



ACADEMIC YEAR: 2017/2018

Course Number COMP201		Course Title Introduction to Computing	
Department Computer Science and Software Engineering	Semester Fall 2017	Type Core	Credits 3.00
Level Undergraduate	Prerequisites MATH 201 or equivalent		
Schedule Class – Mon @ 17:45 – 20:15, GN M100 Tutorial – Tue [H907], Tue [H917] @ 20:30 – 21:20			
Instructor Charalambos Poullis	Office Hours Wed. @ 14:00-16:00 and by appointment	Office EV3.183	Email charalambos@poullis.org
Teaching Assistant Timothy Forbes Chen [Jocelyn] Qiao	Office Hours Mon @ 20:30 – 21:20 and by appointment	Office EV9.113 EV9.119	Email comp201.ta@gmail.com

COURSE DESCRIPTION

This course is an introduction to computer programming using a high-level programming language [Python]. In particular, the course aims at the development of software design and problem-solving skills, and the cultivation of algorithmic and analytical thinking in solving complex problems, all part of the process of software development.

COURSE OBJECTIVES

The primary objective of the course is to provide a comprehensive introduction to computer programming and the related programming principles required when designing and developing software. More specifically the course aims to cover the following:

- Introducing the basic programming principles, algorithmic thinking, and data structures.
- Designing, developing, testing and debugging of software.
- Analyzing and assessing alternative [algorithmic] solutions to a problem.
- Learn a high-level programming language and gaining practical experience in developing software.

LEARNING OUTCOMES

By the end of this course, students will be able to:

- identify and explain the following basic concepts of structured programming: values, data types, variables, values, data types, variables, parameters, arguments, functions, scope, structure and control flow of programs, libraries/modules, data abstraction, hiding information, interfaces.
- identify and explain the following basic algorithmic structures that are used in the creation of programs: sequence/succession, iteration, selection, functions.
- distinguish the basic: (a) programming principles; (b) algorithmic techniques; and (c) program structures.
- demonstrate skills in problem-solving using procedural and/or object-oriented programming.
- design, implement, test (in an organized and systematic manner), debug and document modular (small) programs.
- compare, criticize and assess on the correctness and general behavior of a program and evaluate alternative (algorithmic) solutions for the same problem based on qualitative and other criteria.
- demonstrate the ability for algorithmic thinking regardless of programming language.
- design and develop software using the high-level programming language Python.

COURSE CONTENTS

An brief overview of the course contents:

- Introduction to programming beginning with the use of high-level programming language Python
- Image processing
- Audio editing
- Text editing for the Internet
- Video creation
- Introduction to object-oriented programming
- Tutorial: one hour per week.

TEACHING METHOD

The course comprises of weekly lectures and practical training; both in the form of labs and individual assignments. ***It is emphasized that attendance in lectures and labs is mandatory for learning and performing well in this course.***

ASSESSMENT

Programming Assignments (x3)	3x15% [45%]
Quizzes	2x25%
Participation [Class & Tutorials]	5%

All assignments and quizzes must be completed in order to pass the class.

Programming Assignments: The goal of the programming assignments is to gain practical experience in programming with Python. There are three programming assignments. Please refer to the schedule for the assignment descriptions and due dates. All assignments must be completed to pass the course. The assignments *may* have a small amount of bonus credit. All assignments must be done **individually** and ran during the lab session for evaluation. Lab instructors will test your knowledge of programming the assignment during this evaluation.

Quizzes: Quiz #1 will take place on **October 23rd** and, Quiz #2 will take place on **December 04th**.

Participation: Class participation is an important component of the grade. This involves attending the lectures and labs regularly, asking and answering questions, and participating during the labs.

Submission: The assignments should be submitted by **1700 on the day they are due**. Late submissions will be accepted until 3 days following the initial deadline, however, there will be a penalty of 20% from the total assignment grade and [if applicable] no bonus.

RECOMMENDED TEXTBOOKS

1. Introduction to Computing and Programming in Python (Global Edition) by Mark J. Guzdial, Barbara Ericson. **ISBN-13:** 978-1292109862
2. Python Multimedia: Beginner's Guide by Ninad Sathaye. **ISBN-13:** 978-1849510165

COURSE SCHEDULE

The list below provides a summary of the material that will be covered during the course as well as a tentative schedule. Labs will support topics covered in the lectures and provide hands-on exercises.

Session	Topic	Reading	Notes
01. SEP 11	Syllabus/Introduction to the course Introduction to computing	Ch. 1, 2	
02. SEP 18	String manipulation, arrays, function	Ch. 3	Assignment 1 out
03. SEP 25	Multi-dimensional arrays, iteration	Ch. 4	
04. OCT 02	Conditionals	Ch. 5	
OCT 09	Thanksgiving day – University closed		Assignment 1 due
05. OCT 16	Nested iterations Assignment #1 solution	Ch. 6	Assignment 2 out
06. OCT 23	Quiz #1		
07. OCT 30	Arrays cont'd, iterations cont'd	Ch. 7	Assignment 2 due Assignment 3 out
08. NOV 06	Functions, iterations Assignment #2 solution	Ch. 8	
09. NOV 13	Functions cont'd	Ch. 9	
10. NOV 20	Modular programming	Ch. 11	Assignment 3 due
11. NOV 27	Assignment #3 solution Review		
12. DEC 04	Quiz #2		
13. DEC 05	Quiz #2 solution		Make-up class for OCT 09

COMMUNICATION

This term we will be using Piazza for class discussion. The system is highly catered to getting you help fast and efficiently from classmates, the TA, and myself. Rather than emailing questions to the teaching staff, I encourage you to post your questions on Piazza. If you have any problems or feedback for the developers, email team@piazza.com.

Find our class page at: https://piazza.com/concordia_university/fall2017/comp201/home

ACADEMIC INTEGRITY

There is a plethora of online resources for Python. You are allowed to incorporate code or tips you find on the Web, provided this doesn't make the assignment/project trivial **and** you explicitly acknowledge your sources. You are allowed to discuss assignments with each other, but coding must be done individually.

Please make sure you familiarize yourself with Concordia's Academic Code of Conduct:
<http://www.concordia.ca/students/academic-integrity.html>