CS161 Introduction to Computer Science

Winter 2019, CRN 30165 & 30540

Instructor

Joe Paris
Email: parisj@linnbenton.edu
Office: MKH-116
Office Hours: M/W 11:30–12:20, Tu/Th 9:00–9:50, or by appointment

Course Description

CS161 introduces the principles of computer programming using an object oriented language. It includes problem-solving concepts, verification and validation, representation of numbers, sources of errors, debugging techniques and algorithm development. The Java programming language is used.

Prerequisites

- CS160 Orientation to Computer Science, with a 'C' or better
- MTH95 Intermediate Algebra or equivalent with a 'C' or better

Outcomes

Upon completion of the course you will be able to:

- Demonstrate an understanding of objects in the context of a object-oriented programming language.
- Demonstrate the use of good program development, debugging techniques and documentation.
- Write object-oriented code that includes control statements, while loops, for loops, output to the screen and input from the keyboard and from a file.
- Write, compile and run simple web-based and desktop-based GUI applications using components and containers.
- Write simple, user-designed classes that demonstrate an understanding of encapsulation.
- Write object-oriented code that includes the use of common data structures.

Required Text

Python Programming: An Introduction to Computer Science, 3rd Edition by Zelle; ISBN 978-1590282755

Required Materials

USB thumb drive with a minimum capacity of 4GB.

Internet access.

Python 3.6+ (https://www.python.org/). A well-written guide for installing Python on Windows, Mac, and Linux can be found at https://realpython.com/installing-python/.

A text editor of your choice. Thonny (https://thonny.org/)is an excellent beginner's editor for Python. Visual Studio Code (https://code.visualstudio.com/) is installed on the classroom computers. Both will be used for in-class demonstrations. Other good, Python-oriented options include PyCharm (https://www.jetbrains.com/pycharm/, available with a free license for students) and Spyder (https://www.spyder-ide.org/).

Class Policies

A "normal" class generally asks you to read pages from your textbook(s), then attend a lecture where your instructor covers that same material before giving you an assignment that you are expected to complete on your own time. This class will be different, utilizing a *flipped classroom* approach to learning. Instead of the traditional model above you will still be given reading assignments to do outside of class, however class time will be devoted to working on various projects to reinforce what you have learned. Instead of hours of lecture your instructor will facilitate your work as you do it in class.

This model of instruction means that it is imperative you complete all reading assignments that are given *before* class. Every class period will start with a short quiz over the material you read outside of class to gauge how well the class as a whole understood it. Further discussion or demonstration will be given for any material that needs it followed by an assignment which you will begin working on, in class, while you have access to both your instructor and your peers. Some of these assignments will be individual, some will be small group, all will be collaborative.

Because of the nature of the course you will get the most from it only if you attend every day. Class discussions will likely diverge into material not found in the assigned reading or viewing. Additionally, a lot of your learning will involve class discussion which can only happen if you are present.

All work, unless specifically stated otherwise, is to be submitted electronically. Details of how to do this will be covered in class.

Students come from many different backgrounds and have unique life experiences which can enhance class discussions. For this reason, it is crucial for students to share their insights on course-related topics.

While in class, please set your cell phone to vibrate. If you need to take a call, please step out of the room to do so.

Assignments

All assignments are to be submitted by midnight on the assignment's due date.

Late assignments will not be accepted. No exceptions.

Assignments may not be submitted via e-mail.

Written assignments must be word processed and spell- and grammar-checked.

Programming Assignments

All programming assignments must include a header in the form of a Python docstring (see https://www.python.org/dev/peps/pep-0257/) which includes your name, your name, the assignment number, a description of the assignment, the date submitted, and the names of anyone you collaborated with.

Programming assignments must also follow the coding conventions laid out in PEP-8 (https://pep8.org/). Code lines can reach 100 characters.

Labs

Labs are required to have the same file header and coding conventions as the regular programming assignments.

Labs will be held weekly on Friday.

Lab work is due no later than midnight of the Thursday following the lab session in which it was assigned.

Participation during lab class will constitute 30% of the possible points for each worksheet.

Quizzes

Quizzes will be given every week except for weeks 1, 9 & 10.

The lowest quiz score will be dropped.

Exams

There will be a cumulative exam given during the regularly scheduled final exam time during week 11 of the term. This exam may be replaced by a final project of an appropriate scope.

Missed Work

In case of absence from class, students are responsible for announcements made and materials covered. If an absence results in a missed quiz, that quiz will be treated as the low score which is discarded when total points are calculated at the end of the term.

Collaborative Work

As students, you are encouraged to discuss assignments and course materials with your peers. However, unless the assignment is a group project, the work you turn in to be graded must be yours and yours alone and you must be capable of explaining it verbally to the instructor if asked. If you received pointers or guidance from another student in the class you must clearly state that near the top of the assignment and list the names of everyone who helped.

Work that appears to be directly copied from another student will not be given credit.

Tutors

Tutors are generally available for this and other computer science classes. Check with the instructor and/or the Learning Center if you feel you need further assistance with this course.

Assessment

Grades will be determined according to the following schedule:

Type	Weight
Programming Assignments	25%

Labs	25%
Quizzes	20%
Final Exam/Project	30%

Score	Grade
90 – 100%	A
80 – 89%	B
70 –79%	C
60 –69%	D
< 60%	F

Academic Integrity and Honesty

As per LBCC Administrative Rule 7030-02 (http://bit.ly/LBCC-AR-7030-02):

> *Students at LBCC are responsible for pursuing their studies with honesty and fairness, and in a manner that respects the rights and dignity of others. Students must not engage in acts of dishonesty. Academic dishonesty includes, but is not limited to, such acts as forgery, changing or misuse of college documents and records of identification, cheating, plagiarism, aiding or abetting cheating or plagiarism, knowingly furnishing false information to the college, violating copyright or trademark, or copying college software.*

It would behoove you to familiarize yourself with this Administrative Rule as well as the Student's Rights, Responsibilities, and Conduct Code, LBCC Administrative Rule 7030-01 (http://bit.ly/LBCC-AR-7030-01), and the associated LBCC Board Policy 7030 (http://bit.ly/LBCC-BP-7030).

LBCC Center for Accessibility Resources

Students who may need accommodations due to documented disabilities, who have medical information which the instructor should know, or who need special arrangements in an emergency should speak with their instructor during the first week of class. If you believe you may need accommodations but are not yet registered with the Center for Accessibility Resources (CFAR), please visit the CFAR Website for steps on how to apply for services or call 541-917-4789.