



Rutgers University
Graduate School of New Brunswick

16.137.552
Python Methodologies for Data Sciences (PMDS)
Hybrid Section
(5/20/2017 Revision)

Summer 2017 – 3 Credits
Hybrid
Office Hours: CAVE (Hill Annex, Rm 252, TBA)

Instructor: Lars Sorensen
Email: biglars@cs.rutgers.edu

Note: To reach me, you will be most successful using email.

MEETING DATES

This course is a hybrid that will incorporate SAKAI, online resources and four in person meetings. The in person meetings will be in Beck on Livingston campus (Rm 213). All meetings will be from 4:30pm to 6:30 pm on 5/30, 6/20, 7/11 and 8/1. Attending meetings is mandatory, plan now.

COURSE DESCRIPTION

Python is one of the most popular programming languages in the world. Used to teach programming at six of the top ten computer science programs in the U.S., Python has a reputation for being a well-supported language that is ideal for education. This support and quick learning curve has also made it popular among scientists. This course acts as an introduction to computer programming with the Python programming language. The basics of imperative programming will be covered as well as selected areas of computer science, object oriented programming and data structures. Computer programming is about problem solving so we will begin to think about how to solve problems in discrete steps like computers do. After the beginning of the course, when we have our sea legs, we will begin to introduce ideas from Data Science and use what we have learned about computer programming and problem solving in this area.

COURSE OBJECTIVES

Upon completion of this course, candidates can expect to:

1. Understand problem solving with computer programming, computational thinking and discrete algorithms.
2. Demonstrate experience with the Python programming language and its design environments.
3. Have the ability to create well documented computer programs that uses logical constructs and the syntax of the Python programming language.

OVERVIEW OF COURSE ACTIVITIES

The course includes weekly readings, some additional web content, announced quizzes on the readings, discussions in the discussion forums of SAKAI, a MOOC resource review project, a major project, and weekly programming assignments.

COURSE MATERIALS

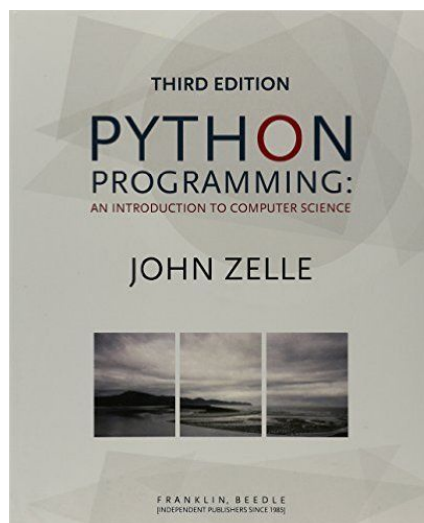
Zelle, John (2017). Python Programming: An Introduction to Computer Science (3rd Edition). Franklin, Beedle & Associates, Inc. (Referred to as Zelle or Z in the assignments).

ISBN-13: 860-1200643879

ISBN-10: 1590282418

Available on Amazon :

https://www.amazon.com/Python-Programming-Introduction-Computer-Science/dp/1590282752/ref=sr_1_1?ie=UTF8&qid=1483059077&sr=8-1&keywords=zelle+python



Other required readings available online and as referenced

**This course will be using Python 3.5 (I know the download page offers 3.6, we will discuss on 5/30)
We will be using IDLE as an IDE for examples in the course. You are free to use the IDE of your choice, but you will be responsible for it working properly and handed in assignment code will need to work in our grader's environment (you guessed it, IDLE)
If you do not know what IDLE is or an IDE is do not worry, all will be explained on 5/30.**

MAJOR COURSE REQUIREMENTS

- 1. Student discussion:** Students will participate in on-line discussions of the assigned readings. Unlike a traditional class where participation may be judged by attendance, in a distributed environment participation is documented by communiqués and information exchange among the class participants. You are encouraged to provide evidence for your statements with cited sources from your assigned readings. Basically, it will be a place to ask questions, make comments and, from time to time, display knowledge that lets me know you're learning. Students will be assigned groups from Unit 2 onwards and will have discussions with the members of their groups.
- 2. MOOC review paper:** Students will prepare a five page double spaced report on three pertinent websites/online tools identified and listed by the instructor. Reports are to be submitted through SAKAI and will later be posted there in a shared folder. **Due: 7/11**
- 3. Final project (group):** Through this project, students will create an authentic problem solving program indicating your *knowledge* (what should the programmer know?) and *skills* (what should the programmer be able to do?) representing the student's proficiencies at the end of the course. The final project will be a group effort and will consist of a Python programming assignment of the student group's choosing that completes a complex data science task. A rubric of programming features the project must employ/use (use of OOP, multiple libraries, use of inheritance, etc.) will be provided. Groups will submit project ideas for approval no later than :

The group project is to be submitted through SAKAI. **Due: 8/15**
- 4. Quizzes:** At the end of units there will be a simple quiz comprised of 3 or 4 multiple choice questions and one coding or essay answer response. Quizzes will be opened the last day of a unit (if unit ends 7/1 then 7/1 is quiz day, etc.)
- 5. Programming Assignments:** You will have one small and one medium size coding project per unit. This is done both in order to have one programming assignment per week and to catch a small break during the unit's first week as you absorb the content and ideas. Criteria for satisfactory completion of the assignments will be posted with the assignment. Both programs will be due at the end of the unit, but it is suggested that the small programming assignment be done during the first week of the unit. As the course progresses we will be exploring data science topics and algorithms through these assignments.

HYBRID COURSE TIPS

Communication

- Discuss, evaluate, and summarize content and context of reading assignments;
- Share resources, ideas, and make suggestions; and
- Discuss and clarify class activities.

I am readily available through email most of the time. In addition, I am available during the office hours posted above for scheduled conferences through Google hangout at other times as needed, including weekends. I'm a nightowl. If you ping me at 2am you may very well get a prompt response. Early AMs, say 9am, not so much ;-)

Guidelines for Email correspondence

- Use a descriptive subject heading
- Follow acceptable e-mail etiquette; and
- Be concise, write short sentences, and use bulleted lists.

Code of Ethics

Students of the university must conduct themselves in accordance with the highest standards of academic honesty and integrity. Plagiarism or violations of copyright policies are a form of academic dishonesty and are treated as ethics violations. The University Code of Student Conduct covers student rights and responsibilities, student complaint procedures, student misconduct, student disciplinary procedures, and other related information.

The Rutgers University Code of Student Conduct can be accessed at:

<http://studentconduct.rutgers.edu/university-code-of-student-conduct>

For information on the Rutgers University academic integrity policy, please go to:

<http://studentconduct.rutgers.edu/academic-integrity>

For any and all assignments and class activities, including quizzes, tests, papers, projects, PowerPoints, and any other class related work, **no copying of any kind is allowed**, unless copied text is placed within quotations and the author/source is appropriately cited. Excessive use of quoted material in essay assignments, quizzes, and tests is strongly discouraged, and this will tend to result in lower grades as candidates need to express themselves using their own language. Clear evidence of extensive plagiarism will likely result in a grade of F for the assignment AND the course.

Quizzes are open-note, open-book, and open-resource. Again, sources of information must be cited in any essay-type responses. Students are strongly encouraged to form study groups and email your instructor about parts of the readings for which you have questions. Quizzes are not a group activity, they are to be completed on your own.

COLLABORATION

Students are encouraged to study and learn together. Another student is often the best resource for working out a complex computation or understanding a difficult concept. However, **in PMDS all programming assignments turned in to fulfill assignments must be the exclusive work of the person submitting them, unless otherwise stated.** In order to allow students to work together, yet submit assignments that represent their own thought, the *Gilligan's Island* and the Freedom of Information rules are adopted.

The *Gilligan's Island* Rule: You are free to meet with fellow students(s) and discuss an assignment with them. Writing on a board or shared piece of paper during the meeting is acceptable; however, you should not take any written (electronic or otherwise) record away from the meeting. Everything that you derive from the collaboration should be in your head. After the meeting, engage in at least a half-hour of mind-numbing activity (like watching an episode of *Gilligan's Island**), before starting to work on the assignment. This will assure that you are able to reconstruct what you learned from the meeting by yourself.

The Freedom of Information Rule: To assure that all collaboration is on the level, you must always write the name(s) of your collaborators on your assignment.

* *Gilligan's Island* was a 1960s sitcom that set the standard for dim-witted TV. *Seinfeld* or anything more intelligent, e.g. some PBS programming, also works for this purpose.

LATE ASSIGNMENTS POLICY

All assignments are expected on time. Late assignments will receive a reduced grade, generally the equivalent of 10% of the total possible project grade per day late (or part thereof) with no chance of late submission after the third day. Students with bona fide emergencies should contact the instructor as soon as possible to discuss special arrangements in order to avoid grade penalties. The instructor will work constructively and collaboratively with students who communicate early regarding special problems completing assignments on time but will not be supportive of last minute requests for extensions on project due dates. **No late submissions for the final project components will be accepted.**

ACCOMMODATIONS FOR RELIGIOUS OBSERVANCE

I will of course accept a student's assertion of the need to be absent from class or unable to use a computer on a certain day for religious reasons, but students are required to notify me in advance that they will miss class or be unable to work in order to observe a religious holiday.

ACCOMMODATIONS FOR DISABILITIES

Rutgers University welcomes students with disabilities into all of the University's educational programs. To receive consideration for reasonable accommodations, a student with a disability must contact the appropriate disability services office at the campus where you are officially enrolled, participate in an intake interview, and provide documentation:

<https://ods.rutgers.edu/students/documentationguidelines>. If the documentation supports your request for reasonable accommodations, your campus's disability services office will provide you with a Letter of Accommodations. Please share this letter with your instructors and discuss the accommodations with them as early in your courses as possible. To begin this process, please complete the Registration form on the ODS website at:

<https://ods.rutgers.edu/students/registration-form>.

COURSE GRADING PLAN

Discussion: 30 points

MOOC review fact sheets: 60 points

Programming Assignments: 75 points

Quizzes: 45 points

Final project: 90 points

Course Grades: (out of 300 possible points)

GRADE	PERCENT	POINTS
A	93.00 – 100.0 %	Point totals will be converted to percentages. Any mid-course modification of course assignments that result in changes to possible point totals will be reflected accurately as percentage of total possible points earned.
B+	87.00 – 92.99 %	
B	80.00 – 86.99 %	
C+	77.00 – 79.99 %	
C	70.00 – 76.99 %	
F	Under 70 %	

GRADING REQUIREMENTS: ADDITIONAL DETAILS

Quizzes will be given at the end of units and completed by students during a specified time period. Quizzes and the open window for taking them will be announced in advance.

SCHEDULE OF UNITS AND CONTENT COVERAGE

Students are expected to keep pace with the posted schedule of readings, discussions, and assignments. Zelle does not sequence its content in the same way we will be seeing it (I do loops before functions) so we will be discussing and tweaking the readings along the way.

Unit	Date	Topic	Activity
1	5/30 – 6/8	Hello World, comments, Arithmetic, variables, assignments, keywords, input, conditional, Booleans, if-then-else control statements. The basics.	<ul style="list-style-type: none">• Zelle – Chapter 1, Chap 2 up to pg 43 (we'll get to loops in the next unit) plus the unit's google slides.• 2 programming assignments• Discussion group postings• Quiz• Online resources as assigned

2	6/9 – 6/18	Sequences – Strings, String manipulations, Lists, list manipulations and methods, Dictionaries Loops- Range function, for loops, while loops, indefinite and definite loops	<ul style="list-style-type: none"> • Zelle – Chapter 5, Chap 8 plus the unit’s google slides. • 2 programming assignments • Discussion group postings • Quiz • Online resources as assigned
3	6/19 – 7/6	Functions – def, scoping, return values, fruitful vs void Modularity – using libraries, import, custom libraries Files and file I/O – reading, writing, iterating Exception handling	<ul style="list-style-type: none"> • Zelle – Chapter 6, chapter 7 and chapter 3 plus the unit’s google slides. • 2 programming assignments • Discussion group postings • Quiz • Online resources as assigned
4	7/7 - 7/16	Computer Science topics – sorting, searching, recursion vs iteration, Big O Graphics – simple object graphics, events and event driven programming	<ul style="list-style-type: none"> • Zelle – Chapter 13 plus the unit’s google slides. • 2 programming assignments • Discussion group postings • Quiz • Online resources as assigned • Midterm Project Due July 11th
5	7/17 – 8/3	Object Oriented Programming – Classes, data, methods, constructors, inheritance, polymorphism	<ul style="list-style-type: none"> • Zelle – Chapter 10, chapter 12 plus the unit’s google slides. • 2 programming assignments • Discussion group postings • Quiz • Online resources as assigned • 7/21 - Proposals Due • 8/1 - Design Docs Due
6	8/4 – 8/13	Data Structures – simple stacks and queues, linked lists and binary trees	<ul style="list-style-type: none"> • Unit 6 google slides. • 1 programming assignment • Discussion group postings • Quiz • Online resources as assigned • REMEMBER – August 15th, Final project due