EXPANDED COURSE DESCRIPTION
ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
Lassonde School of Engineering
Electrical Engineering Computer Science
LE / EECS 2030 3.0 SECTION C
ADVANCED OBJECT ORIENTED PROGRAMMING
FALL 2019 / WINTER 2020

Last Modified Date: 08/07/2019

COURSE CALENDAR DESCRIPTION

This course continues the separation of concern theme introduced in LE/EECS 1020 3.00 and LE/EECS1021 3.00. While 1020 and 1021 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), aggregations (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. Prerequisites: cumulative GPA of 4.50 or better over all major EECS courses (without second digit "5"); LE/EECS1021 3.00 or LE/EECS 1020 (prior to Fall 2015) 3.00 or LE/EECS1022 3.00 or LE/EECS 1720 3.00. Course credit exclusions: AP/ITEC 2620 3.00. Previously offered as: LE/EECS1030 3.00, LE/CSE 1030 3.00.

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. C / LECT / F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ADDITIONAL INFORMATION

COURSE LEARNING OUTCOMES
Implement an API (Application Program 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) analyse the running time of (recursive) algorithms

ASSESSMENT
Assessment is based on a combination of labs, in-lab tests, and an exam.

ACADEMIC INTEGRITY LINKS
• Senate Policy on Academic Honesty -
  http://secretariat-policies.info.yorku.ca/policies/academic-honesty-senate-policy-on/
• Academic Integrity - http://lassonde.yorku.ca/academic-integrity

STUDENT LINKS
LAND ACKNOWLEDGEMENT

We acknowledge our presence on the traditional territory of many Indigenous Nations. The area known as Tkaronto has been care taken by the Anishinabek Nation, the Haudenosaunee Confederacy, the Huron-Wendat, and the Métis. It is now home to many Indigenous Peoples. We acknowledge the current treaty holders, the Mississaugas of the New Credit First Nation. This territory is subject of the Dish With One Spoon Wampum Belt Covenant, an agreement to peaceably share and care for the Great Lakes region.

The Indigenous Framework for York University: A Guide to Action can be found here: http://indigenous.info.yorku.ca/

Meaning of a land acknowledgement: http://healthydebate.ca/opinions/indigenous-land-acknowledgements

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

Moodle @ York University