

INLS 560: Programming for Information Professionals Spring 2020

Basic Information

Date and time: Tuesdays and Thursdays, 12:30 to 1:45 p.m.

Location: Manning 14

Instructor Information

Instructor: Sayamindu Dasgupta

E-mail: sayamindu@unc.edu

Office: Manning 22

Office hours: Tuesdays from 3 p.m. to 5 p.m. (or by appointment)

Course overview

This course provides an introduction to computer programming focusing on language fundamentals and programming techniques for library and information science applications. Course content emphasizes problem-solving through the development of practical applications.

This course is taught with a focus on the Python programming language. However, the core concepts are relevant to most modern programming languages such as Java, JavaScript and C++.

Please note that this course is designed for students with no prior programming experience. If you already know how to program, this is not the course for you.

Learning objectives

At the end of this course, you will:

- Be familiar with fundamental concepts and techniques that are used in library and information science programming contexts.
- Be able to apply those concepts using the Python programming language.
- Be prepared to succeed in further SILS coursework.

Grading

You will be assessed based on the following elements:

- 7 take home assignments: 30 points each (total: 210 points)
- Final exam: 100 points
- Class participation: 90 points¹

There is a total of 400 points.

¹ Class participation also includes an ungraded, take-home midterm exam. Completing the exam is worth 10% of your class participation grade, regardless of how accurately you answer the questions.

Final grades will be assigned according to the following schedule:

A	380 to 400
A-	360 to 379
B+	348 to 359
B	336 to 347
B-	320 to 335
C+	308 to 319
C	296 to 307
C-	280 to 295
D+	268 to 279
D	240 to 267
F	<240

Assessment Details

Take-home assignments

There will be **seven take-home assignments**. Each assignment will be distributed on Sakai on the date indicated below.

<i>Assignment</i>	<i>Date distributed</i>	<i>Date due</i>
Assignment #1	January 9	January 16
Assignment #2	January 21	January 28
Assignment #3	January 28	February 4
Assignment #4	February 4	February 20
Assignment #5	February 20	March 5
Assignment #6	March 19	April 2
Assignment #7	April 2	April 21

You should create a new PyCharm project for each assignment. This can be done by clicking the "*Create New Project*" button on the window that appears when first launching the PyCharm application. Your project must be named according to the following convention: assignment<n>_<onyen>, where <onyen> is replaced with your onyen and <n> is replaced with the assignment number. For example, my onyen is "sdg1". Therefore, I would name my project for Assignment 3 as follows: assignment3_sdg1.

For each programming assignment, your submission will consist of a zip folder containing your source code for your solution. You are to submit your zip file via the Assignments section of Sakai. I will share a document in the Sakai resource section that shows how to create the zip file.

Late work

Late exams are penalized 5 points for each day that the assignment is late. A day begins when the assignment is due (that is, at the beginning of class) and continues until 24 hours have passed. Extensions will be granted in exceptional cases only.

Final exam

The final exam for this class will start at 12:00 noon on Friday, 1 May, 2020.

Participation

Participation will be graded according to these criteria:

- *Midterm*
There will be one ungraded, take home midterm exam that will be distributed on March 5 and due on March 10. Completing the exam is worth 10% of your class participation grade, regardless of how accurately you answer the questions.
- *Attendance*
It is important for you to attend class. Please be seated and ready when class begins. If personal difficulties (serious illness, etc.) make attendance problematic, please consult with me so that we can make an appropriate plan.
- *Department*
You should be attentive in class and respectful of your classmates and the instructor. Turn off cell phones and other devices that might disrupt class. Use laptops and other devices to support current course activities only.
- *Engagement*
Engagement includes: reading the assigned materials before class; asking questions when you do not understand the readings; making observations about the readings, being able to summarize their main points, and being able to respond to questions about the readings; participating in class activities; responding to discussion questions or other questions that I might ask during a lecture; actively listening and taking notes. I welcome productive disagreement (especially with me!), as long as it is expressed constructively and courteously. I value all informed opinions and encourage you to share them.

Engagement will be weighted more heavily than attendance and department.

Course technology

Sakai

Sakai will be used for assignments, midterms, and resources. A copy of this syllabus, as well as the textbook will be made available in the resources section of Sakai.

Python and PyCharm

The officially supported IDE for this course will be PyCharm Community Edition 5, from JetBrains software. It is available for free from this website: <https://www.jetbrains.com/pycharm/download/>.

Before using PyCharm, however, you'll need to have the Python interpreter installed. This can be downloaded for free from the official Python website: <https://www.python.org/downloads/>. Complicating things slightly is the fact that there are two major branches of Python language: version 3.x and version 2.x. For our class, we'll use the Python 3.x branch. Therefore, please download the latest version of Python 3.x and install it on your computer.

PyCharm and Python should both be installed on the computers in the SILS computer lab. In addition, PyCharm should be available via the ITS Virtual Lab under the SILS collection of software. However, it is *strongly recommended* that you install both Python and PyCharm on your personal laptop. This will allow you to actively participate in class exercises (a key component of your learning experience, not to mention your class participation grade). It will also allow you to complete your homework assignments without being tied to the computer lab and/or internet.

Piazza

A class discussion forum has been created at Piazza. Use this forum as the first place to go when seeking help. In this way, questions and answers will be visible to the entire class, allowing us all to learn from each other. Moreover, you'll often get a faster answer via Piazza because your classmates can help provide answers via Piazza as well. In fact, I encourage all of you to actively participate in the Piazza forum. To emphasize this, you may find that I respond to your email by asking you to "ask on Piazza" before I provide an answer. Our forum is available via the following link:

<https://piazza.com/unc/spring2020/inls560001> Please sign up for the Piazza forum through this link.

Semester Calendar

Note: This is a tentative schedule and is subject to change. Any changes will be announced in class and by email.

Most readings are from the *Python for Everybody* textbook, which you can download from <https://www.py4e.com/book> and from the resources section in Sakai. In the calendar and elsewhere in this syllabus, the book may be referred to as P4E. We'll start with a high-level discussion of how computers work, then begin digging into a range of programming concepts. Most classes will combine lecture-style instruction with hands-on class exercises when you'll "learn by doing". **Therefore, be sure to bring your laptop to class each day.**

For each day of the course, read the listed materials before class. In addition, try to complete the exercises that are indicated (if any). While ungraded, this preparation is essential for your success in this course. It will provide you with a foundation that we will build upon in class through both lectures and hands-on exercises, both of which are designed with the assumption that you will have completed the assigned preparation tasks. In addition, it will help you gain the knowledge and skills you need to complete your programming assignments. Finally, the material covered in these pre-class assignments will be potential topics for exam questions.

Date	Topic	To read before class	Exercise
Thursday, January 09	Introductions		
Tuesday, January 14	Fundamental ideas	Read chapter 1 of P4E and watch video at https://www.youtube.com/watch?v=p3q5zWCw8J4	Attempt exercises 1, 4, 6, 9 from P4E chapter 1
Thursday, January 16	Variables, expressions, and statements	Read chapter 2 of P4E	Attempt exercises 2, 3, 4 from P4E chapter 2
Tuesday, January 21	Conditionals	Read chapter 3 of P4E (sections 3.1-3.6)	Attempt exercises 1, 2 from P4E chapter 3
Thursday, January 23	Functions (part 1)	Read chapter 4 of P4E (sections 4.1-4.8)	Attempt exercises 1, 2, 3 from P4E chapter 4
Tuesday, January 28	Functions (part 2)	Read chapter 4 of P4E (sections 4.9-4.12)	Attempt exercises 4, 5 from P4E chapter 4
Thursday, January 30	Iteration (part 1)	Read chapter 5 of P4E (sections 5.1-5.4) and chapter 3 (section 3.7)	Attempt exercise 1 from P4E chapter 5
Tuesday, February 4	Iteration (part 2)	Read chapter 5 of P4E (sections 5.5-5.7)	Attempt exercise 2 from P4E chapter 5
Thursday, February 06	Abstraction and decomposition	Read Abstraction and How to Solve it on Wikipedia	
Tuesday, February 11	Review 1		Come prepared with questions

Date	Topic	To read before class	Exercise
Thursday, February 13	Strings	Read chapter 6 of P4E	Attempt exercise 6 from P4E chapter 6
Tuesday, February 18	Files	Read chapter 7 of P4E	Attempt exercise 1 from P4E chapter 7
Thursday, February 20	Lists (part 1)	Read chapter 8 (sections 8.1-8.7) of P4E	To be announced
Tuesday, February 25	Lists (part 2)	Read chapter 8 (sections 8.8-8.14) of P4E	Attempt exercise 1 from P4E chapter 8
Thursday, February 27	Dictionaries (part 1)	Read chapter 9 (sections 9.1-9.2) of P4E	Attempt exercise 1 from P4E chapter 9
Tuesday, March 3	Dictionaries (part 2)	Read chapter 9 (sections 9.3-9.5) of P4E	Attempt exercise 2 from P4E chapter 9
Thursday, March 5	Tuples	Read chapter 10 of P4E	Attempt exercises 1, 2 from P4E chapter 10
Tuesday, March 10	<i>Spring break (no class)</i>		
Thursday, March 12	<i>Spring break (no class)</i>		
Tuesday, March 17	Review 2		Come prepared with questions
Thursday, March 19	Regular expressions	Read chapter 11 of P4E	Attempt exercise 1 from P4E chapter 11
Tuesday, March 24	Networked programming	Read chapter 12 of P4E	Attempt exercises 1, 2 from P4E chapter 12
Thursday, March 26	Object-oriented programming	Read chapter 14 (sections 14.1-14.5) of P4E	
Tuesday, March 31	Classes and instances (part 1)	Read chapter 14 (sections 14.6-14.9) of P4E	To be announced
Thursday, April 2	Classes and instances (part 2)	Read chapter 14 (sections 14.10) of P4E	To be announced
Tuesday, April 7	Review 3		Come prepared with questions
Thursday, April 9	Visualization	Read chapter 16 of P4E	To be announced
Tuesday, April 14	Recursion	Read https://realpython.com/python-on-thinking-recursively/	To be announced
Thursday, April 16	Version control	To be announced	
Tuesday, April 21	Unit testing	Read https://jeffknupp.com/blog/2013/12/09/improve-your-python-understanding-unit-testing/	
Thursday, April 23	Review 4 and reflections		

Final examination: 12:00 noon on Friday, 1 May, 2020

Policies

Instructor communication

For specific, concrete questions, that you do not want to share on Piazza, e-mail is the most reliable means of contact for me. You should receive a response within a day or so, but sometimes it may take 2-3 days. If you do not receive a response after a few days, please follow up. Please keep this in mind when you are scheduling your own activities, especially those related to assignment or exam

preparation. If you wait until the day before a due date to ask me a clarification question, there is a good chance that you will not receive a response in time.

It is always helpful if your e-mail includes a targeted subject line that begins with "INLS 560." Please use complete sentences and professional language in your e-mail.

For more complicated questions or help, come to office hours (no appointment necessary!) or make an appointment to talk with me at a different time. I cannot discuss grades over e-mail; if you have a question about grading, you must talk with me in person.

You are welcome to call me by my first name ("Sayamindu" -- pronounced "Shayomindoo"). However, you may also use "Dr. Dasgupta" or "Professor Dasgupta" if that is more comfortable for you. Any one of those is fine.

Academic integrity

The UNC Honor Code states that:

It shall be the responsibility of every student enrolled at the University of North Carolina to support the principles of academic integrity and to refrain from all forms of academic dishonesty..

This includes prohibitions against the following:

- Plagiarism.
- Falsification, fabrication, or misrepresentation of data or citations.
- Unauthorized assistance or collaboration.
- Cheating.

All scholarship builds on previous work, and all scholarship is a form of collaboration, even when working independently. Incorporating the work of others, and collaborating with colleagues, is welcomed in academic work. However, the honor code clarifies that you must always acknowledge when you make use of the ideas, words, or assistance of others in your work. This is typically accomplished through practices of reference, quotation, and citation.

If you are not certain what constitutes proper procedures for acknowledging the work of others, please ask the instructor for assistance. It is your responsibility to ensure that the [honor code](#) is appropriately followed. (The [UNC Office of Student Conduct](#) provides a variety of honor code resources.)

The UNC Libraries has online tutorials on [citation practices](#) and [plagiarism](#) that you might find helpful.

Students with disabilities

Students with disabilities should request accommodations from the UNC office of Accessibility Resources and Service (<https://accessibility.unc.edu/>).

Acknowledgements and thanks

This syllabus includes elements of INLS 560 sections taught by Professor David Gotz.