CSc 7003: Programming for Data Science

1. Instructor Information

Raj Sunderraman (raj@gsu.edu)
Office Hours:

TR 3.00 to 5.00 (Room 629, 1 Park Place)
Sat 1.00 to 3.00 (WebEx: https://gsumeetings.webex.com/meet/rsunderraman)

2. Class Information

Location: 205 Aderhold Learning Center

Time: TR 5.30 to 8.30 PM

3. Catalog Course Description:

This is an introductory and review course on programming in Python for Data Science and will provide the programming preparations for students in sharpening their programming skills. The course covers a variety of topics including algorithmic complexity, object-oriented programming, lists, hash tables, recursion, binary trees, heaps, sorting algorithms, and graphs.

4. Course Outcomes

By the end of the semester students will be able to:

- Understand how to formulate programming solutions in Python
- Understand how to measure space and time complexities of algorithms.
- Understand and use the concept of abstract data type to represent and implement heterogeneous data structures
- Write pseudocode with lists, hash tables, trees and graphs
- Grasp skills in analyzing and designing recursive algorithms and methods
- Learn the different searching and sorting algorithms
- Understand which data structures are most effective for different problems

5. Textbooks and Resources

Class website: http://tinman.cs.gsu.edu/~raj/7003/su22

References:

- Michael T. Gooodrich, Roberto Tamassia and Michael Goldwasser. Data Structures and Algorithms in Python, Wiley, 2013.

Note: It is not necessary to get this textbook; but is a good reference for Data Structures in Python. I will be providing online materials.

6. Attendance Policy

Students are required to attend all lectures. It is strongly suggested that students do not miss class, as most of the learning will happen in class in the form of problem solving in Python.

7. Grading

Programming Assignments: 100%

8. Course Schedule and Topics[1]

Python Basics:

Primitive Data Types
List, Tuple, String, Set, Dictionary, File

Data Structures:

Stack, Queue, Linked Lists, Priority Queues
Trees, Binary Search Trees, Huffman Encoding, Recursion
Sorting (Quick Sort, Merge Sort, Heap Sort)
Graphs

All of these topics will be illustrated through programming problems.

[1] This is a preliminary schedule. Some changes might be needed.