CISC 121: Introduction To Computing Science I  
Fall 2014 Syllabus and Administrative Policies  
last updated Sep 9, 2014

This syllabus is a brief summary of the most important administrative information for CISC 121. For complete details, you must read to the course web page: http://research.cs.queensu.ca/home/cisc121/2014f/home.html. The course will also have a Moodle area which will be available before the start of classes. Students must read the "news" and "general" Moodle forums regularly, as they will contain important information.

Instructor: Margaret Lamb  
office: Goodwin 554  
office phone: 613-533-6059  
home page: http://cs.queensu.ca/home/malamb  
office hours: may vary during the semester; refer to home page for current details  
e-mail: malamb@cs.queensu.ca

Pre or Co-requisite: Any first-year course in Mathematics. If you've taken a university-level math course and SOLUS isn't accepting it, please talk to the CISC 121 instructor.

Pre-requisite: Some programming experience (such as high-school level programming or CISC 101/3.0 or CISC 110/3.0) will be expected.

Students who have not passed CISC 101 or CISC 110 will be given a screening test during the first week to evaluate their readiness for CISC 121. You must either pass the screening test or move into CISC 101. More details on the web page.

Calendar Description:  

Purpose: This course is designed to give students an introduction to many important techniques and ideas necessary for Computer Science. It is a pre-requisite for almost every other course in our Computing programs at Queen's. It is not just a "how to program" course, but programming is an important part of the course as students will be asked to apply the ideas in this course to programming problems.

Learning Outcomes: By the end of the course you should be able to:

- analyze a small problem and design a Python program to solve it, separating the program into multiple functions (including parameters and return results)
- use Python lists, strings and dictionaries
- discuss and use good programming practices, including top-down design, documentation, readable style, careful testing and debugging
- implement and use some simple data structures, including linked lists, stacks, queues, and binary search trees
- predict the outcome of a recursive function and design recursive functions to solve problems
- be able to explain, reproduce or modify several searching and sorting algorithms
- interpret complexity notation ("big O") and be able to determine the complexity of a simple iterative or recursive algorithm
- discuss at a high level how digital computers store numbers and the consequences for simple examples of numerical programming

Required Textbook: The textbook for this course is Think Python: How to Think Like a Computer Scientist, by Allen B. Downey. You can use the online version from www.greenteapress.com/thinkpython or buy a hard copy from the campus book store. There will also be assigned readings from course notes published on Moodle.

Course Organization: This course is divided into several topics. For most topics there will be assigned readings from the textbook or other sources and questionnaires on Moodle for students to answer questions about those readings. Lectures will not review the readings in detail but will focus on examples to illustrate the techniques and ideas in the readings and to demonstrate good programming techniques. Students will be given several marked assignments as well as many unmarked practice problems to work on.

Course Schedule: Available on the web page. Students will have adequate warning of all quizzes and assignment dates.
Marking Scheme:
The basic marking scheme is as follows:
- assignments: 10%
- quizzes (average of the best 2 of 3): 46%
- final exam: 40%
- participation: 4%
- total: 100%
+ up to 2 possible bonus points

However, students must achieve a passing grade on their quizzes and final exam (at least 43 out of the 86 points allocated to those marks) to pass the course. Full details on how students will be evaluated are available on the web page.

Lecture Times: (slot 2)
- Mondays 9:30-10:20
- Wednesdays 8:30-9:20
- Thursdays 10:30-11:20

All of our lectures will be in BioSci 1102.
There will be no lecture on Thanksgiving Day (Monday, October 13)

Labs: Each student must sign up for and attend one of the following labs:
- Lab 002: Mondays 3:30-5:20
- Lab 003: Tuesdays 2:30-4:20
- Lab 004: Tuesdays 11:30-1:20

Labs will take place in Jeffrey 155. There will be no labs during the first week of classes or on Thanksgiving Day (Monday, October 13).

Disability Accommodations:
Queen's University is committed to achieving full accessibility for persons with disabilities. Part of this commitment includes arranging academic accommodations for students with disabilities to ensure they have an equitable opportunity to participate in all of their academic activities. If you are a student with a disability and think you may need accommodations, you are strongly encouraged to contact the Disability Services Office (DSO) and register as early as possible. For more information, including important deadlines, please visit the DSO website at [http://www.queensu.ca/hcds/ds/](http://www.queensu.ca/hcds/ds/).

If you have a disability and require accommodations in this course, please see the course web site for more specific information.

Academic Integrity: For the full statement of Queen's Academic Integrity policy, please see Arts & Science Academic Regulation 1 at [http://www.queensu.ca/artsci/academic-calendars/regulations/academic-regulations/regulation-1](http://www.queensu.ca/artsci/academic-calendars/regulations/academic-regulations/regulation-1). You are responsible for familiarizing yourself with these principles, which apply to all Queen's courses.

More details about how Academic Integrity applies to this course may be found on the course web site. Please be warned that this instructor takes Academic Integrity quite seriously and WILL make formal charges if violations are detected. These charges will affect your mark and possibly your status at Queen's.

Tips For Doing Well in CISC 121:
- If you don't have much programming experience, consider taking CISC 101 first. Yes, you can still finish a Computing degree in four years if you start with 101.
- Read all of the assigned readings carefully and on time.
- Attend lectures and labs regularly.
- Get an early start on the assignments; don't leave them until the last minute.
- Do the practice problems as well, even though they will not be marked. Practicing is the best way to make sure you understand the course material and can apply it on your quizzes and exam, which are marked.
- When you have a question or don't understand something, don't be shy about asking! Help is available for you; this is part of what you're paying tuition for. Everyone has questions from time to time. The web site contains many suggestions about ways to get help.