MET CS 521 - INFORMATION STRUCTURES WITH PYTHON: SYLLABUS

Instructor: Anatoliy Aleksandrov

BS/MS, Technology of Underground Mining and Construction, Moscow State Mining University, Moscow, Russia
PhD, Civil Engineering, Russian Academy of Sciences, Skochinsky Institute of Mining, Moscow, Russia

Email: aaleksan@bu.edu
Phone: 617-489-9061

Course Description

This course covers fundamental concepts of the object-oriented approach to software design and development using the Python programming language. It includes a detailed discussion of programming concepts starting with the fundamentals of data types, control structures methods, classes, arrays and strings, and proceeding to advanced topics such as inheritance and polymorphism, creating user interfaces, exceptions and streams. Upon completion of this course students will be capable of applying software engineering principles to design and implement Python applications that can be used in conjunction with analytics and big data.

Course Objectives and Learning Goals

The students should be able to

- Readily use the Python programming language
- Organize and modularize programs
- Understand and apply object-oriented program design and development
- Apply various data types and control structures
- Use class inheritance and polymorphism
- Create user interfaces
- Deal with exceptions
- Integrate Web access into applications
- Understand and begin to implement secure, robust, and scalable code

Resources

Contemporary programming languages like Python enjoy rich online documentation. Indeed, they are built on the premise that programmers are continually in contact with such documentation, and are not expected to memorize any but a small fraction of it. The textbook for the course is below but you should not have high expectations of it as a resource because of the dominance of online documentation and tutorials.

"Python Programming: An Introduction to Computer Science" by John M. Zelle, Franklin Beedle & Associates (December 2003), ISBN-10: 1887902996, ISBN-13: 978-1887902991

Week-by-Week Topics

Weekly Activities

Each week you will need to:

- Attend the lectures
- Read recommended pages in the textbook (listed below)
- Complete the interim assessment for feedback
- Complete the homework assignments
- Participate in discussions

Week	Module	Topic	Readings
1 and 2	1	Module 1: Introduction -- Relationship between computers and programs -- Data types, variables, expressions, and statements -- Using the Python interpreter -- Control Structures: loops and decision -- Numerical computing	Zelle Chapters 1, 2, 3, 7, 8
3 and 4	2	Module 2: Modularization and Files -- Defining functions -- Modules -- Packages -- Standard modules -- Reading and writing files	Zelle Chapter 6
5 and 6	3	Module 3: Classes and Object Oriented Design -- Functions and their arguments -- Defining Classes -- Object Oriented Programming -- Inheritance and Polymorphism	Zelle Chapter 10 (10.1, 10.3, 10.4, 10.5), 12
7 and 8	4	Module 4: Exceptions and User Interfaces -- Error processing -- Exception Raising and Handling -- Graphics -- Coordinates and color -- Data Structures (Arrays, Lists, Maps, and Iterators)	Zelle Chapter 4, 7.4,
9 and 10	5	Module 5: Data Structures -- Sets, dictionaries, and hashing -- Linked Lists -- Stacks -- Queues -- Database access	Zelle Chapter 11
Re-maining	6	Module 6: Multithreading and Web Access -- Multithreading and synchronization -- Networking -- Internet computing -- Web Access	Notes and references cited there: https://docs.python.org/2/tutorial/stdlib2.html#multi-threading

GRADING:
Programming assignments/Homework - 40%
Class Participation and Attendance - 10%
Midterm - 25%
Final - 25%

PROGRAM EVALUATION CRITERIA
Program correctness - 60%
Documentation - 20%
Readability - 20%