Course Syllabus

Course Title
Foundations of Information Processing

Semester
Spring 2021

Course Number
IS430 – AO2/AO4/AOU

Instructor
Kevin Trainor

Teaching Assistant
Colleen Heinemann

Class Times
Online meetings will be held on Thursdays from 6:00 PM till 8:00 PM using Zoom. I look forward to joining you there. Please use a headset.

Online Lab Sessions
On Sunday mornings from 9:00 AM till 10:00 AM, I will be holding an optional Online Lab Session using Zoom. Please drop by to ask a question, to discuss solutions to previous assignments, to get help with the current assignment, to discuss the final project, or just to say hello. I will work with students on a first come, first served basis. Please use a headset.

If you need to arrange a private meeting with me, or if you need to arrange a lab session on a different day/time, please send me an *Individual Meeting Request* via the Service Desk (see *Contacting Instructor or TA* below).

Contacting Instructor or TA
The preferred method for contacting me or Colleen is by entering a request using the Service Desk for this course. PLEASE, DO NOT send requests to our regular email addresses.

The Service Desk for this course has been implemented using the JIRA Service Management product. If you are new to using the service desk in one of my courses, please visit the Service Desk Introduction for instructions and tips.

On an emergency basis, you may contact me using my mobile phone number: 847-650-9706.

Catalog Description
Covers common data, document processing, and programming constructs and concepts. Focuses on problem solving and abstraction with a programming language. By the end of the course students will be able to design, develop and test a moderately complex computer program to manage full text, bibliographic records or multimedia. The course prepares students for working with applications in data analytics, data science,

digital libraries, text mining and knowledge management. No prior programming background is assumed.

Course Description
This is a first course in computer programming using the Python programming language. The overall goal of this course is to develop an information processing perspective of the world. More specifically, the objectives are:

1. To gain a broad introduction to computer programming concepts, principles, styles and techniques through lectures and discussion.
2. To develop a working proficiency in a programming language through lab sessions and hands-on assignments.
3. To nurture the ability to detect opportunities to apply programming in new scenarios.

Course Topics
- Computers and Programs
- Writing Simple Programs
- Computing with Numbers
- Sequences: Strings, Lists, and Files
- Defining Functions
- Decision Structures
- Loop Structures and Booleans
- Simulation and Design
- Defining Classes
- Data Collections
- Algorithm Design
- Creating Custom Classes

Learning Outcomes
After completing this course, you should be able to:

- Understand the role that programming plays in information processing and management.
- Identify opportunities for the information worker to solve every day work problems using programming.
- Design, code, and test procedural programs in Python that make use of object-oriented classes provided by others.
- Design, code, and test simple object-oriented classes in Python.
- Use the PyCharm integrated development environment competently.

Required Texts
Zelle, J. (2016). Python Programming: An Introduction to Computer Science, 3rd Ed
ISBN: 9781590282755

Severance, C. (n.d.). Python for Everybody (https://www.py4e.com/book).
This text is available as an electronic book in several formats. Although it is also available as a printed book, I DO NOT RECOMMEND PURCHASING a printed copy. I invite you to take advantage of this author's generosity by using one of these electronic formats.

Technology Requirements
You will be completing coding assignments and conducting your final project using your own computer. I recommend that you use a computer that runs Windows 10 or a recent release of MacOS. While the software that we will be using for this course does run on Linux computers, there will be substantially less technical support available for Linux. If you want to use a Linux computer for your coursework, please contact me first.

You will be installing the following software on your computers for this class:

- The Anaconda open data science platform for Python
- The PyCharm integrated development environment (IDE) for Python

All of the software and services that you will be using are free for your use during this class. I will publish brief instructions and detailed tutorial videos to help you download, install, and begin using the required software on computers running Windows 10 and recent releases of MacOS. Links to these resources will be provided in our Weekly Schedule.

Course Schedule
The schedule for this course will be available via our Weekly Schedule at:

https://courseinfo.ligent.net/2021sp/_illinois/is430_ao2_ao4_aou/index.html

The course schedule is always subject to reasonable change to account for changes in circumstances and to correct errors. When I make changes to the schedule, I will announce them via our Moodle Announcements Forum. Postings to this forum should result in you being sent an email copy of the announcement as well.

Course Elements

1. *Readings*

   Required readings will be assigned from the text books listed above and from other resources that will be identified in the Weekly Schedule. Generally, readings are chosen to accompany any lecture video or tutorial video provided for the week. So, you should expect to complete the readings before playing the videos.

   This course has two required text books (Zelle and Severance). Most of the required reading assignments are from the Zelle text. A few required reading assignments are from the Severance text.

   In most weeks, optional readings are identified in the Weekly Schedule. Often, these are presentations of the same material from the other text book. I have found that reading a second approach to the same subject is particularly helpful for first time programmers and for anyone having difficulty with a particular topic. I expect you to use your best judgement in deciding whether the optional readings are adding appropriate value for you.

2. *Lecture Videos*

   I typically will not be using our online class time for lectures. Instead, I have provided links to pre-recorded video lectures for each unit that we cover. While most of the material covered in the recorded lectures is from the textbook, I occasionally cover supplemental material in the lectures as well. I always include commentary that I believe adds value to the text.

   While I have marked the lecture recordings for this course as *required*, I am leaving the decision to you regarding which combination of reading and playing lecture recordings will be the best investment for you. While many graduate-level courses do not include lectures on the material from the text, some graduate-level learners prefer learning by listening. I invite you to sample all of the content and then settle into a usage pattern that works best for you.

   Before playing my lecture videos, make sure that you have previously played *Tips on Playing My YouTube Videos* to assure that you get the most from your viewing experience.

3. *Tutorial Videos*

   I have created tutorial videos for the following use cases:

   - There are a number of tutorials that explain how to do activities necessary for the course. These include installing software, reading grading rubrics, submitting assignments, and related activities.
   - Many tutorials are coordinated with the exercises in your coding assignments. They represent a demonstration of the skill you will be expected to use when doing the exercise.

Because tutorial videos do not duplicate material covered by other resources, I recommend that you always play these recordings. Before playing my tutorial videos, make sure that you have previously played *Tips on Playing My YouTube Videos* to assure that you get the most from your viewing experience.

4. *Coding Assignments*

There will be weekly Coding Assignments. As mentioned above, coding assignments will often be paired with tutorial videos. These should allow you to complete your Coding Assignment using the same general approach that has been demonstrated in the video.

A Moodle submission activity will be provided for submitting each assignment. Instructions for each Coding Assignment and a grading rubric will be published in the Weekly Schedule.

Solutions to Coding Assignments will be posted to our Moodle site before our next online class session. You can expect us to review your solutions and mine at the beginning of our next class.

A major goal for this course is to build your proficiency in self-evaluation of your work. To build this skill, I will expect you to be able to estimate your grade on each Coding Assignment. The solutions to coding assignments posted to our Moodle site and our review of those solutions during the next class will serve as your primary feedback for the Coding Assignment.

As secondary feedback, your Coding Assignment submissions will be graded and commented upon. This feedback will be published to the Moodle assignment submission activity within 2 weeks.

The grading rubric for Coding Assignments has been designed to promote two important behaviors:

- Submitting your work in a properly named and formatted file. This helps substantially with grading workflow.

- Submitting your work by the week-ending deadline. This assures that you will get the benefit of having tried to solve the problem on your own before seeing the solutions of others.

While separate grading rubric and assignment submission instructions documents will be published, the following is a summary of the coding assignment grading rubric features:

- 10 points will be awarded for submitting a single, properly named and properly formatted file to the proper Moodle assignment submission activity.

- A minimum of 75 points will be awarded for submissions that are submitted on time, and that demonstrate a good faith effort on all parts of the assignment. Late submissions will be awarded 74 points or fewer in this category.

The implication of this grading scheme is that you can expect a score of 85 or higher on all Coding Assignments that meet both of these criteria.

5. *Participation*

Your participation grade will be earned based upon participation points that you will accumulate throughout the semester. The table below lists activities for which you may earn participation points and the points earned for each instance.

Activity	Points Earned
A *greetings* post made to the Service Desk for this course by the end of Week 2	10
1 post or reply made in the *Open Discussion* forum.	1
1 speaking contribution during class	2
1 chat contribution during class	1
1 presentation of your Coding Assignment solution during class	5
1 presentation as spokesperson for your a breakout group during class	5

Your participation grade for the course will be calculated at the end of the semester based upon the number of participation points earned. Grading will be done on a curve. A student with the highest number of participation points can expect to earn a grade of 100. A student with the median number of participation points can expect to earn a grade of 85. Students with fewer than 10 participation points can expect to earn a grade of 0.

6. *Final Project*

The Final Project will be a significant multi-part assignment to be completed in the final weeks of the semester. This assignment will require you to design, code, and test an application solution consisting of multiple programs and using techniques covered throughout the course.

The Final Project is an individual assignment, NOT a group assignment. More detailed instructions and a grading rubric for this assignment will be published separately.

Distinction Between 4-Credit and 2-Credit Enrollment
I recommend that you enroll in this course for 4 credits. If you enroll for 4 credits, you will be expected to submit all of the coding assignments (see above) as well as the final project (see below). I believe that this will lead to the best learning experience.

Nevertheless, you may choose to enroll for 2 credits. If you enroll in this course for 2 credits, you will be expected to submit only the coding assignments. Because I believe that the final project provides substantial value, I recommend against choosing this option.

Basis for Determining Grade
The various components of student work will contribute to the final grade based upon the following percentages:

Category	4-Credit Students	2-Credit Students
Participation	10%	10%
Coding Assignments	45%	90%
Final Project	45%	NA

Letter grades will be determined as follows:

- A+: 97 - 100%;
- A : 93 - 96%;
- A- : 90 - 92%;
- B+: 87 - 89%;
- B : 83 - 86%;
- B- : 80 - 82%;
- C+: 77 - 79%;
- C : 73 - 76%;
- C- : 70 - 72%;
- D+: 67 - 69%;
- D : 63 - 66%;
- D- : 60 - 62%;
- F : 0 - 59%;

ISCHOOL AND UNIVERSITY ACADEMIC POLICIES

Academic Integrity
The iSchool has the responsibility for maintaining academic integrity so as to protect the quality of education and research in our school and to protect those who depend on our integrity. Consequences of academic integrity infractions may be serious, ranging from a written warning to a failing grade for the course or dismissal from the University.
See the student code for academic integrity requirements:
http://studentcode.illinois.edu/article1/part4/1-401/

Statement of Inclusion
http://www.inclusiveillinois.illinois.edu/mission.html
As the state's premier public university, the University of Illinois at Urbana-Champaign's core mission is to serve the interests of the diverse people of the state of Illinois and beyond. The institution thus values inclusion and a pluralistic learning and research environment, one which we respect the varied perspectives and lived experiences of a diverse community and global workforce. We support diversity of worldviews, histories, and cultural knowledge across a range of social groups including race, ethnicity, gender identity, sexual orientation, abilities, economic class, religion, and their intersections.

Religious Observances
In keeping with our Statement of Inclusion and Illinois law, the University is required to reasonably accommodate its students' religious beliefs, observances, and practices in regard to admissions, class attendance, and the scheduling of examinations and work requirements.

If you anticipate the need for an accommodation, please communicate with your instructor in the first two weeks of class. If you are an undergraduate student and your instructor requires an absence letter, you must fill out the Religious Observance Accommodation Request form: https://cm.maxient.com/reportingform.php?UnivofIllinois&layout_id=19 . Other accommodations may be available.

Accessibility Statement
To obtain accessibility-related academic adjustments and/or auxiliary aids, students with disabilities must contact the course instructor and the Disability Resources and Educational Services (DRES) as soon as possible. To contact DRES you may visit 1207 S. Oak St., Champaign, call 333-4603 (V/TTY), or e-mail a message to disability@uiuc.edu .

COVID-19 Statement
In keeping with University and iSchool policy, all students are required to engage in appropriate behavior to protect the health and safety of our community. If you are on campus, this includes wearing a facial covering properly, maintaining social distance (at least 6 feet from others at all times), disinfecting the immediate seating area, and using hand sanitizer.

If you feel ill or are unable to come to class or complete class assignments due to issues related to COVID-19, including but not limited to: testing positive yourself, feeling ill, caring for a family member with COVID-19, or having unexpected child-care obligations, you should contact your instructor immediately and cc your advisor.

Contact Hours
This course will require approximately 54 contact hours.

Last Revised
2021-01-14