EXPANDED COURSE DESCRIPTION

ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Lassonde School of Engineering
Electrical Engineering Computer Science

LE / EECS 2030 3.0 SECTION E
ADVANCED OBJECT ORIENTED PROGRAMMING
FALL 2017 / WINTER 2018

Last Modified Date: 08/29/2017

COURSE CALENDAR DESCRIPTION

This course continues the separation of concern theme introduced in EECS1020 and EECS1021. While EECS1020 focuses on the client concern, this course focuses on the concern of the implementer. Hence, rather than using an API (Application Programming Interface) to build an application, the student is asked to implement a given API. Topics include implementing classes (non-utilities, delegation within the class definition, documentation and API generation, implementing contracts), implementations (implementing aggregates versus compositions and implementing collections), inheritance hierarchies (attribute visibility, overriding methods, abstract classes versus interfaces, inner classes); applications of aggregation and inheritance in concurrent programming and event-driven programming; recursion; searching and sorting including quick and merge sorts; stacks and queues; linked lists; binary trees. Three lecture hours and weekly laboratory sessions. Prerequisites: General Prerequisite; LE/EECS1021 3.00 or LE/EECS 1020 3.00 or LE/EECS1022 3.00 or LE/EECS 1720 3.00. Course credit exclusions: LE/EECS1030 3.00, AP/ITEC 2620 3.00. (NOTE: The General Prerequisite is a cumulative GPA of 4.50 or better over all major EECS courses. EECS courses with the second digit "5" are not major courses.)

INSTRUCTOR(S)

<table>
<thead>
<tr>
<th>Name</th>
<th>Section / Format / Term</th>
<th>Contact Email</th>
<th>Contact Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ma, Burton</td>
<td>Sec. E / LECT / F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TOPICS AND CONCEPTS

List of Topics

Topic 1: Non-static features: UML diagrams, memory diagrams, this keyword, constructor, accessor and mutator, equals, hashCode, toString, compareTo, tests, class invariant, Javadoc

Topic 2: Aggregation and composition: UML diagrams, memory diagrams, == versus equals, defensive copying in composition, aliasing, shallow copying, deep copying, tests

Topic 3: Inheritance: UML diagrams, memory diagrams, which features are inherited, subclass constructors, overriding methods, super keyword, contracts in inheritance hierarchies, final, abstract classes, interfaces, tests, Javadoc

Topic 4: Concurrent programming: java.lang.Thread, starting, pausing, interrupting, joining, application of aggregation and inheritance tests (and their limitations) [This topic is optional and will be presented at the end of the course if time permits.]

Topic 5: Event-driven programming: application of aggregation and inheritance

Topic 6: Recursion: examples of recursive methods, implementation of recursive methods, binary search, merge/quick sort, proving (informal) termination and correctness, solving recurrence relations, big-O (informal proofs)

Topic 7: Linked lists: implementation of singly linked list(size, contains, addFirst, add, get, remove), implementation of doubly linked list, implementation of Iterator, application of recursion, big-O (informal proofs)
Topic 8: Stacks and queues: precondition, postcondition, class invariants (informal proofs), applications, implementation using arrays and List, big-O (informal proofs)

Topic 9: Binary trees: implementation: array and linked, application of recursion

LIST OF LEARNING OUTCOMES AND EXAMPLES OF

Learning Outcomes
- Implement an API (Application Programming Interface).
- Test the implementation.
- Document the implementation.
- Implement aggregations and compositions.
- Implement inheritance.
- Use recursion.
- Implement linked lists.
- (Informally) prove that recursive algorithms are correct and terminate.
- (Informally) analyze the running time of (recursive) algorithms.

GRADED ASSESSMENT

Evaluation
The final grade of the course is usually based on a combination of lab tests, lab exercises, and a final exam.

Missed tests: Students with a documented reason for missing a test, such as illness, compassionate grounds, etc., will have the weight of the missed test shifted. For Test 1, the weight will be shifted to Test 2. For all other tests, the weight will be shifted to a make-up test held during the exam period. This make up test will cover all the material presented in the course.

The only accepted documentation for missing a test due to illness is a completed Attending Physician's Statement. However, once a student begins writing a test, the weight of that test will not be shifted for any reason. Thus, if a student is not feeling well, it is recommended that the student not attend the test, seek the advice of a physician, and submit a completed Attending Physician's Statement to the instructor as soon as possible.

ADDITIONAL INFORMATION

Course Text
Absolute Java, 5th or 6th edition, by Savitch and Mock. Published by Pearson.

Academic Honesty
During tests, students are expected to do their own work. Looking at someone else's work during the test, talking during the test, using aids not permitted (such as a phone) during the test, and impersonation are all examples of academically dishonest behaviour. Students are expected to read the Senate Policy on Academic Honesty.

Additional Information

Academic Integrity: There is an academic integrity website with comprehensive information about academic honesty and how to find resources at York to help improve students' research and writing skills, and cope with University life. Students are expected to review the materials on the Academic Integrity website.

Access/Disability: York University is committed to principles of respect, inclusion and equality of all persons with disabilities across campus. The University provides services for students with disabilities (including physical, medical, learning and psychiatric disabilities) needing accommodation related to teaching and evaluation methods/materials. These services are made available to students in all Faculties and programs at York University.
Students in need of these services are asked to register with disability services as early as possible to ensure that appropriate academic accommodation can be provided with advance notice. You are encouraged to schedule a time early in the term to meet with each professor to discuss your accommodation needs. Please note that registering with disabilities services and discussing your needs with your professors is necessary to avoid any impediment to receiving the necessary academic accommodations to meet your needs.

Additional information is available at the following websites:

- Counselling & Disability Services
- York Accessibility Hub

**Religious Observance Accommodation:** York University is committed to respecting the religious beliefs and practices of all members of the community, and making accommodations for observances of special significance to adherents. Should any of the dates specified in this syllabus for an in-class test or examination pose such a conflict for you, contact the course director within the first three weeks of class. Similarly, should an assignment to be completed in a lab, practicum placement, workshop, etc., scheduled later in the term pose such a conflict, contact the course director immediately. Please note that to arrange an alternative date or time for an examination scheduled in the formal examination periods (December and April/May), students must complete an Examination Accommodation Form, which can be obtained from Student Client Services, Student Services Centre or online.

**Student Conduct in Academic Situations:** Students and instructors are expected to maintain a professional relationship characterized by courtesy and mutual respect. Moreover, it is the responsibility of the instructor to maintain an appropriate academic atmosphere in the classroom and other academic settings, and the responsibility of the student to cooperate in that endeavour. Further, the instructor is the best person to decide, in the first instance, whether such an atmosphere is present in the class. The policy and procedures governing disruptive and/or harassing behaviour by students in academic situations is available online.

---

Many courses utilize Moodle, York University's course website system. If your course is using Moodle, click here to access it.

*Moodle @ York University*