

Introduction to Computer Science II

Syllabus

CSC 242-604

Spring 2019-2020 (v2)

Class sessions: Weekly via Panopto

Labs: Wednesday, 10:10am - 11:40 pm, Zoom

Instructor: Anthony Zoko

Lab instructor: John Lynch

Contact information

azoko@cdm.depaul.edu

Office hours

By appointment 1 day in advance using Zoom

Course web site

All information for this course is posted to the Desire 2 Learn (D2L) site. To log onto the D2L page visit <https://d2l.depaul.edu/>. Class notes, programming assignments, lab assignments, study guides for the midterm and final exam, and other course materials will be available through the D2L site. There will also be links to course recordings which are useful for reviewing material.

Prerequisites

You must have taken CSC 241: Introduction to Computer Science II or an equivalent course that introduces problem-solving techniques and programming in Python and earned a passing grade (C- or better). I will also assume that:

- You know how to create, debug, compile, and run Python, and you use a reasonable coding style (i.e. your code is easy to read and relatively concise)
- You know Python's basic control structures and types
- You can solve basic algorithmic problems

Course topics and learning goals

This course is the second of a two-course sequence introducing computer science skills, including problem solving, algorithm development, recursion, and programming using Python. In this course, we will apply these skills in several application areas of computer science: graphical user interface (GUI) development, database development, and Internet and distributed computing. The concept of a class and object-oriented programming will be motivated and introduced.

After you have taken this class:

- You will strengthen your Python programming skills
- You will know how to design classes and understand the fundamental principles of object-oriented programming
- You will be able to design basic graphical user interfaces
- You will be able to apply recursion as a problem-solving and programming technique
- You will be able to write simple Internet client programs
- You will have a basic understanding of the database API

Course calendar

The following gives all the important dates for this course. The topics covered are subject to change.

Week	Date	Topic/Deadline
1	Week of Tuesday, March 31st 2020	Introduction to the course, a review of namespaces and scope, and using operators and constructors, Object Oriented Programming
		<i>Friday April 3rd- Last Day to Add Classes</i>
2	Week of Tuesday, April 7 th , 2020	Object-oriented programming
		<i>Monday April 13th - Last Day to Drop Classes no penalty</i>
3	Week of Tuesday, April 14th 2020	Object-oriented programming
4	Week of Tuesday, April 21 st 2020	Graphical user interface development
5	Week of Tuesday, April 28th 2020	Graphical user interface development
6	Week of Tuesday, May 5 th 2020	MIDTERM from 10:10am to 11:40am via D2L Quiz
7	Week of Tuesday, May 12 th , 2020	Recursion
		<i>Friday February 21st -Last day to withdraw from classes</i>
8	Week of Tuesday May 19 th 2020	Recursion and searching
9	Week of Tuesday, May 26th, 2020	Introduction to HTML and parsing
10	Week of June, 2 nd , 2020	Crawlers and Final Review
11	June 11 th 2020 from 8:30am to 10:45am	FINAL EXAM via D2L Quiz

Textbook

The required textbook for the course is **Introduction to Computing using Python: An Application Development Focus, Second Edition**, Ljubomir Perković, John Wiley & Sons, 2015. *Make sure you have the electronic version of the text* since it contains case studies that we will be using. The electronic text has ISBN 978-1-118-89105-6. You can buy the eBook directly from the publisher if you like: <http://www.wiley.com/WileyCDA/WileyTitle/productCd-EHEP003201.html#student>

Grading policy

Course assessments include lab attendance, programming assignments, and a midterm and final exam. The course grade will be computed as follows:

Assessment	Percentage
Labs	10 %
Programming assignments	25%
Midterm exam	32 %
Final exam	33 %

In order to do well in this class, you must attend the lectures and labs regularly, participate in class discussions, read the chapters in the book as indicated in the homework assignment, start work on the assignments early, and ask questions early and often. The answers to the programming assignments and the lab and exam questions should be written in a way that is rigorous, clear, and concise.

Grading Scale:

94-100: A

89-93: A-

85-88: B+

80-84: B

75-79: B-

70-74: C+

65-69: C

60-64: C-

55-59: D+

50-54: D

0-49: F.

Lab attendance and exercises

Each week you will have a lab session conducted by our teaching assistant. Your attendance at the lab session and completion of lab exercises is required and will count for the portion of the grade indicated above. You are encouraged to work with classmates in completing the lab, although you must credit anyone you worked with during the lab on your lab submission. No late lab submissions are accepted for any reason. Your lowest lab score will be dropped in the calculation of your course grade.

You may decide to not come physically for up to two lab sessions but must have the lab submitted by the time the lab is due. After two missed lab sessions, it is an automatic 0 for that lab.

Programming assignments

Each week you will have a programming assignment. You can consult with your homework partners, the teaching assistant, the instructor, and the CDM tutors on the programming assignments, but you may not under any circumstances submit code that you have not helped to write nor may you consult anyone beyond those specified when completing your assignments. Each programming assignment will have a posted deadline, specified on the assignment. No late assignments are accepted for any reason. Your lowest assignment score will be dropped in the calculation of your course grade.

Midterm and final exams

The midterm and final exams will be cumulative. Both exams will be conducted in a lab and will require you to write Python code. You will be required to work on the lab machines, and you cannot use your own laptop. The exams are individual assessments, and you may not work with anyone on the exams.

Make-up exams will not be given. If you wish to petition for a make-up exam, you must notify me in advance and provide documented evidence of the emergency that will cause you to miss the exam. Failure to contact me in advance of the exam date and time will disqualify you from being allowed to take a make-up exam. If a make-up exam is granted, it will be of a form of my choosing.

Online course evaluations

Evaluations are a way for students to provide valuable feedback regarding their instructor and the course. Detailed feedback will enable the instructor to continuously tailor teaching methods and course content to meet the learning goals of the course and the academic needs of the students. They are a requirement of the course and are key to continue to provide you with the highest quality of teaching. The evaluations are anonymous; the instructor and administration

do not track who entered what responses. A program is used to check if the student completed the evaluations, but the evaluation is completely separate from the student's identity. Since 100% participation is our goal, students are sent periodic reminders over three weeks. Students do not receive reminders once they complete the evaluation. Students complete the evaluation online in Campus Connect: <http://campusconnect.depaul.edu/>

Academic integrity

The course adheres to the DePaul University's Academic Integrity Policy. For complete information about Academic Integrity at DePaul University, please see: <http://academicintegrity.depaul.edu/>.

Cheating is any action that violates university norms or instructor's guidelines for the preparation and submission of assignments. This includes, but is not limited to, unauthorized access to examination materials prior to the examination itself; use or possession of unauthorized materials during the examination or quiz; having someone take an examination in one's place; copying from another student; unauthorized assistance to another student; or acceptance of such assistance. Plagiarism involves the presentation of the work of another as one's own. Plagiarism includes, but is not limited to the following: the direct copying of any source, such as written and verbal material, computer files, audio disks, video programs or musical scores, whether published or unpublished, in whole or part, without proper acknowledgment that it is someone else's; copying of any source in whole or part with only minor changes in wording or syntax, even with acknowledgment; submitting as one's own work a report, examination paper, computer file, lab report or other assignment that has been prepared by someone else (including research papers purchased from any other person or agency); the paraphrasing of another's work or ideas without proper acknowledgment; working so closely with another person other than those authorized so as to produce identical code.

The use of others' web/publication content (text, graphics, code) is regarded as plagiarism if credit is not given (see the above description of plagiarism). When you directly quote someone's work, you must put it in quotation marks. Without such quotations and reference, it is regarded as an act of plagiarism (see the above description of plagiarism). Using materials that the student prepared for other purposes (e.g., for another course or for his/her work) needs the course instructor's prior permission.

A charge of cheating and/or plagiarism is always a serious matter. It can result in an automatic F in the course and possible expulsion.

Incomplete

An incomplete grade is given only for an exceptional reason such as a death in the family, a serious illness, etc. Any such reason must be documented. Any incomplete request must be made at least two weeks before the final, and approved by the Dean of the College of

Computing and Digital Media. Any consequences resulting from a poor grade for the course will not be considered as valid reasons for such a request.