**CS 110 Spring 2019**

**MWF 9 – 10, Th 9- 10:40 SAM 202**

**Introduction to Computer Science with Python**

|  |  |
| --- | --- |
| **Instructor:** Clarke Wellman, SAM 221 | **Office Hrs:** MTuWF 10-11 or by apt. |
| https://sites.google.com/site/csc110spr19/ | [**ctwellman@gmail.com**](mailto:ctwellman@gmail.com) |

**Course Description**

This is an introductory course in computer science using the Python language. Python is a great first language - we can focus on ideas & on problem solving with programming instead of getting bogged down in language syntax. We will cover Chapters 1-8, parts of 9,11,20 in **How to Think Like a Computer Scientist: Learning with Python 3**, by Peter Wentworth, Jeffrey Elkner, Allen B. Downey and Chris Meyers, which I will refer to as **HTTLACS**. *Read* assigned material *before* coming to class so we can focus on the process of problem solving and coding.

This is a fast-paced course! Keep up with the reading. Do homework & submit it on time or I will not accept it. Come to class for lectures & activities. If you need help, see me during office hours, let me know by email that you are having trouble or go to the Tutoring Center. Youtube Khan Academy style explanations for key concepts are posted on the website. Let me know if you are having trouble with material & I can post a youtube covering the topic. Go to <https://www.python.org/> & download Python 3.6 on your computer There is only one way to do well in this class: do lots of programming and learn how to problem solve with Python. Python makes this easier, but you still have to put in the work. In this regard, I like the sports analogy of practicing & doing the workouts to become skilled & competent.

**Text**

**How to Think Like a Computer Scientist: Learning with Python 3**. **HTTLACS** is on the website in the references page as a pdf. Download this & put it on a usb. Tests are open book and open computer but no internet is allowed. Do assigned reading before the material is discussed in class. I will cover the main ideas in lecture. Focus during class will be on programming & problem solving.

**Video Presentations for Homework, Extra Credit and Projects**

You will be required to submit an onscreen voice & video explaining your code for 1 problem per homework set (the “youtube” problem), for every extra credit assignment & for your project presentation. Use an onscreen recorder for the presentation – free ones can be had online. <http://www.screencast-o-matic.com/> works well for PC or MAC. A usb microphone for your computer may be desirable. I do want reasonable sound quality in your youtubes. Once you make your onscreen recording, post your video to youtube & send put the link to your youtube video in your code as described below. Please do not sent me video files – they are typically large files.

**Submitting Homework, Extra Credit & Projects**

Email me all of the above separately – do not combine homework, extra credit and project submissions.

For homework:

1. In the email subject, put:

csc110 hw\_xx john smith

where xx is the homework number, john smith is your full name. Use lowercase, just as shown above.

1. Attach each python file separately, using the following name convention:

hw\_xx\_yy.py xx is the homework number, yy is the exercise number

hw\_xx\_yy\_yt.py yt indicates this is the youtube assignment. Put your youtube link in your python code & comment it out.

For extra credit:

1. In the subject put:

csc110 xc\_xx john smith

where xx is the extra credit number. All extra credit assignments require a youtube link. Put your youtube link in your python code & comment it out.

1. Attach your python file to the email.

For the project:

1. In the subject put:

csc110 project john smith

1. Attach your python file(s) one at a time
2. Attach your Powerpoint file
3. Put a youtube link for presentation on the title page of the Powerpoint file.

**Homework**

(1) [50%] Programming problems, 1 set per chapter, from **HTTLACS**. Submit as discussed above. Note that each .py file is separate.

(2) [50%] Starting with hw2, a single problem per week for which you (a) submit your Python code as a separate file and (b) which you present in a short video made with an onscreen video posted to youtube. Put the link to your youtube in the python code & comment it out. If you cannot record good sound quality during your presentation, I recommend that you invest in a usb microphone for your computer. Your goal is to write clear and transparent code & your grade will be based on the code working & on clearly written and clearly explained code. Your youtube must contain appropriate sample runs. I want to hear you talking & explaining your code. Please do not submit youtubes where you simply type the code during the video. Send your homework to me via email as explained above.

Homework is generally due by Saturday midnight. Please submit homework on time. If homework is submitted late, I will grade it the following week, but you will get at most 50% credit.

**Worksheets**

There will be many in class programming problems given out as worksheets & collected at the end of class. Note that these worksheets are in the Lecture Notes. Please make sure you solve, handwrite code & return these, as they are part of your grade. Input should be in the form of handwritten code with your name and the worksheet name. Worksheets need to be completed and turned in within a week from the time they are done in class or they will be graded down.

**Quizzes & Tests**

There will be weekly quizzes starting in Week # 1. Most if not all quizzes will be a short in class quiz with a takehome due the first class day of the following week. Quizzes and the final will often be open computer, open book and open notes & will include take home quizzes. You can use your homework and worksheets during the quizzes. No midterms are planned, although this may change. Please plan on being present for the quizzes - unless you have a bona fide excuse, a missed quiz is dropped. Quizzes count for a big part of your grade. You get to drop your lowest quiz score. Quiz dates may change, in which case I will alert you by email.

**Project**

You are required to do an individual programming project. See the detailed writeup on the website, which describes a project using the turtle module to draw a pattern. It is your responsibility to read the posted writeup. Let me know if you have questions, but I will not answer questions until you have read the writeup. The project will be emailed to me as described above on or before the Wednesday before finals week by 11:45 pm. Early projects get extra credit (2% per day early, for a max of 10%), late projects get penalized (5% per day late, no projects accepted after final). Get started early on your project & get it in on time or ahead of time. The project will include your Python code, sample runs and a write up in Powerpoint with a description of your program, including inputs, outputs and a flow chart if you would like to use one. Put a youtube link of a video with you presenting your Powerpoint charts (voice + video) and demonstrating your program in your Powerpoint presentation. Project submission is discussed above. Please do not send any videos, just put the youtube link on the title page in your Powerpoint charts. The youtube is not instead of the writeup, it is in addition to the writeup.

**Grading**

Homework: 20% ( ½ for set, ½ for youtube)

Worksheets 10%

Project: 15%

Quizzes: 40% (drop low quiz)

Final 15%

**usb Drives & Python**

We will be using Python 3.6, which you should download on your own machines from http://python.org/download/releases/3.6/. Python 3.6 is available on the student network as well.

Don’t depend on keeping your programs by saving them on the student computer hard drive – your code will be deleted by system maintenance scripts. Save Python programs to your usb drive & bring the usb with your Python code to class every day. Do not leave your usb in the school computer! Set up a Python Program folder on your usb for all your programs.

**Sharing Code**

The best way to do the work for this class is to work individually. It’s also OK to discuss solutions with other students, but copying someone else’s code is not allowed. You are taking this class to learn how to use Python to solve problems & you will gain nothing if you do not actually write the code and think through the problems. Suppose you are taking an exercise class to prepare for a Himalayan trek. Would you hire someone to do the exercises for you? What will you do when you are on Annapurna at 15,000 ft? What will you do on the quizzes & on the final if you are not doing the work yourself? What will you do in future programming classes or when you are on the job & your employer expects you to have basic programming skills?

Students with documented disabilities who need course accommodations, have emergency medical information, or require special arrangements for building evacuation should contact the instructor within the first two weeks of class.

Title IX Responsibilities: Seattle Central College faculty are committed to supporting our students and upholding gender equity laws as outlined by Title IX. Therefore, if a student chooses to confide in a member of Seattle Central College’s faculty regarding an issue of sexual misconduct, that faculty member is obligated to tell Seattle Central College Title IX Coordinator. The Title IX coordinator will assist the student in connecting with all possible resources both on and off campus.