Introduction to Programming for the Spatial Sciences

Instructor
Professor Grant McKenzie
613A Burnside Hall
grant.mckenzie@mcgill.ca
Office Hour: Fridays 12:00p-1:00p

Teaching Assistant
TBA

GENERAL INFORMATION
This course introduces students to conceptual and practical aspects of programming for the spatial sciences. The primary focus of this course is on developing a solid understanding of programming concepts and techniques irrespective of the specific programming language, framework, or software. Topics will include spatial data structures, flow control, classes & objects, and basics of geospatial data modeling and analysis. Students in this course will develop a proficiency in applying these programming principles to real-world geospatial problems. Supplemental to a conceptual and practical understanding of programming, students will be introduced to a number of leading commercial frameworks and cutting-edge open source tool-kits.

This course is open to any and all students with an interest in the spatial sciences and related disciplines. The prerequisite for this course is GEOG 201 or permission from the instructor. Students taking the course are expected to be familiar with geographic data formats and demonstrate a basic understanding of core geospatial concepts. No previous programming experience is expected.

LEARNING OUTCOMES FOR THE COURSE
Upon successful completion of the course students will be able to:

- Demonstrate an understanding of the complexity involved in developing computational models as abstractions of the real world.
- Demonstrate an understanding of the class-object model employed in object-oriented programming.
- Present a basic understanding of how geospatial (GIS & RS) software works ‘under-the-hood.’
- Automate geospatial processing tasks using the Python programming language.
- Organize, communicate, and solve theoretical and practical geospatial problems both individually and in a team environment.

COURSE SCHEDULE
- Wednesdays (10:00a - 11:30a) – Lecture
- Fridays (10:00a - 11:30a) – Demonstration & Hands-on Session
- Fridays (2:30p - 4:30p) – Lab Section
COURSE MATERIALS

- **Required textbook:**
- **Optional (physical) textbooks:**
  - Romalho, Luciano (2015) *Fluent Python: Clear, Concise, and Effective Programming*
  - Lawhead, Joel. (2013) *Learning Geospatial Analysis with Python*
- **Online resources:**
  - Online Python documentation: (Python version 3.x)
  - ESRI’s ArcPy Documentation
  - GDAL Cookbook
- **Technical materials:**
  - USB Flash drive (8GB minimum)
  - Laptop computer (optional but recommended) – Any operating system is fine

COURSE COMMUNICATION

The main course communication will be carried out through the course portal within the McGill University course management system (myCourses). All students enrolled in the course have access to the system. In addition to communications, this portal will be used by the instructor and the TA to post assignments and grades, and by the students to submit their assignments.

CLASS STRUCTURE

LECTURE COMPONENT

The lecture sessions will introduce concepts, techniques, analytic methods, and theoretical problems that are fundamental to understanding geographic data types, formats, and programming for the geographical sciences. Lecture attendance is mandatory.*

DEMONSTRATION & HANDS-ON SESSION

While new concepts will be introduced in the lecture component, the demonstration session will involve hands-on problem solving, demonstrations, writing code in-class, and some group discussion. A weekly problem set will be posted to the course website at the beginning of the week, and the demonstration sessions will involve going over example problems that will be useful for the assignments and potentially a student’s future career. Bringing your laptop is not necessary, but highly encouraged.

LAB ASSIGNMENTS

The course includes a lab component, which is of equal importance to the lecture and demo session. During the lab exercises students will work individually and in small groups to practice skills developed
in class within the course’s programming environment. All deliverables, as described in each lab assignment, should be submitted online before posted deadlines. Late assignments will be given a 0.007% penalty per minute (roughly 10% per day) up to a maximum of 5 days (including weekends). Assignment submitted more than 5 days late will be given a grade of 0. Please note that lab presence is not mandatory, though highly recommended. A teaching assistant will be available during the lab session to provide help students with the current assignment.

*Attendance will be taken randomly during 3 lectures throughout the semester. Students can miss one lecture without penalty. 2% will be deducted from a student’s final grade for non-official-document-supported absences after that. Lab attendance is not mandatory but help during office hours (Instructor and TA) is only available to students that attend lab sections (sign lab attendance sheet).

**PROVISIONAL SCHEDULE**

Please, note that modifications may be introduced to the schedule as the semester progresses. Updated schedules will be made available to all students via the course website as soon as possible.

<table>
<thead>
<tr>
<th>WEEK</th>
<th>DATES</th>
<th>TOPIC</th>
<th>LAB</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sep 4 / 6</td>
<td>Course overview &amp; Introduction</td>
<td>Lab 0: IDE (not graded)</td>
</tr>
<tr>
<td>2</td>
<td>Sep 11 / 13</td>
<td>Data types, Variables, &amp; Operators</td>
<td>Lab 1: The basics</td>
</tr>
<tr>
<td>3</td>
<td>Sep 18 / 20</td>
<td>Logic &amp; Control</td>
<td>Lab 2: Soil Sampling</td>
</tr>
<tr>
<td>4</td>
<td>Sep 25 / 27</td>
<td>Methods &amp; Scope</td>
<td>Lab 3: Population Growth</td>
</tr>
<tr>
<td>5</td>
<td>Oct 2 / 4</td>
<td>Input, Output, &amp; Exceptions</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Oct 9 / 11</td>
<td>Classes &amp; Objects</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Oct 16 / 18</td>
<td>Inheritance &amp; Polymorphism</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Oct 30 / Nov 1</td>
<td>Geo-semantics &amp; Ontologies</td>
<td>Lab 5: ArcPy &amp; Breweries</td>
</tr>
<tr>
<td>10</td>
<td>Nov 6 / 8</td>
<td>Geo-processing &amp; Modeling</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Nov 13 / 15</td>
<td>Analysis &amp; Automation</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Nov 20 / 22</td>
<td>Tool-kits &amp; Frameworks</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Nov 27 / 29</td>
<td>Data Mining &amp; Machine Learning</td>
<td>Take Home Exam</td>
</tr>
<tr>
<td>Dec ?</td>
<td>Take Home Exam Due during Final Exam Period</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

©2019 Grant McKenzie grant.mckenzie@mcgill.ca
GRADE DETERMINATION

- Lab Assignments: 50% (5 x 10%)
- Midterm: 25%
- Final Project: 25%

MID-TERM EXAM

There is one written exam in this course that is presented in Week 8 of the semester. The first seven weeks of the course introduce the fundamental building blocks for computer programming and data modeling. There are both theoretical and practical concepts that must be learned before continuing on to the second half of the course. A written midterm containing a combination of multiple choice, pseudo-code, and short answer questions will be presented to students. The exam will be closed-book, hand-written (no computers), and completed during allotted class time.

FINAL (TAKE-HOME) EXAM

A take home exam is one of the best ways to assess the knowledge gained by students taking a programming and conceptual modeling course. This exam mirrors the type of work that students may be presented with after graduating and places the student in a ‘project-deliverable’ type situation with access to all available real-world resources. The final exam deliverables consist of:

a) A final project report (50%)

b) Functioning project code with relevant documentation (50%)

Students will work in teams of two/three (depending on enrollment) and be expected to find their own (geographic) datasets.

EXPECTATIONS OF STUDENTS IN THE CLASS

- Students should be aware that most of the material covered in the class is not available in the course eBook and will be presented in lectures only. Students are strongly encouraged to take careful notes during the lectures as not all material will be presented in the slides.
- Students are expected to complete their lab assignments and attend lab sessions. All lab assignments are to be submitted via McGill’s myCourses by the specified due date and time.
- Students are expected to treat each other with respect. Disruptive behavior of any kind will not be tolerated. Students who are unable to demonstrate civility with one another, the teaching assistants, or the instructor will be subject to referral to the Office of Student Conduct or to the McGill Campus Security. You are expected to adhere to the Code of Student Conduct.
- In this class, students will be allowed and encouraged to use their personal computers or other means of technology to take class notes and problem sets. Use of technology for tasks other than those related to the course is not permitted.