

# INSC 360 Programming for Information Professionals

## COURSE INFORMATION

INSC 360, Spring 2021, 3 Credit Hours  
University of Tennessee, Knoxville  
Course Mode: On-campus, in-person;  
Class Meetings: BUE-334, MWF, 3:30PM – 4:20PM, EST  
ZOOM: <https://tennessee.zoom.us/j/6473366886>

### Faculty Contact Information

- Dr. Benjamin Horne, Assistant Professor
- he/him/his
- [bhorne6@utk.edu](mailto:bhorne6@utk.edu)
- Office Hours (by Zoom): 10am to 11:30pm Wednesdays

### SIS Office Information

- 450 Communications Bldg.
- 1345 Circle Park Drive
- Knoxville, TN 37996-0341
- SIS Office: 865.974.2148
- Fax (SIS): 865.974.4667

### Welcome Statement

Welcome to INSC 360, Programming for Information Professionals. This class is designed for students who have never programmed before. Hence, we will focus on the fundamentals of programming at a beginner's pace.

While the class will be taught in Python, I will put emphasis on computational thinking and solving data-oriented problems using programming. This course will set you up for future data analysis and visualization courses, as well as your future career. I have written data-oriented programs in Python nearly every week of my life since college (except for those weeks where I was writing my dissertation), and I will show you the fundamentals that I have used over and over again doing data analysis.

Programming is notoriously a difficult thing to learn without prior experience. I vividly remember struggling to understand it as an undergrad CS student. If you find parts of this class difficult, I encourage you to stick with it. Eventually things will click and I will be there to help you along the way.

Remember these two things when learning difficult skills:

1. Practice makes perfect.
2. Focus on learning, not on the grade.

## **COVID-19 Related Information**

As you already know, this semester will be different from other semesters with the continued COVID-19 outbreak. This outbreak will require us to be flexible as the semester moves along and follow some important safety rules, especially as we meet face-to-face. The following rules will apply to our class:

1. Masks must be worn at all times, even while you are socially distant from others in the classroom. Because of this rule, eating and drinking during class is prohibited.
2. Practice social distancing in the classroom. The university has worked hard to ensure all classrooms are setup with clear social distancing guidelines.
3. Speaking of where you will sit in the classroom, once your spot is chosen on the first day of class that is where you must sit the rest of the semester.
4. When you arrive in the classroom, its best practice to wipe off your desk/work area with a sanitary wipe and use hand sanitizer on your way into the classroom. I will be doing the same as I enter the room.

### **What constitutes an excused absence due to COVID-19?**

Examples of the need to self-isolate include:

- Having tested positive for COVID-19
- Developing symptoms of a COVID-19 infection
- Awaiting COVID-19 test results
- Having had close contact with someone known to be diagnosed with COVID-19
- Having been advised to self-isolate by a health authority
- Having recently returned to the US after traveling abroad
- Having recently returned from a cruise (ocean or river)

### **What does it mean for an absence to be excused?**

Scholars whose absence is excused may not receive an academic penalty for their absence and must be allowed to either make up any in-class assignment, quiz, or exam or to complete a substantially equivalent assignment, quiz, or exam.

### **When and how should scholars report they are self-isolating?**

If a scholar's self-isolation or health condition could affect their ability to participate in classes, they should communicate directly with their instructors before the scheduled class time. Self isolating students should also complete the online self-isolation form.

Scholars are being asked to follow the guidance on the coronavirus website on when to self-isolate and what to do if they feel sick.

- When to self-isolate: <https://www.utk.edu/coronavirus/guides/when-to-self-isolate>
- What to do if you feel sick: <https://www.utk.edu/coronavirus/guides/what-to-do-if-you-feel-sick>

## **COURSE INFORMATION**

## **Catalog Description**

This course is an introduction to object-oriented programming using Python with emphasis on data structures, in-built functions, user-defined variables, design, syntax, and control structures for processing and visualizing datasets.

## **Student Learning Outcomes**

Students who complete this course will be capable of demonstrating awareness, knowledge, and/or understanding of:

- Capable of independently setting up and configuring their own Python development environment on their operating system of choice.
- Familiarity with different data types and containers in Python.
- How to develop simple but robust programs that leverage flow control, exception handling, and logging features
- Identifying and implementing appropriate solutions for accessing data under a variety of circumstances, such as data on disk, breaking up very large files, using web services, and web scraping.
- Using Python libraries for plotting and visualization of data.

## **Required Text(s)**

There are no official required texts for this course, and I will not rely on a single textbook. Any readings that are required will be posted on Canvas. However, I recommend the following books for support outside of the lecture material:

- Practical Programming: An Introduction to Computer Science Using Python by Campbell, Gries, and Montojo (Second Edition)
- Python Crash Course: A Hands-On, Project-Based Introduction to Programming by Eric Matthes (Part 1 of the book may be helpful in class)

## **COMMUNICATION**

### **Email**

I am required to communicate with you through your UTK email address. If you prefer to use another address, consult the [OIT Helpdesk](#) to obtain directions for forwarding your UTK email to your preferred address if you do not wish to check both accounts.

### **Instructor Availability**

Feel free to email me with any questions or concerns. I will typically respond in one to two business days.

## **COMPUTING REQUIREMENTS AND RESOURCES**

## Requirements

Bring your laptop to every class, as there will be in-class programming assignments throughout each week.

For some parts of the class, we will use the Wing IDE (<https://wingware.com/>), but do not get attached to an IDE. IDEs (as we will discuss in class) are useful tools to help develop and debug programs. They are not necessary to program, and you should not become reliant on them. Another good IDE that you can use include PyCharm (<https://www.jetbrains.com/pycharm/>).

More importantly, everything will be done in Python 3, not Python 2. Maybe we will have time to discuss the historical battle between Python 3 and Python 2, but at least know they are different enough to matter in your homework assignments.

You must store your programs on some type of cloud storage (Google Drive, OneDrive, Dropbox, etc.)! There will not be time for “my dog ate my computer” like excuses. Please be organized in storing files.

## Course Resources

Everything for class will be hosted on Canvas.

## COURSE ATTENDANCE AND PARTICIPATION POLICIES

### Learner Expectations

- Be prepared for all classes
- Be respectful of others
- Actively contribute to the learning activities in class
- Abide by the UT Honor Code Instructor Expectations

### Instructor Expectations

- Be prepared for all classes
- Evaluate all work fairly and equitably
- Provide timely feedback
- Be respectful of all students
- Be responsive to student emails and requests for meetings
- Create and facilitate meaningful learning activities
- Behave according to University codes of conduct

### Attendance and Participation

As described in the Assignments section, you are expected to attend class and participate.

### Inclement Weather

The chancellor (or appointed representative) may officially close or suspend selected activities of the university because of extreme weather conditions. When a decision to close is made, it applies to all classes (whether on-campus or online). The information is distributed to the campus community, shared with local media, and posted on the University homepage at <http://utk.edu>.

## ADDITIONAL POLICIES AND POINTS OF INFORMATION

## **Disability Services**

Any student who feels they may need an accommodation based on the impact of a disability should contact [Student Disability Services](#) in Dunford Hall at 865-974-6087, or by video relay at 865-622-6566, to coordinate reasonable academic accommodations.

## **University Civility Statement**

Civility is genuine respect and regard for others: politeness, consideration, tact, good manners, graciousness, cordiality, affability, amiability and courteousness. Civility enhances academic freedom and integrity, and is a prerequisite to the free exchange of ideas and knowledge in the learning community. Our community consists of students, faculty, staff, alumni, and campus visitors. Community members affect each other's well-being and have a shared interest in creating and sustaining an environment where all community members and their points of view are valued and respected. Affirming the value of each member of the university community, the campus asks that all its members adhere to the principles of civility and community adopted by the campus. For more information, see the [UT Principles of Civility and Community](#).

## **CCI Diversity Statement**

The College of Communication and Information recognizes that a college diverse in its people, curricula, scholarship, research, and creative activities expands opportunities for intellectual inquiry and engagement, helps students develop critical thinking skills, and prepares students for social and civic responsibilities. All members of the College benefit from diversity and the quality of learning, research, scholarship and creative activities is enhanced by a climate of inclusion, understanding and appreciation of differences and the full range of human experience. As a result, the College is committed to diversity and equal opportunity and it recognizes that it must represent the diversity inherent in American society. The College is acutely aware that diversity and fairness are foundations that unite the College's faculty, staff, students, and the larger communication and information community.

## **Instructor Status as a Title IX Mandatory Reporter**

University of Tennessee faculty are committed to supporting our students and upholding gender equity laws as outlined by Title IX. Please be aware that if you choose to confide in a faculty member regarding an issue of sexual misconduct, dating violence, or stalking, we are obligated to inform the University's Title IX Coordinator, who can assist you in connecting with all possible resources both on- and off-campus. If you would like to speak with someone confidentially, the Student Counseling Center (865-974-2196) and the Student Health Center (865-974-3135) are both confidential resources. For additional resources and information, visit [titleix.utk.edu](http://titleix.utk.edu).

## **ASSIGNMENTS, ASSESSMENTS, AND EVALUATIONS**

### **Academic Integrity**

Students should be familiar with the [Hilltopics Student Handbook](#), and comply with all academic policies. This includes the University of Tennessee Honor Statement and the Academic Integrity Policy.

The Honor Statement reads: *"An essential feature of the University of Tennessee, Knoxville is a commitment to maintaining an atmosphere of intellectual integrity and academic honesty. 'As a student of the university, I pledge that I will neither knowingly give nor receive any inappropriate*

*assistance in academic work, thus affirming my own personal commitment to honor and integrity.*” For more information, see the Honor Statement on the Academic Policies and Procedures page of the current [catalog](#) for student and faculty responsibilities.

The Academic Integrity policy reads: “*Study, preparation and presentation should involve at all times the student’s own work, unless it has been clearly specified that work is to be a team effort. Academic honesty requires that the student present his or her own work in all academic projects, including tests, papers, homework, and class presentation. When incorporating the work of other scholars and writers into a project, the student must accurately cite the source of that work.*” For additional information, see the [Student Code of Conduct](#).

## **Plagiarism**

Plagiarism in any of its forms is intolerable, and attention to matters of documentation in all written work is expected and required. Inadvertence, alleged lack of understanding, or avowed ignorance of the various types of plagiarism are not acceptable excuses.

*Specific examples of plagiarism are:*

1. Copying without proper documentation (quotation marks and a citation) written or spoken words, phrases, or sentences from any source.
2. Summarizing without proper documentation (usually a citation) ideas from another source (unless such information is recognized as common knowledge).
3. Borrowing facts, statistics, graphs, pictorial representations, or phrases without acknowledging the source (unless such information is recognized as common knowledge).
4. Collaborating on a graded assignment without the instructor’s approval.
5. Submitting work, either in whole or in part, created by a professional service and used without attribution (e.g., paper, speech, bibliography, or photograph).

Students who may be unsure of the nature of plagiarism should consult the instructor or a guide for writing research reports. Resources are available through the University Libraries, including a [Citing Sources guide](#).

Infractions of academic integrity are penalized according to the severity of the infraction but may include a course grade of "F."

## **Assignments**

This table provides a brief summary of assignment by name, due date, point value and percentage of final grade. A brief description of each assignment follows the table.

<b>Assignment</b>	<b>Point Value</b>	<b>Percentage of Final Grade</b>	<b>Due Date</b>
In-Class Programming (12 @ 10 points)	120	24%	~ Every Week
Quizzes (6 @ 10 points)	60	12%	~ Every Other Week
Homework (4 @ 61.25 points)	245	49%	#1 - 02/08 #2 - 02/26 #3 - 03/19

Assignment	Point Value	Percentage of Final Grade	Due Date
			#4 - 04/09
Final Exam (1 @ 75 points)	75	15%	TBD
<b>TOTAL</b>	<b>500</b>	<b>100%</b>	

### **In-Class Programming (24%)**

The key to learning programming is practice and exposure. Hence, you are expected to complete programming in real time with me during class (**nearly every day**). At the end of each week, you will submit your in-class programming file. Please utilize in-class programming to take notes through comments. This will serve you well in the homework and studying for the final. **In-Class Programming cannot be made-up**

### **Quizzes (12%)**

During some of our class meetings, we will have short in-class quizzes. The format of these 'quizzes' will vary, with some essay and some multiple-choice questions. The goal of the quizzes will be to reinforce concepts we have gone over in class. In other words: “do you recognize this tool in your programming toolbox?” **Quizzes cannot be made-up.**

### **Homework (49%)**

Homework assignments are designed to give you an opportunity to apply, and engage with, the material from different sections of the class (roughly ¼ of the class for each homework).

- You will complete four homework assignments.
- **Start working on the homework early!**
- While I am okay with you working with others, you must turn in your own solutions to each of the problems. Be sure to understand the solutions yourself. You will be tested on the material.
- Do not copy something from Stackoverflow or use libraries that we have not discussed in class. Remember, focus on learning, not the grade.
- Follow the instructions for each homework carefully.
- Please upload a separate file for each problem. For some problems you will need to upload a Python file (.py), others you will just need to upload a text file (.txt). Please name your files: HWyProblemx\_FirstInitialLastName, where y is the homework number and x is the problem number.

### **Final Exam (15%)**

- Exam will be comprehensive.
- More details will be provided later in the semester.

### **Submitting Assignments, Late Assignments**

Assignments should be submitted to the “assignments” area of Canvas and are due (officially) at 11:59 p.m. EST on the due date listed on the Syllabus. Late assignments are not accepted unless prior arrangements have been made, or if you have an unexpected emergency. Quizzes and in-class programming have firm due dates and cannot be made-up.

## Grading Scale

Semester grades will be assigned according to the following scale:

A	93-100	Superior performance (4 quality points)
A-	90-92.99	Intermediate superior performance (3.7 quality points)
B+	88-89.99	Very good performance (3.3 quality points)
B	83-87.99	Good performance (3.0 quality points)
B-	80-82.99	Intermediate good performance (2.7 quality points)
C+	78-79.99	Fair performance (2.3 quality points)
C	73-77.99	Satisfactory performance (2.0 quality points)
C-	70-72.99	Unsatisfactory performance (1.7 quality points)
D+	68-69.99	Unsatisfactory performance (1.3 quality points)
D	63-67.99	Unsatisfactory performance (1.0 quality points)
D-	60-62.99	Unsatisfactory performance (0.7 quality points)
F	0-59.99	Failure performance (0.0 quality points)
S		Satisfactory; only assigned for C or better work when a course is taken on a S/NC grading basis. Carries no point value.
NC		No Credit; indicates failure to complete a course satisfactorily, and is only assigned for C- or worse work when a course is taken on a S/NC grading basis. Carries no point value.
I		Under extraordinary circumstances and at the discretion of the instructor, the grade of I (Incomplete) may be awarded to students who have satisfactorily completed a substantial portion of the course but cannot complete the course for reasons beyond their control. An I carries no quality points. If the I grade is not removed within one calendar year or upon graduation, it shall be changed to an F and count as a failure in the computation of the grade point average.
W		Indicates student has officially withdrawn from the course or the university. Carries no point value.

## Incompletes

Based on adopted University of Tennessee-Knoxville policy, a grade of *I* (Incomplete) is reserved for emergencies that prevent the student from completing the course on time. Incompletes are granted only under "the most unusual of circumstances" and solely at the discretion of the instructor. Plan your semester's course of study carefully to insure sufficient time to complete the required work. For students who simply "disappear" without contacting the instructor and without completing the required form, an "F" is submitted.

## COURSE EVALUATION

You will be invited by email to evaluate the course at the end of the term via TNVoice. Please participate in this valuable process. I also invite your comments throughout the course and read all comments, suggestions, and recommendations.

## DISCLAIMER



*The instructor reserves the right to revise, alter or amend this syllabus as necessary. Students will be notified in writing / email of any such changes.*

*Parts of this syllabus/course design/exercises are from or inspired by Dr. Sibel Adali's and Wes Turner's CS 1 course at Rensselaer Polytechnic Institute.*

## COURSE OUTLINE

Week	Date	Module	Topic	Class Activities
1	01/20	1	Course Overview	
--	01/22	2	Paradigms and Python: <ul style="list-style-type: none"> <li>• Programming Paradigms</li> <li>• How close are you to the machine?</li> <li>• Interpreting vs. Compiling</li> </ul>	<b>Due Before Class: Quiz 1</b> Reading: PP2E_CHP1
2	01/25	3	Python Setup: <ul style="list-style-type: none"> <li>• Installing Python</li> <li>• Editors and IDEs</li> <li>• Basic Syntax and Comments</li> <li>• Saving and Running</li> <li>• Hello World</li> </ul>	
--	01/27	4	Python as a Calculator 1: <ul style="list-style-type: none"> <li>• Expressions and values</li> <li>• Types</li> <li>• Precedence</li> </ul>	Reading: PP2E_CHP2 (2.1 - 2.3)
--	01/29	4	Python as a Calculator 2: <ul style="list-style-type: none"> <li>• Variables and memory</li> <li>• Errors</li> <li>• Typing directly in the interpreter vs. running programs; use of print</li> <li>• Documentation and variable names</li> </ul>	Reading: PP2E_CHP2 (2.4 - 2.9) <b>Submit In-Class Prog 1</b>
3	02/01	5	Python Strings 1: <ul style="list-style-type: none"> <li>• String basics</li> <li>• String operations</li> </ul>	Reading: PP2E_CHP4 (4.1 - 4.4)
--	02/03	5	Python Strings 2: <ul style="list-style-type: none"> <li>• Input to and output from your Python programs</li> </ul>	Reading: PP2E_CHP4 (4.5 - 4.6)
--	02/05	6	Python Functions and Methods: <ul style="list-style-type: none"> <li>• Numerical Functions</li> <li>• String Methods</li> <li>• Example: round() vs</li> </ul>	Reading: PP2E_CHP3 (3.1 - 3.2) <b>Submit In-Class Prog 2</b> <b>Quiz 2 in Class</b>

			.format	
4	02/08	6	Python Modules: <ul style="list-style-type: none"> <li>• Importing Libraries</li> <li>• Installing Libraries</li> </ul>	<b>Homework 1 is Due</b>
--	02/10	7	Writing your own function: <ul style="list-style-type: none"> <li>• Functions and Breaking Down Problems:</li> <li>• Arguments, parameters, and local variables</li> </ul>	Reading: PP2E_CHP3 (3.3 - 3.6)
--	02/12	7	Writing your own function 2: <ul style="list-style-type: none"> <li>• More complex functions</li> <li>• Scope and flow</li> <li>• “Pass-by” model</li> </ul>	Reading: PP2E_CHP3 (3.7 - 3.10) <b>Submit In-Class Prog 3</b>
5	02/15	8	Decisions 1: <ul style="list-style-type: none"> <li>• If statements</li> <li>• Assignment vs. equivalency check</li> </ul>	
--	02/17	9	Lists 1: <ul style="list-style-type: none"> <li>• Whats a list?</li> <li>• Indexing</li> <li>• Memory Model for Lists</li> <li>• Changing the Values</li> <li>• Functions on Lists</li> <li>• Appending, Inserting, Deleting</li> <li>• Lists of Lists</li> </ul>	Reading: PP2E_CHP8 (8.1 - 8.3)
--	02/19	9	Tuples: <ul style="list-style-type: none"> <li>• Tuple data type</li> <li>• Some more list practice</li> </ul>	Reading: PP2E_CHP11 (11.2) <b>Submit In-Class Prog 4</b> <b>Quiz 3 in Class</b>
6	02/22	10	While Loops	
--	02/24	9	Lists 2: <ul style="list-style-type: none"> <li>• List aliasing, lists and functions</li> <li>• For loops to operate on lists</li> <li>• Slicing to create copies of lists and to create sublists</li> </ul>	Reading: PP2E_CHP8 (8.4 - 8.7), PP2E_CHP9 (9.1 - 9.2)
--	02/26	8	Decisions 2: <ul style="list-style-type: none"> <li>• Boolean logic</li> </ul>	Reading: PP2E_CHP5 <b>Homework 2 is Due</b> <b>Submit In-Class Prog 5</b>

			<ul style="list-style-type: none"> <li>• Nested if statements</li> <li>• Assigning boolean variables</li> </ul>	
7	03/01	10	Controlling Loops: <ul style="list-style-type: none"> <li>• Ranges and control of loop iterations</li> <li>• Nested loops</li> <li>• Lists of lists</li> <li>• Controlling loops through break and continue</li> </ul>	Reading: PP2E_CHP9 (9.3 - 9.9)
--	03/03	11	Reading in and Writing out data: <ul style="list-style-type: none"> <li>• Pure Python</li> <li>• CSV module</li> </ul>	Reading: PP2E_CHP10 (10.1 - 10.3, 10.5)
--	03/05	13	Problem Solving and Design	<b>Submit In-Class Prog 6 Quiz 4 in Class</b>
8	03/08	12	Sets	Reading: PP2E_CHP11 (11.1)
--	03/10	12	Dictionaries 1: <ul style="list-style-type: none"> <li>• Dictionaries and dictionary operations</li> </ul>	Reading: PP2E_CHP11 (11.3)
--	03/12	12	Dictionaries 2: <ul style="list-style-type: none"> <li>• Dictionaries of string/set pairs</li> <li>• Converting dictionaries with one key to another</li> <li>• Combining information from multiple dictionaries</li> <li>• Storing attribute/value pairs.</li> </ul>	Reading: PP2E_CHP11 (11.4 - 11.6) <b>Submit In-Class Prog 7</b>
9	03/15	13	Putting it all together: Practice data analysis	
--	03/17	13	Putting it all together: Practice data analysis	
--	03/19	13	Putting it all together: Practice data analysis	<b>Homework 3 is Due Submit In-Class Prog 8 Quiz 5 in Class</b>
10	03/22	14	Classes and OOP review	
--	03/24	14	Classes: <ul style="list-style-type: none"> <li>• Define our own types and associated functions</li> </ul>	

			<ul style="list-style-type: none"> <li>• Initializer / Constructor</li> </ul>	
--	03/26	14	Classes: <ul style="list-style-type: none"> <li>• Calling class methods from within the class</li> <li>• Class objects storing other objects, such as lists</li> <li>• Lists of class objects</li> </ul>	<b>Submit In-Class Prog 9</b>
11	03/29	15	Plotting with Matplotlib	
	03/31	15	Plotting with Matplotlib	
	04/02		Spring Recess	
12	04/05	16	Parsing the Web: <ul style="list-style-type: none"> <li>• APIs</li> </ul>	
--	04/07	16	Parsing the Web: <ul style="list-style-type: none"> <li>• JSON</li> </ul>	
--	04/09	16	Parsing the Web <ul style="list-style-type: none"> <li>• Irregular tabular data</li> </ul>	<b>Homework 4 is Due Submit In-Class Prog 10 Quiz 6 in Class</b>
13	04/12	17	Handy Libraries ( <i>time permitting</i> ): <ul style="list-style-type: none"> <li>• Datetime</li> </ul>	
--	04/14	17	Handy Libraries ( <i>time permitting</i> ): <ul style="list-style-type: none"> <li>• Collections and Counter</li> </ul>	
--	04/16	17	Handy Libraries ( <i>time permitting</i> ): <ul style="list-style-type: none"> <li>• Pandas</li> </ul>	<b>Submit In-Class Prog 11</b>
14	04/19	18	Better, Faster, Stronger: <ul style="list-style-type: none"> <li>• Map and filter</li> <li>• Functions as parameters</li> <li>• Lambda functions</li> </ul>	
--	04/21	18	Better, Faster, Stronger: <ul style="list-style-type: none"> <li>• Stable sort</li> <li>• List comprehensions</li> <li>• Zip</li> </ul>	
--	04/23	18	Better, Faster, Stronger: <ul style="list-style-type: none"> <li>• Error Handling</li> <li>• Sleep</li> <li>• Logging</li> </ul>	<b>Submit In-Class Prog 12</b>

			<ul style="list-style-type: none"><li>• Fast/Effective Python</li></ul>	
15			Final	