The Objectives of 1011 are threefold: providing a first exposure to procedural programming, teaching students a set of soft computing skills (such as reasoning about algorithms, tracing programs, test-driven development), and demonstrating how computers are used in a variety of engineering disciplines. It uses problem-based pedagogy to expose the underlying concepts and an experiential laboratory to implement them. An integrated computing environment (such as MATLAB) is used so that students can pick up key programming concepts (such as variables and control flow) without being exposed to complex or abstract constructs. The problems are chosen with consultation with the various engineering disciplines in the Faculty with a view of exposing how computing is used in these disciplines. Two hours per week for instructor's lectures and three hours per week for lab work and tutorials. Prerequisites: None. Course credit exclusions: LE/EECS1541 3.00

The objectives of 1011 are threefold: providing a first exposure to procedural programming, teaching students a set of soft computing skills (such as reasoning about algorithms, tracing programs, test-driven development), and demonstrating how computers are used in a variety of engineering disciplines. It uses problem-based pedagogy to expose the underlying concepts and an experiential laboratory to implement them. An integrated computing environment (such as MATLAB) is used so that students can pick up key programming concepts (such as variables and control flow) without being exposed to complex or abstract constructs. The problems are chosen with consultation with the various engineering disciplines in the Faculty with a view of exposing how computing is used in these disciplines.

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>Smith, James A.</td>
<td>Sec. E / LECT / F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TOPICS AND CONCEPTS

Main Topics
1. The Computing Environment: Workspace, built-in commands, the debugger, unit testing, plots, etc.
2. Variables and Expressions: Types, operators, precedence, roundoff errors
3. Control Structures: Selection and Iteration
4. Encapsulation: Script files and functions
5. Computational Thinking: Process-based problem solving, unit tests as specification

Soft Computing Skills
1. Reasoning about algorithms
2. Tracing program
3. Test-driven Development

Applications
1. General Science and Mathematics
2. Engineering applications derived from the various engineering programs in the Faculty.
LIST OF LEARNING OUTCOMES AND EXAMPLES OF

By the end of the course, the students will be able to:
1. Use a set of soft computing skills such as reasoning about algorithms, tracing programs, and test-driven development for programming applications.
2. Explain and apply the fundamental constructs in procedural programming, including variables and expressions, control structures (conditionals/loops), and documentation.
3. Write simple programs using functions defined in m-files.
4. Use the computing environment to implement/simulate selected applications from science, math, and engineering.

GRADED ASSESSMENT

- Labs (8 at 3% each): 24%
- Lab Tests (2 at 10% each): 20%
- Flipped class assignments (approx. 5 at 2.8% each): 14%
- Midterm (1 at 17%): 17%
- Final exam (1 at 25%): 25%

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
- Student Rights and Responsibilities - http://oscr.students.uit.yorku.ca/student-conduct
- Religious Observance - https://w2prod.sis.yorku.ca/Apps/WebObjects/cdm.woa/wa/regobs
- Counselling and Disability Services - http://cds.info.yorku.ca/

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