CMPT 120 - D100 Intro.Cmpt.Sci/Programming I

Instructor(s): Diana Cukierman

Calendar Objective/Description:
Intro.Cmpt.Sci/Programming I

Instructor's Objectives:
An elementary introduction to computing science and computer programming, suitable for students with little or no programming background. Students will learn fundamental concepts and terminology of computing science, acquire elementary skills for programming in a high-level language and be exposed to diverse fields within, and applications of computing science.

* CMPT 120 PLACEMENT TEST – do you have enough computing knowledge to pass our placement test? Have you taken Computing Science courses in High School or elsewhere? Have you worked in a business or volunteered as a computer programmer? Have you written programs of 200-300 lines of code or more? IF YES you may NOT have to take CMPT 120 and could enroll directly into the next required courses – CMPT 125 and 127. Challenge yourself and take our placement test. https://courses.cs.sfu.ca/forms/cmpt-cmpt-120-placement-test/ IF NO – please go ahead and enroll in CMPT 120.

This course will be offered in person, face to face. Some assessments (such as coding exercises) will be online, some assessments may be in person, possibly including paper based tests. Details of technology and tools needed in class and for tests will be provided the first class of the semester.

All the course information and communication will be centralized in the Canvas course website which will be available by the first day of class, including class materials, readings, assignments, and points. Students should ensure that they receive notifications when announcements are posted on Canvas.

Prerequisites:
see go.sfu.ca

Topics:
- Algorithms and computational thinking
- Procedural programming in Python
- Data types and control structures
- Application areas within computing science
- Fundamental algorithms, including searching, sorting, basics of recursion

Midterm Date: The CMPT 120 midterm exam will be held on Tue. June 28, 2022 6:00 pm to 8:00 pm PST. All students enrolled in this course are expected to be available during this time. If you experience a course conflict when registering for this course on goSFU due to the midterm, see the CS FAQ landing page for procedures to enroll in this course.
- Computability and complexity, introduction
- Basics of binary encoding

**Grading:**
There will be assignments and multiple quizzes/exams. A more detailed marking scheme and technology details will be provided in the first class of the semester.

Students must attain an overall passing grade on the weighted average of exams in the course in order to obtain a clear pass (C- or better).

**Required Books:**
Think Python - How to Think Like a Computer Scientist: Interactive Edition https://runestone.academy/runestone/books/published/thinkcspy/index.html, This interactive text is available online for free

**Reference Books:**
Computer Science Illuminated, Nell Dale, John Lewis, Jones & Bartlett, 2012, 9781449672843
Starting out with Programming Logic and Design, Tony Gaddis, Pearson, 2015, 9780133985078

**Academic Honesty Statement:**
Academic honesty plays a key role in our efforts to maintain a high standard of academic excellence and integrity. Students are advised that ALL acts of intellectual dishonesty will be handled in accordance with the SFU Academic Honesty and Student Conduct Policies (http://www.sfu.ca/policies/gazette/student.html).