

Students taking a COMP course can access the SCS computer labs. The lab schedule and location can be found at: https://carleton.ca/scs/tech-support/computer-laboratories/. All SCS computer lab and technical support information can be found at: https://carleton.ca/scs/tech-support/. Technical support staff may be contacted in-person or virtually, see this page for details: https://carleton.ca/scs/tech-support/contact-it-support/.

University Policies

For information about Carleton's academic year, including registration and withdrawal dates, see https://calendar.carleton.ca/academicyear/

In accordance with the Carleton University Undergraduate Calendar Regulations, the letter grades assigned in this course will have the following percentage equivalents:

A+ = 90-100	B+ = 77-79	C + = 67-6	9 D+ = 57-	59
A = 85-89	B = 73-76	C = 63-66	D = 53-5	6
A- = 80-84	B- = 70-72	C = 60-62	D- = 50-5	52
F = <50	WDN = Withdrawn from the cour	se D	EF = Deferred	

University Policies:

• Academic Accommodations

Carleton is committed to providing academic accessibility for all individuals. Please review the academic accommodation available to students here: https://students.carleton.ca/course-outline/.

• Academic Integrity

Student Academic Integrity Policy. Every student should be familiar with the Carleton University Student Academic Integrity policy. A student found in violation of academic integrity standards may be sanctioned with penalties which range from a reprimand to receiving a grade of F in the course, or even being suspended or expelled from the University. Examples of punishable offences include plagiarism and unauthorized collaboration. Any such reported offences will be reviewed by the office of the Dean of Science. More information on this policy may be found on the ODS Academic Integrity page: <u>Academic Integrity | Faculty of Science (carleton.ca)</u>.

Plagiarism. As defined by Senate, "plagiarism is presenting, whether intentional or not, the ideas, expression of ideas or work of others as one's own". Such reported offences will be reviewed by the office of the Dean of Science. More information and standard sanction guidelines can be found here: https://science.carleton.ca/students/academic-integrity/. Please note that content generated by an unauthorized A.I.-based tool *is* considered plagiarized material.

Unauthorized Collaboration. Senate policy states that "to ensure fairness and equity in assessment of term work, students shall not co-operate or collaborate in the completion of an academic assignment, in whole or in part, when the instructor has indicated that the assignment is to be completed on an individual basis".

Student Rights & Responsibilities

Students are expected to act responsibly and engage respectfully with other students and members of the Carleton and the broader community. See the <u>7 Rights and Responsibilities Policy</u> for details regarding the expectations of non-academic behaviour of students. Those who participate with another student in the commission of an infraction of this Policy will also be held liable for their actions.

Student Concerns

If a concern arises regarding this course, your first point of contact is me: Email me and I will do my best to address your concern. If I am unable to address your concern, the next points of contact are (in this order):



Note: You can also bring your concerns to <u>Ombuds services</u>.

Mental Health

If you are struggling, please do not hesitate to reach out. I am happy to help and/or direct you to resources that might help. Remember that Carleton also offers an array of mental health and well-being resources, which can be found here.

Online Community Expectations for Social Platforms

With the growing use of social platforms (e.g., Discord) on campuses, it is important to keep in mind that university codes of conduct still apply to the behaviour of students online. Please be considerate and respectful while engaging with peers and remember that we are all humans, and that your words matter. If any student witnesses or experiences harassment, I encourage you to reach out to me. Alternatively, you can contact <u>Ombuds Services</u> or <u>Carleton Equity and Inclusive Communities</u>.

Community Guidelines

The following values are fundamental to academic integrity and are adapted from the International Center for Academic Integrity*. Please behave with these values in mind:

	As students, we will	As a teaching team, we will
Honesty	 Honestly demonstrate our knowledge and abilities on assignments and exams Communicate openly without using deception, including citing appropriate sources 	 Give you honest feedback on your demonstration of knowledge and abilities on assignments and exams Communicate openly and honestly about the expectations and standards of the course through the syllabus, and with respect to assignments and exams
Responsibility	 Complete assignments on time and in full preparation for class Show up to class on time, and be mentally/physically present Participate fully and contribute to team learning and activities 	 Give you timely feedback on your assignments and exams Show up to class on time, and be mentally & physically present Create relevant assessments and class activities
Respect	 Speak openly with one another, while respecting diverse viewpoints and perspectives Provide sufficient space for others to voice their ideas 	 Respect your perspectives even while we challenge you to think more deeply and critically Help facilitate respectful exchange of ideas
Fairness	 Contribute fully and equally to collaborative work, so that we are not freeloading off of others Not seek unfair advantage over fellow students in the course 	 Create fair assignments and exams, and grade them in a fair, and timely manner Treat all students equitably
Trust	 Not engage in personal affairs while on class time Be open and transparent about what we are doing in class Not distribute course materials to others without authorization 	 Be available to all students when we say we will be Follow through on our promises Not modify the expectations or standards without communicating with everyone in the course

Courage

- Say or do something when we see actions that undermine any of the above values
- Accept a lower or failing grade or other consequences of upholding and protecting the above values
- Say or do something when we see actions that undermine any of the above values
- Accept the consequences (e.g., lower teaching evaluations) of upholding and protecting the above values

Land Acknowledgement

We acknowledge that the land on which we gather is the traditional and unceded territory of the Algonquin nation.

 $^{^{\}star}$ This class statement of values is adapted from Tricia Bertram Gallant, Ph.D.