CS 151: Introduction to programming in Python

Willamette University, Spring 2025

Instructor:

Fred Agbo, PhD
fjagbo@willamette.edu
Office: Ford Hall 209
Office Hour: T, TH (11:00 – 1:00 pm)

Web: https://willamette.instructure.com/courses/7417
Lecture: M/W/F (10:20am–11:20am/12:00pm-1:00pm)
Lecture Hall: Ford Hall 301
Course Discord Server: https://discord.gg/UyGqNqbU44

(This syllabus is subject to change as the semester progresses - particularly the schedule. I will inform when an update is made.)

Brief Course Description

This course is an introduction to computer science using Python. Introduces students to the fundamental concepts of programming and computational problem solving. Students will study and create programs that perform various tasks, including text and file manipulation, data structures, and testing. Topics will include general programming idioms such as variables, logic and loops as well as Python specific idioms such as list comprehension and generators. Object-oriented programming will be introduced.

Prerequisite(s): There is no prerequisite for enrolling in this course. As a matter of fact, the course is open to all students.

Resources for this course:

Mainly, we will be using the book entitled "Programming in Python" authored by Professor Eric S. Roberts throughout the semester to supplement what we talk about in class and other online resources that I may provide as the course progresses. The book is available for free via PDF on the course website here.

Course Objectives:

Over the semester, students will gain working knowledge in:

1. The fundamentals of coding: variables, logic, and loops.
2. Implementing fundamentals in Python to create basic programs.
3. Applying decomposition and stepwise refinement to a problem-solving strategy.
4. Designing data structures to hold and model information.
5. Debugging and testing programs to ensure they are working as intended.

The field of computer science and programming is vast and the entirety of what is possible in Python could never be contained in a single course. This course seeks to provide solid fundamentals and confidence to students so that they might continue their learning on their own or in whatever direction their creativity might take them.

Tentative Course Outline:

Week/Topic	Date	Chapter	Description	Due
1	Mon. Jan 13	Ch0	Kickoff lecture	
Karel the Robots	Wed. Jan 15	Ch0	Karel the Robot	
	Fri. Jan 17	Ch0	Karel the Robot	
2	Mon. Jan 20		*Martin Luther King Jr. Day*	
Intro to Python	Wed. Jan 22	Ch 1	Introducing Python	
	Fri. Jan 24	Ch 1	Introducing Python	S1: Karel the Robot
3	Mon. Jan 27	Ch 2	Control Statements	Problem Set 1
Control Statements	Wed. Jan 29	Ch 2	Control Statements	S2: Simple Python and Control Statements
	Fri. Jan 31	Ch 3	Algorithms	
4	Mon. Feb 3	Ch 7	Strings	Problem Set 2
Strings	Wed. Feb 5	Ch 7	Strings	S3: Strings
	Fri. Feb 7	Ch 7	Strings	
5	Mon. Feb 10	Ch 4	Wordle Overview & Simple Graphics	Problem Set 3
Graphics	Wed. Feb 12	Ch 4	Simple Graphics	S4: Graphics & Wordle
	Fri. Feb 14	Ch 5	Functions	
6	Mon. Feb 17	Ch 6	Writing Interactive Programs	Project 1: Wordle
Interactive Programs	Wed. Feb 19	Ch 6	Writing Interactive Programs	S5: Midterm 1 Prep
	Fri. Feb 21		**Midterm 1 exam (Ch 1 – 5 & 7)**	
7	Mon. Feb 24	Ch 6	Writing Interactive Programs	
Interactive Programs	Wed. Feb 26	Ch 6	Writing Interactive Programs	S6: Interactive Graphics
	Fri. Mar 28	Ch 6	Writing Interactive Programs	
8	Mon. Mar 3	Ch 8	Breakout Overview & Lists	Problem Set 4
Lists	Wed. Mar 5	Ch 8	Lists	S7: Breakout
	Fri. Mar 7	Ch 8	Lists	
9	Mon. Mar 10	Ch 8	Lists	Project 2: Breakout
Lists and Debugging	Wed. Mar 12	Ch 8	Debugging	S8: Lists
	Fri. Mar 14	Ch 10	Classes and Objects	
10	Mon. Mar 17	Ch 10	Classes and Objects	Problem Sets 5
Classes and Objects	Wed. Mar 19	Ch 10	Classes and Objects	S9: Midterm 2 Prep
	Fri. Mar 21		**Midterm 2 exam (Ch 6 & 8)**	
11	Mon. Mar 24			
Spring Break	Wed. Mar 26		*Spring Break*	
	Fri. Mar 28			
12	Mon. Mar 31	Ch 11	Imageshop Overview	
Dictionaries and Sets	Wed. Apr 2	Ch 11	Dictionaries and Sets	S10: Image Manipulation
	Fri. Apr 4	Ch 11	Dictionaries and Sets	
13	Mon. Apr 7		Personal Project Overview	Project 3: Imageshop
Data Structure	Wed. Apr 9	Ch 12	Data Structure	S11: Data Structure
	Fri. Apr 11	Ch 12	Data Structure	
14	Mon. Apr 14	Ch 12	Overview of Adventure (Prof. Eric)??	Project 4: Personal Project
Algorithmic analysis	Wed. Apr 16	SSRD	*No Class*	S12: Adventure
	Fri. Apr 18	Ch 12	Data Structure	
	Mon. Apr 21	Ch 12	Data Structure	
15	Wed. Apr 23	Ch 9	Algorithmic Analysis	S13: Adventure
	Fri. Apr 25	Ch 9	Algorithmic Analysis	
16	Mon. Apr 28		Last Day of Class: Reviews	Project 5: Adventure S14: Final exam prep

* SSRD - *Student Scholarship Recognition Day*

** Homeworks are due on a Monday following the week it was published at 10:00 PM*

10:20 – 11:20 M/W/F -> Friday 2nd May (8:00pm - 11pm)
12:00 – 1:00 M/W/F -> Wednesday 7th May (2:00pm - 5pm)

Potential holidays/No class	🟧
Guest lecture	▨

Student Learning Objectives (SLO):

This course has been designed to ensure that at its completion, students should be able to:

- Describe, design, implement and test structured programs to solve a problem. Problem-solving is at the heart of programming, and students must be able to a take a given problem, break it up into solvable steps, and implement each individual piece to achieve their solution.
- Explain what an algorithm is and how it relates to computer programming. Many types of problems share similar solutions. Knowing when they can be used, how to use them, and the benefits of one method over another is important.
- Recognize and construct common programming concepts, including variables, loops, functions, I/O, and logic. The bread-and-butter of basic programming. These are the tools in a student's toolbox from which students can pull to construct all their programs. Knowing them well leads to both increases in efficiency and creativity in how they can be used.
- Recognize the difference between various data structures like lists, dictionaries and tuples and decide the proper times to use each. Python has a variety of ways in which to store data. Choosing the correct one for a problem can eliminate many issues further down the line.

Feedback and Course Grading

I will ensure that feedback is provided on each homework, projects, mid-term exams, and final exam. Aside from the feedback, certain percentage of the grade will be awarded to students for active participation in the classroom, lab session, and via other communication channels created for this course. The weighting of the grades include:

- Participation 10%
- 5 Homework (Problem sets) 20%
- 4 Defined Projects 20%
- 1 Personal Project 10%
- 2 Mid-term exams 20%
- A Final Exam 20%

Grade distribution:

>= 92.00	A	72.00 - 77.99	C
90.00 - 91.99	A-	70.00 - 71.99	C-
88.00 - 89.99	B+	68.00 - 69.99	D+
82.00 - 87.99	B	62.00 - 67.99	D
80.00 - 81.99	B-	60.00 - 61.99	D-
78.00 - 79.99	C+	<= 59.99	F

Course Structure and Assessment

Participation

- Participation in lectures will be scored. Questions will be asked in class and students will respond via polling technology. Simply responding to each question will earn *all* your participation points for the day, but answering correctly will earn you some extra credit. If you have an excused absence and cannot be in class, you can earn the extra credit participation points by emailing me asking a question about a specific slide. Over the course of the semester, that extra credit can really add up, so I recommend showing up and taking the polling questions somewhat seriously!
- Another way students can earn participation points is by collaborating in class through an online tool Pensieve AI, which I'm going to try out for the first time.

Roster attendance will be taken at the end of the class, in which the total record will be combined with the small sessions attendance record to determine average attendance. Note! You must comply with the attendance policy of SCIS (see here)

-

Small Sections

- To better facilitate students getting assistance, guidance, and role-models, we are continuing our small section model this semester. Students will be split into small groups of 7-10 and assigned to a section leader. All section leaders are previous students who excelled in the course and are excited to help you do the same! Sections will meet for 1 hour with their section leader each week, wherein they will work through a few short problems and demonstrations. **Section leaders will also serve as experts and peers that students can contact to ask questions and get extra help and guidance on problem sets and projects**. Section attendance is graded as part of your course participation, and you are expected to show up and participate each week. Sections will meet on Wednesdays or Thursdays, and a poll concerning your availability on those days will go out on the first day of classes. Aside from getting all your help from the section meeting with the student leaders, you can also receive help on your problem sets and projects from QUAD center (see this link for their opening hours https://willamette.edu/offices/quad/hours/index.html). As a policy, my professor colleagues and I will **only** attend to students with questions on **project and or problem sets** if and only if that student attends small sections meeting regularly and had exhausted all options of asking the section leaders and QUAD center without finding an answer to the question.

Homework

- There will be problem sets which will be due at 10:00 pm on Mondays of the week. Ensure you submit as is at the due date/time. Each problem sets will comprise of 2-3 questions which require writing a program to accomplish a particular task. In some cases, template files are provided, and so, students have to simply add the necessary code to the existing file. Problem set templates and submissions will be handled through GitHub Classroom. We will also explore a new tool to solve the assignment if necessary. We will discuss and demonstrate these tools and how to use them in class. Assignments will be posted each week and the provided link should be followed to download that week's assignment materials. If you work with anyone on the homework or use any online tools such as generative artificial intelligence, please provide their name or the name of the AI tool at the top of the file as a comment in your code.

- General Info: Programming is very hands-on, and the odds are high that you will not do well in the course if you do not practice! As far as I know, the best method to gain proficiency is solving problems and writing code. There are no real "shortcuts". The number and length of problems assigned is my best estimate for having you adequately practice and learn the material without being an excessive burden on your time. Working in groups and helping others is very encouraged, though students must turn in their own work. I highly recommend helping and instructing other classmates if you feel proficient on a topic, both to help them and because there is no better way to identify gaps in your own knowledge than when you attempt to teach something. *But clearly indicate who you worked with at the top of each problem!*

Projects

- There are 5 larger projects scattered throughout the semester. Unlike the problem sets, which tend to be more narrowly focused, the projects are a chance to pull from all the areas of computer science and programming that we have been discussing to form a cohesive and functional program. The first 4 projects are well defined in terms of milestones and deliverables while the 5th (personal) project is loosely defined such that students have the liberty to determine its aim, scope, and functionalities but must be reasonably advanced to reflect use of problem-solving skill. Because of the broader scope of all projects, they will be scored on a more holistic scale:

Score	**Corresponding values**	**Description**
++	105	An absolutely fantastic submission that goes far above and beyond the set requirements. Comes along maybe only a few times a semester, if at all.
+	100	A submission that exceeds expectations. The program must reflect additional work beyond the requirements or get the job done in a particularly elegant way.
✓+	95	A submission that satisfies all the requirements for the assignment. A job well done!
✓	90	A submission that meets the requirements of the assignment, but which is either stylistically weak or has a few minor problems.
✓−	80	A submission that falls significantly short of the requirements of the assignment.
−	60	A submission that shows even more serious problems but nonetheless shows some effort and understanding.
−−	30	A submission showing little effort and not representing passing work.

Personal Project

- There will be ONE personal project that will be similar in format but distinct in content from other defined projects or homework. Whereas homework will include programming assignments for which there is some correct answer that may be measured against an answer key, the personal project will be a student-centered exploration of computation meant to give you an opportunity to apply what you've learned in the course while still receiving support from an instructor. Thus, a strong final project can allow the student to earn a high course grade, regardless of prior scores on midterms and homework.

Tests/Exam

- There will be two (2) midterms and one (1) final exam this semester: All exams will be conducted directly on Canvas, meaning that you will be allow the use of computers and internet ONLY to some extent, for example, accessing the Canvas platform and NOT for browsing codes from the internet or compiling codes in any editor such as the Visual Studio Code. This policy is a frequent cause of confusion for students, but it has been shown to actually improve student test scores, as students can focus on the larger picture of their code (which is what they are really being scored on) rather than getting hung up on one small error that they can't fix. There will be study guides and examples of old exams available before each exam so students can feel prepared for the types of questions that may appear.

Course Policies:

General things to note:

- The course assignment and project platforms are (i) GitHub classroom, (ii) Pensieve or CodeGrade? and (iii) Canvas
- Collaboration is highly encouraged in this course both during and outside of the classroom. Students may work together to understand how to solve the homework and prepare for the exams, and other class activities.
- Learning programming most of the times is by copying, integrating, and debugging codes.
    - Use these strategies in a very helpful manner
    - Be honest and transparent
    - trust the instructor not to mark you down when you disclose your helpers (course mate, online tools, GenAIs, etc)
- The biggest learning outcomes in this course is to be able to program in Python. This means any helpers you use to circumvent this singular; big learning outcomes would mean a disservice to yourself.
    - Follow the guidelines for using helpers in this class and disclose your helpers in all submissions
- Students are responsible for all missed work, regardless of the reason for absence. It is also the absentee's responsibility to get all missing notes or materials.

Policy on the Use of AI and Generative AI in Intro to Programming with Python Class

I recognize the transformative potential of Artificial Intelligence (AI) and Generative AI (GenAI) in intro programming course. As such, I encourage students to explore and utilize these technologies ONLY as supplementary learning tools. AI can be an excellent resource for:

- Explaining complex concepts
- Providing coding examples
- Offering debugging suggestions
- Recommending best practices
- Exploring new ideas and approaches

Responsible Use of AI and Academic Integrity

While I'm not directly against the use of AI for learning, it is crucial to maintain academic integrity and ensure genuine skill development. Therefore:

- Do not use AI to directly solve assignments: Assignments are designed to develop your skills and understanding. Using AI to generate solutions for assignments is strictly prohibited.
- Acknowledge AI usage: When you use AI or GenAI tools for inspiration or assistance in your work, you must clearly acknowledge this in your submissions. Include a brief note explaining how and where AI was used.
- Understand the code you submit: Ensure you fully comprehend any code or concepts you include in your work, regardless of whether AI assisted in generating them.

Warning: The Perils of Over-Reliance on AI

I strongly caution against over-reliance on AI for completing coursework. Doing so will:

- Hinder your learning: The hands-on experience of writing code, debugging, and problem-solving is irreplaceable in developing your skills.
- Impede your growth: Relying on AI to solve problems will prevent you from developing the critical thinking and problem-solving skills essential for a successful career in web development.
- Limit your creativity: Over-dependence on AI can stifle your ability to come up with innovative solutions and unique approaches to problems.
- Compromise your future career: Employers value developers who can think independently, solve complex problems, and write efficient code. These skills can only be developed through practice and personal effort.

Consequences of Misuse of AI

Misuse of AI tools, including using them to directly complete assignments without proper acknowledgment, will be treated as academic dishonesty and may result in disciplinary action.

Late Submission and Incomplete Assignment Policy

Late submission is not allowed in this course!

- I totally understand that as humans, things can sometimes come up or go wrong and you are unable to get an assignment turned in on time. This kind of situation calls for some flexibility where I could consider accepting a late submission. However, this flexibility MUST be subject to my awareness and approval. Therefore, if any student is in this kind of unfortunate situation and would need more time to submit homework or the project a bit late, please, contact me immediately. I must receive an email and reply to it before the due date in order to implement this policy. As a matter of rule, no lateness beyond 1 day (24 hours) can be tolerated for any given homework/project. In some cases, a penalty (i.e., losing a portion or an entire grade for a particular submission) could be applied by the instructor based on the assessment of reasons for late submission, which is solely at the instructor's discretion.

- All students are allowed maximum of three (3) late submission in all assignments (problem sets and projects) throughout the course, provided the late submission has been granted by the instructor following the protocol outlined above.

- Similarly, there are three (3) unexcused missed section meetings allowable throughout the semester. Violation of this rule will result to 5% grade deduction from the student's cumulative grade. If a student must miss any of the section meetings, they would have to inform the section leader earlier enough before the section meeting starts and must provide a concrete reason for missing the section. The section leader has the prerogative to grant or deny the request to be excused.

Willamette Policies:

This section has been largely developed/adapted from the Willamette University Academic Policy which can be accessed via this link https://willamette.edu/arts-sciences/catalog/policies/. As a result, the employer's view constitutes the information represented here except for minor edits made by the instructor to adapt the context to the specific case study of this class. Regarding the University's policies on course registration, add, drop, and withdrawal, see this link: https://portal.willamette.edu/offices/policies/Pages/Add-Drop-Withdrawal-Policy.aspx. I would appreciate it if students could reach out to me on any issues that have not been addressed regarding policies that guide this class.

Academic Honesty

Cheating is defined as any form of intellectual dishonesty or misrepresentation of one's knowledge. Plagiarism, a form of cheating, consists of intentionally or unintentionally representing someone else's work as one's own. Integrity is of prime importance in a college setting, and thus cheating, plagiarism, theft, or assisting another to perform any of the previously listed acts is strictly prohibited. An instructor may impose penalties for plagiarism or cheating ranging from a grade reduction on an assignment or exam to failing the course. An instructor can also involve the Office of the Dean of the School of Computing and Information Sciences for further action. For further information, read the School of Computing and Information Science students handbook. Additional information, visit: https://willamette.edu/arts-sciences/catalog/policies/plagiarism-cheating.php

Academic Integrity

Students of Willamette University are members of a community that values excellence and integrity in every aspect of life. As such, we expect all community members to live up to the highest standards of personal, ethical, and moral conduct. Students are expected not to engage in any type of academic or intellectually dishonest practice and are encouraged to display honesty, trust, fairness, respect, and responsibility in all they do. Plagiarism and cheating are especially offensive to the integrity of courses in which they occur and against the College community as a whole. These acts involve intellectual dishonesty, deception, and fraud, which inhibit the honest exchange of ideas. Plagiarism and cheating may be grounds for failure in the course and/or dismissal from the College. http://willamette.edu/cla/catalog/policies/plagiarism-cheating.php

Classroom Conduct

As an educational institution, the School of Computing and Information Sciences is committed to supporting the ideals and standards that help create a constructive and healthy learning community. That requires, among other things, encouraging positive classroom behaviors, discouraging disruptive classroom behaviors, and setting clear standards for both of those things.

Constructive classroom behaviors are those that support learners and teachers in an environment that promotes trust, respect, and collaborative learning.

Disruptive classroom behaviors are those that undermine or interfere with the abilities to learn and to teach. Clear examples of disruptive behaviors include but are not limited to: interrupting others or persistently speaking out of turn; distracting the class from the subject matter or discussion at hand; making unauthorized recordings or photos of a class meeting or discussion (except as permitted as part of an Accessible Education Services-mandated accommodations); and in extreme cases, any physical threat, physical, psychological, sexual harassment, ridicule, or abusive act towards a student, staff member, or instructor in a classroom or related setting.

Commitment to Positive Sexual Ethics

Willamette is a community committed to fostering safe, productive learning environments, and we value ethical sexual behaviors and standards. Title IX and our school policy prohibit discrimination on the basis of sex, which regards sexual misconduct — including discrimination, harassment, domestic and dating violence, sexual assault, and stalking. We understand that sexual violence can undermine students' academic success, and we encourage affected students to talk to someone about their experiences and get the support they need. Please be aware that as a mandatory reporter, I am required to report any instances you disclose to Willamette's Title IX Coordinator. If you would rather share information with a confidential employee who does not have this responsibility, please contact our confidential advocate at confidential-advocate@willamette.edu. Confidential support also can be found with SARAs at the GRAC (503-851-4245); and at WUTalk - a 24-hour telephone crisis counseling support line (503-375-5353). If you are in immediate danger, please call campus safety at 503-370-6911.

Diversity and Disability Statement

Willamette University values diversity and inclusion; we are committed to a climate of mutual respect and full participation. My goal is to create a learning environment that is usable, equitable, inclusive, and welcoming. If there are aspects of the instruction or design of this course that result in barriers to your inclusion or accurate assessment or achievement, please notify me as soon as possible. Students with disabilities are also encouraged to contact the Accessible Education Services office in Smullin 155 at 503-370-6737 or Accessible-info@willamette.edu to discuss a range of options for removing barriers in the course, including accommodations.

SOAR Center Offerings: Food, Clothing, and School Materials

The Students Organizing for Access to Resources (SOAR) Center provides free, confidential, and equitable access to food, toiletries, professional clothing, textbooks, and scholarly resources for all WU and WU-affiliated students. The SOAR Center is located on the Putnam University Center’s third floor. The space houses the Bearcat Pantry, Clothing Share, and First-Generation Book Drive and is maintained by committed students and advisors. Please check www.willamette.edu/go/soar for current hours of operation and email soar-center@willamette.edu for any questions or concerns.

Land Acknowledgement

We are gathered on the land of the Kalapuya, who today is represented by the Confederated Tribes of the Grand Ronde and the Confederated Tribes of the Siletz Indians, whose relationship with this land continues to this day. We offer gratitude for the land itself, for those who have stewarded it for generations, and for the opportunity to study, learn, work, and be in community on this land. We acknowledge that our University’s history, like many others, is fundamentally tied to the first colonial developments in the Willamette Valley. Finally, we respectfully acknowledge and honor past, present, and future Indigenous students of Willamette.

Time Commitment:

Willamette's Credit Hour Policy holds that for every hour of class time, there is an expectation of 2-3 hours of work outside of class. Thus, for this class, you should anticipate spending 6-9 hours outside of class engaged in course-related activities. Examples include reading course materials, preparing for discussion, and doing assignments and exams.

Intellectual Property & Privacy

Class materials and discussions including recorded lectures are for the sole purpose of educating the students enrolled in the course. The release of such information (including but not limited to directly sharing, screen capturing, or recording content) is strictly prohibited unless the instructor states otherwise. Doing so without the permission of the instructor will be considered an Honor Code violation and may also be a violation of other state and federal laws, such as the Copyright Act.