

CS 100 Roadmap to Computing

Course Syllabus, Fall 2016

Course Description

An introduction to programming and problem solving skills using Python, a very high level language. Topics include basic strategies for problem solving, conditional, repetition, function and other constructs that control the flow of execution of a program and the use of high level data types such as lists, strings and dictionaries in problem representation. The course also includes a project in which the student investigates and reports on a topic of current interest in computing.

Computing is a profession that requires lifelong learning, which is pursued through activities and using types of materials that are similar to those employed by students. In this course, the student, in addition to mastering the programming and problem solving materials, is expected to learn to effectively use learning strategies and materials – learning how to learn efficiently in preparation for a knowledge intensive profession. This includes effective use of knowledge resources -- reading documentation, asking and answering peer questions, consulting with more experienced persons, and searching on-line for answers. It also includes tools and methodology – testing to verify the correctness of code, use of an integrated development environment (IDE) and debugger, writing specifications and documentation.

Learning this material requires extensive hands-on practice. You should plan to spend twice as much time studying and working problems outside of class (that is, about 6 hours a week) as you do in class.

Course resources

Textbook: Ljubomir Perkovic, *Introduction to Computing Using Python, second edition*. There are print and digital editions of the book, at different prices.

Other course materials:

- Python language V 3.4 can be gotten at <https://www.python.org/downloads/> This includes the IDLE development environment, help files, modules and other parts of the standard distribution. You will need to get Python and install it on your personal desktop and/or laptop computer. You can download Python for Windows, Mac or Linux environments. There is no charge for Python.
- PythonTutor, a program for stepping through and visualizing the execution of Python code at pythontutor.com
- The Wingware IDE, which can be downloaded for various platforms at <http://wingware.com/downloads/wingide-101> .
- Optional materials: codecademy.com/tracks/python and various other online resources.

Class attendance

Class attendance is mandatory. A student who misses more than five classes will be dropped, without credit. Getting to class late or leaving early counts as half an absence.

Weekly individual recitation

Meeting with an instructor or classroom assistant weekly is a mandatory part of the course. This is an important checkpoint in assuring your grasp of the material being covered and correctly solving assigned problems. When you go for recitation you should have already read the assigned material and worked on current homework. In recitation, will demonstrate your solutions to homework problems or ask the questions you have identified that need clarification. This will check whether your understanding of this material is correct. A student who misses more than three recitations will be dropped, without credit. You may meet with the instructor or classroom assistant for your section or for another section. All instructors and classroom assistants have posted office hours.

Homework

Homework must be submitted through Moodle (and when specified in hard copy) by the beginning of class on the due date. It will not be accepted late except for special circumstances (such as jury duty or medical problem), for which you must provide documentation.

A homework assignment will typically require you to write code that produces a specified output. No credit will be given for code that does not run. Getting a correct solution will often require that your solution be written, tested, and then rewritten multiple times until it fulfills the specification. Expect that the bulk of your time will be spent getting it right. Remember: only code that is correct is worth anything. During the write-test-debug cycle you may – and are encouraged – use the debugging facilities in the development environment, pose questions on Moodle, and discuss the problem with others.

Roadmap project

Each student will work on one or more Roadmap projects, consisting of a written and an oral presentation, either individually or with a partner. (Partners will be expected to produce two projects.)

Class participation

Presenting your homework answers and presenting your projects in class is a regular part of the course. Asking and answering questions, taking quizzes, solving programming problems – individually or in groups – is a regular part of class meetings.

Cell phones must be turned off during class. During class time you may not play games, text, email, browse the web or engage in other activities that are not part of the class.

Course communication

Moodle (<http://moodle.njit.edu/>) will be used to post lecture notes, to submit homework and for course discussion. You may also email instructors and classroom assistants.

Collaboration and individual responsibility

You are encouraged to study and to work on assignments together with others; collaboration is a basic learning technique. You may not take credit for the work of others. You must understand and be able to explain all work that you submit.

What You Will Learn

By the end of this course, you will be expected to know and be able to use these pieces of the computing toolkit to compute the solution of a specified problem:

- Devise a problem representation (model) and a sequence of steps (algorithm) that correctly solve the problem posed
- Write a program that implements the algorithm, using
 - A core set of Python language elements (keywords, syntax, variables, modules).
 - Basic data types (integers, floats, strings, booleans, lists, dictionaries, files) and operations on them
 - Statements that perform input and output
 - Statements that control the sequence of execution of a computer program (if/else, for, while, function call/return).

Each homework assignment gives you practice on these concepts and skills, and provides feedback on your progress. You are expected to submit working solutions to every homework assignment. Each element of this course builds on previous material, and any gaps in your understanding will compromise your ability to successfully complete the course. You understand material when you are able to use it to solve problems and to explain your solutions. Each of the two midterm exams and the final exam test your mastery of the material.

Topics to be covered

Starting to code – Python and IDLE

Built-in data types

Variables and expressions

Conditional execution (if/else)

Functions and methods

Modules (turtle, math, random, url)

Passing parameters / Returning values

Iteration (for and while)

Data files

Debugging and testing

Scope and namespaces
User input

Grading Formula

Homework 10%
Attendance at office hours 4%
Midterm 1 and 2, 20% each
Final Exam 30%
Roadmap Projects 10%
Misc 6%

In addition to an overall course score, a minimum grade must be achieved on the final to pass the course.

Exams

There are two common midterms on Monday, October 3 and Monday, November 7, 4:00-5:45pm. Final exam period is December 16-22 – the exact day has not yet been set. ***Be sure that you will be present for all of your final exams.***

You must bring ID to all exams. Students with special needs are advised to make arrangements with Disability Services.

There are no makeup exams. If you miss an exam because of a documented special circumstance you may receive an imputed grade based on the other midterm.

If you believe that you deserve more credit than you have been awarded on a particular common exam problem, you may request, within 48 hours of the exam being returned, that it be regraded. Your entire exam will be regraded, which may result in points being added or subtracted.

Exams do not require any portable electronic devices, such as a mobile phone or calculator. It is preferable that you do not bring any such device to the exam, but if you do bring one you must leave it with the proctor during the exam.

You should read the [University Code on Academic Integrity](#). It describes infractions of academic integrity and penalties for violations, including, for the most serious violations, an XF grade in the course or expulsion. All work that you represent as your own must, in fact, be your own. Work done by others must be given proper credit.

Students will be informed of any modifications of the syllabus during the semester.