Syllabus

CSE 30: Programming Abstractions in Python

Dr. Larissa Munishkina

Department of Computer Science and Engineering

University of California, Santa Cruz

Course Information. CSE 30 class is the continuation of CSE20 class, Beginning Programming in Python, where you have been introduced to the Python programming language and learned how to use standard Python library modules such as random, math, regular expressions, datetime, turtle graphics, tkinter, or others.

In this class, you will continue to learn Python modules from standard and external libraries including NumPy, Matplotlib, Tkinter, OpenCV, SciPy, TensorFlow, PyGame, and others. You will be introduced to more advanced concepts in programming such as Abstract Data Types (ADT) and Data Structures (Stacks, Queues, Priority Queues, Trees, Graphs, and Hash Tables), programming approaches for Searching, Sorting, and Optimizing, as well as Recursions, Threading, Event-Driven Programming, Functional Programming, Linear Programming, and Dynamic Programming. We will cover some advanced topics in programming including AI, Machine Learning, Computational Complexity, and Satisfiability (SAT).

Please, check the class Syllabus page for more information on the class policies, grading, DRC services, and the instructor's, teaching assistants' and tutors' contacts.

Thank you for joining the class! I hope you will have a wonderful coding adventure with us! Good luck!

Important Information. Quizzes, labs, and programming assignments will be submitted on Canvas. If you encounter problems, let us know: post your question on Canvas, attend discussion sections, or contact the teaching assistants or instructor.

The class schedule can be found here: Class Schedule. All recorded lectures can be found in the Assignments folder under the name Video Lectures or YuJa. All other class material can be found in the Canvas Files folders.

Outcomes. CSE30 Programming Abstractions in Python is a class where you will continue learning the programming language Python and be introduced to new concepts in programming including Abstract Data Types (Stack, Queues, Trees, Graphs), Algorithm Analysis and Approaches (Greedy

Algorithms, Dynamic Programming), and Programming Paradigms (Functional and Event-Driven Programming).

Instructor:	Larissa (Lara) Munishkina (mlarissa@ucsc.edu)
Main Textbooks:	Problem Solving with Algorithms and Data Structures using Python — Problem Solving with Algorithms and Data Structures (runestone.academy) CSE 30 (Larissa Munishkina's Google Colab)
Optional Textbooks:	Programming Abstractions in Python | abstractions-in-python (lucadealfaro.github.io) Python Cookbook, 3rd Edition [Book] (oreilly.com) Introducing Python, 2nd Edition [Book] (oreilly.com) Fluent Python, 2nd Edition [Book] (oreilly.com) Learning the Unix Operating System, 5th Edition [Book] (oreilly.com)

Main Online Resources

Canvas webpage:	Programming Abstractions: Python (ucsc.edu)
SOE webpage:	Programming Abstractions: Python | Course Web Pages (ucsc.edu)
Additional Online Resources:	The Python Standard Library — Python 3.8.9 documentation Download Python | Python.org NumPy SciPy.org — SciPy.org TkDocs Tutorial Gallery — Matplotlib 3.4.1 documentation turtle — Turtle graphics — Python 3.8.9 documentation Pygame Front Page — pygame v2.0.1.dev1 documentation UNIX / Linux Tutorial for Beginners (surrey.ac.uk)
UCSC ITS and UNIX Timeshare:	UC Santa Cruz - Information Technology Services (ucsc.edu) Software for the Campus (ucsc.edu) UNIX/AFS Handouts (ucsc.edu)

Course Work. This course is a programming course. You will have five programming assignments, ten lab assignments, four quizzes, and the final exam. The final grade will be calculated according to the following schema:

Programming Assignments	20
Lab Assignments	30
Quizzes	30
Final	15
Class Participation	5

Participation includes participation in lectures, discussion sections, and Canvas posting.

Please note that you will receive points for attending lectures and discussion sections (three lectures and one discussion section a week). Discussion sections have been scheduled to accommodate students having different schedules.

If you cannot attend a lecture, you can watch a corresponding lecture video in the Assignment folder under the name Video Lectures or on YuJa. You have to submit the assignment to get credit for the lecture attendance (if you attend the lecture, you still need to submit the assignment, but you do not have to watch the video).

Assessments and Grades. You can calculate your own grade. All assignments have points, and you can obtain 1000 points in total (without extra credit). Letter grades are assigned according to the following schema:

Letter Grade	Points
A+	975
A	925
A-	900
B+	875
B	825

B-	800
C+	775
C	725
C-	700
D+	675
D	625
D-	600
F	0

Class Participation. Class participation includes lecture assignments, discussion section attendance, and participation in discussion forums on Canvas.

Late Policy. Please note that all assignments have a due date. Assignments can be submitted late (up to one week) with penalty. The late penalty is 5 percent per day for all late submissions. Missing assignments will receive zero points.

Labs. You will have ten lab assignments. The main purposes of taking lab assignments are to learn Python libraries and packages and to get prepared for quizzes and programming assignments. Labs will include the following topics: using standard library modules including random, math, datetime, re (regular expressions), sys, os, html, and internal and external libraries such as Turtle, Tkinter, Matplotlib, NumPy, SciPy, PyGame, and others. Lab assignments should be completed and submitted on Canvas.

Quizzes. There will be four quizzes. Quizzes will be administered every other week and should take around 40 minutes to complete. If you need special accommodation, please contact me and I can schedule a quiz at a different time.

Final Exam. There will be a final exam in the final week of the quarter. If you are ill or have an unexpected family emergency, you must notify the instructor as soon as possible and before the examination day. Otherwise, you may have a zero grade on your exam.

Programming Assignments. All four programming assignments must be done individually and have to be submitted on time on Canvas. All assignments will be graded according to the grading rubrics published under each assignment folder.

You can consult the instructor, teaching assistants, tutors, or students about general approaches of the program, its syntax, and debugging the code. However, do not copy other's code or share your own code with other students or post the code on the Internet. If the same or very similar programming codes are found between students, the students will receive a lower or a zero score for the assignment. Administrative action may also be taken with the university. Please read about the Academic Integrity Policy written below.

Academic Integrity Policy. Academic integrity is the cornerstone of a university education. Academic dishonesty diminishes the university as an institution and all members of the university community. It tarnishes the value of a UCSC degree. All members of the UCSC community have an explicit responsibility to foster an environment of trust, honesty, fairness, respect, and responsibility. All members of the university community are expected to present as their original work only that which is truly their own. All members of the community are expected to report observed instances of cheating, plagiarism, and other forms of academic dishonesty to ensure that the integrity of scholarship is valued and preserved at UCSC.

In the event a student is found in violation of the UCSC Academic Integrity policy, he or she may face both academic sanctions imposed by the instructor of record and disciplinary sanctions imposed either by the provost of his or her college or the Academic Tribunal convened to hear the case. Violations of the Academic Integrity policy can result in dismissal from the university and a permanent notation on a student's transcript.

For the full policy and disciplinary procedures on academic dishonesty, students and instructors should refer to the Academic Integrity page at the Division of Undergraduate Education.

Accommodation and External Help. UC Santa Cruz is committed to creating an academic environment that supports its diverse student body. If you are a student with a disability who requires accommodations to achieve equal access in this course, please submit your Accommodation Authorization Letter from the Disability Resource Center (DRC) to me privately by email or during my office hours within the first two weeks of the quarter. At this time, I would also like us to discuss ways we can ensure your full participation in the course. I encourage all students who may benefit from learning more about DRC services to contact DRC by phone at 831-459-2089, or by email at drc@ucsc.edu.

Small Group Tutoring. Small Group Tutoring (SGT) supports students academically to advance educational equity by designing inclusive learning environments outside of the classroom. SGT is open to all students enrolled in the class and they must sign up on an online system: TutorTrac.

You can sign up using this link: https://ucsc.go-redrock.com/tracweb40. You can also find the link on the website: https://lss.ucsc.edu/index.html.

Title IX. The university cherishes the free and open exchange of ideas and enlargement of knowledge. To maintain this freedom and openness requires objectivity, mutual trust, and confidence; it requires the absence of coercion, intimidation, or exploitation. The principal responsibility for maintaining these conditions must rest upon those members of the university community who exercise most authority and leadership: faculty, managers, and supervisors. The university has therefore instituted a number of measures designed to protect its community from sex discrimination, sexual harassment, sexual violence, and other related prohibited conduct. Information about the Title IX OfficeLinks to an external site., the online reporting link , applicable campus resources, reporting responsibilities, the UC Policy on Sexual Violence and Sexual Harassment , and the UC Santa Cruz Procedures for Reporting and Responding to Reports of Sexual Violence and Sexual Harassment can be found at titleix.ucsc.edu. The Title IX Office is actively responding to reports and requests for consultation. If you are not currently working with someone in the office and want to make a report/request a consult, you can expect the fastest response by using our online reporting link. For more information, please visit the Title IX Operations under Covid-19 page.

CSE 30 Class Schedule

	Date		Lecture	Reading	Assignment
1	M	Week 1	Intro to class, Syllabus	Canvas webpage	LAB 1 posted
2	W	Week 1	Programming Abstractions, Algorithm Analysis	Ch.1.1-1.7, 3	**PA 1 posted**
3	F	Week 1	Data Types (part 1): sequences, bytes, arrays, data collections, OOP	Ch. 1.7 – 1.17, 2	
4	M	Week 2	Data Types (part 2): generators and iterators	Lecture slides	LAB 1 due / LAB 2 posted
5	W	Week 2	Debugging and Testing: Modular Programming, Import and Assert Statements, UnitTest	Lecture slides	
6	F	Week 2	Overview	Lecture slides	**Quiz 1**
7	M	Week 3	Functional Programming	Lecture slides	LAB 2 due/ LAB 3 posted
8	W	Week 3	Recursions (part 1)	Ch. 5	**PA 2 posted**

9	F	Week 3	Recursions (part 2)	Ch. 5	
10	M	Week 4	Event-Driven Programming: Turtle Graphics	Lecture slides	LAB 3 due / LAB 4 posted
11	W	Week 4	Data Analysis and Visualization: NumPy and Matplotlib	Matplotlib and NumPy webpages Lecture slides	
12	F	Week 4	Overview		**Quiz 2**
13	M	Week 5	Event-Driven Programming: Tkinter and PyGame	Lecture slides	LAB 4 due / LAB 5 posted
14	W	Week 5	Abstract Data Types: Stack, Queue, Deque	Ch. 4	**PA 3 posted**
15	F	Week 5	Abstract Data Types: Heap, Priority Queue	Ch. 7	
16	M	Week 6	Abstract Data Types: Tree, BST	Ch. 7	LAB 5 due / LAB 6 posted
17	W	Week 6	Postfix, Infix, Prefix Notations, Expression Trees, Tree Traversals	Ch. 7 Lecture slides	
18	F	Week 6	Overview	Lecture slides	**Quiz 3**
19	M	Week 7	Graphs (part 1)	Ch. 8	LAB 6 due/ LAB 7 posted
20	W	Week 7	Graphs (part 2)	Ch. 8	**PA 4 posted**
21	F	Week 7	Sorting Algorithms (part 1)	Ch. 6.6 – 6.12	
22	M	Week 8	Sorting Algorithms (part 2)	Ch. 6.6 – 6.12	LAB 7 due/ LAB 8 posted
23	W	Week 8	Search Algorithms (part 1)	Ch. 6.1 – 6.5, 8	
24	F	Week 8	Overview	Lecture slides	**Quiz 4**
25	M	Week 9	Search Algorithms (part 2)	Ch. 6.1 – 6.5, 8	LAB 8 due/ LAB 9 posted
26	W	Week 9	Greedy Algorithms, Dynamic Programming, Games	Lecture slides	**PA 5 posted**
27	F	Week 9	Optimization Theory, Linear Programming, SciPy	Lecture slides	
28	M	Week 10	Advanced Topics: AI, ML, NLP	Lecture slides	LAB 9 due/ LAB 10 posted
29	W	Week 10	Advanced Topics: Threading, OS	Lecture slides	

30	**F**	Week 10	FINAL DATE OF INSTRUCTIONS Final Overview	Lecture slides	**EVERYTHING DUE** LAB 10 due, **PA 5 due**
31		**Final Week**	**FINAL EXAM**	**TBA**	**FINAL EXAM DUE**