Boston University Metropolitan College

Introduction to Programming

MET CS 201 C1 (On Campus)

Fall Term (Wednesday, September 5 through Wednesday, December 12)
Wednesday 6:00pm-8:45pm in SHA 201

Instructor

John Keklak
jkeklak@bu.edu
Office hours: by appointment

Course Description

This course provides an introduction to computer programming. While the skills taught in this course apply to any programming language, the lectures and homework exercises will utilize the Python language.

The Python language features to be covered include, but are not limited to: flow of control constructs such as 'if' and 'while' statements; functions; file input and output; strings; lists, tuples, sets and dictionaries; comprehensions; classes and object-oriented programming; recursion; graphical user interface programming.

The course also requires students to formulate algorithms to solve certain types of problems (for instance, to efficiently locate a value in a list), to write clear and efficient Python code to implement these algorithms, and to produce fully-tested, debugged and working programs.

Class time will be divided between lecture and in-class exercises.

Homework exercises will consist mostly of programming. Students will be required to meet with the instructor to demonstrate their working programs, and to explain the thinking behind the code they write.

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

(1) explain basic concepts of how computers operate
(2) explain how to design an algorithm to solve certain types of problems
(3) write short programs in Python without referring to documentation

(4) explain the nature of the Python features covered in this course, and provide examples of their use.

This course is designed for students who are interested in learning to program. No prior programming experience is required, but is strongly recommended. Good algebra and logic skills are strongly recommended. This course will move quickly through a wide range of programming language features that students will be expected to master quickly. Please contact me before registering if you would like to better understand the depth and pace of this course.

Laptops are required for class exercises at each class.

Books

The Python Software Foundation on-line documentation

Courseware
TBA

Class Policies

1) **Attendance & Absences** – Due to the rapid pace of the course and the volume of material to be covered, attendance is essential. Attendance will be taken at the beginning of each class. Absences will affect your grade significantly.

   I also ask that you arrive to class on time, since it is highly disruptive to have students flowing in throughout the class period. Late arrivals will also adversely affect your grade.

2) **In-Class Exercises on Laptops** – A portion of class time will be reserved for in-class exercises to explore the Python features covered in the lecture portion of the class. All students will be required to bring laptop computers suitable for performing these exercises. During the in-class work, students will have the opportunity to confer with the instructor, and may be randomly called upon to demonstrate their work to the instructor.

3) **Assignment Completion & Late Work** – With the very rapid pace of the course, and because solutions will be discussed in class on the day assignments are due, it is extremely important that homework be completed on time. Late homework will be levied a penalty of 30 points if submitted after the homework is discussed in class. An additional 10 point penalty will be levied each day the assignment is late (up to 3 days).

After 3 days, no credit will be given. It is advisable to complete homework even for no credit, since the homework exercises are essential preparation for exams.

4) **Academic Conduct Code** –Cheating and plagiarism will not be tolerated in any Metropolitan College course. They will result in no credit for the assignment or examination and may lead to disciplinary actions. Please take the time to review the Student Academic Conduct Code: http://www.bu.edu/met/metropolitan_college_people/student/resources/conduct/code.html.

NOTE: This should not be understood as a discouragement for discussing the material or your particular approach to a problem with other students in the class. On the contrary – you should share your thoughts, questions and solutions. Naturally, if you choose to work in a group, you will be expected to come up with more than one and highly original solutions rather than the same mistakes.

Grading Criteria

Exam Dates

Quizzes	One per module; one module every two weeks
Final	Wednesday, August 8

Grades will be determined by the following weighting:

Quizzes	-	50%
Final	-	50%
Instructor's discretion	-	+/-10%

Incompletes will not be given.

Each student may view their grades on their individual grade sheet (link to be provided).

Grade conversions:

A	93-100

A-	90-92.999…
B+	87-89.999…
B	83-86.999…
B-	80-82.999…
C+	77-79.999…
C	73-76.999…
C-	70-72.999…
D	60-69.999…
F	59.999… and below

Grades are not curved.

Class Meetings, Lectures & Assignments

Lecture topics and assignments are subject to change, and will be announced in class as applicable within a reasonable time frame.

Date	Topic	Assignments
September 5	Module 1: Introductions, Computer programming in general; install Python; first Python program; printing	TBA
September 12	Module 1 (con't)	TBA
September 19	Module 2: Lists and strings; basic math; basic looping	TBA
September 26	Module 2 (con't)	TBA

October 3	Module 3: More math; random numbers; more on strings and lists; ‘if’ and ‘while’ logic; intro to functions; applications	TBA
October 10	Module 3 (con’t)	TBA
October 17	Module 4 Inside the computer; memory; narratives; reading code; debugging code	
October 24	Module 4 (cont’t)	
October 31	Module 5 Tuples and dictionaries; key-value concept; applications	
November 7	Module 5 (con’t)	
November 14	Module 6 More on functions; recursion; applications	
November 21	No class	

November 28	Module 7 Applications; graphing; stochastic simulations; graphics	
December 5	Module 7 (con’t)	
December 12	Special topics Where to go from here; classes and object-oriented programming; web programming; IT services; other languages; software development and software engineering; review	
Finals Week	Final date and time TBD	