Course Syllabus

CS – 205 Data Structures and Algorithms I

Credits: 3 Contact hours: 3

Instructor or Course Coordinator: Jiacun Wang

Textbook(s):

M. Goodrich, R. Tamassia and M. Goldwasser, *Data Structures and Algorithms in Java (6th edition)*, Wiley, 2014.

**Course description:**

Introduction to the design, implementation, and use of elementary data types (list, stack, queue, binary tree); algorithms for sorting and searching; and informal complexity analysis.

**Prerequisite(s):**

Computer Science 176, passed with a grade of "C" or higher and either Mathematics 120, 130, or Computer Science 202, passed with a grade of "C" or higher.

Specific outcomes of instruction:

When students complete this course they should be able to:

* Identify a class of solutions (algorithms) given a problem
* Create data structures on their own (not using libraries)
* Design algorithms to correctly solve problems which need these data structures for their solutions
* Analyze the complexity of algorithms and how using various data structures will change the complexity of the algorithms
* Use Java Collections in the context of this course

Relationship of course to student outcomes listed in criterion 3:

* Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline. (Learning Outcome 2)

Topics covered:

* Introduction to algorithms and a review of Java basics
* Informal asymptotic analysis
* Arrays and searching
* Binary search and recursion
* Sorting and recursion
* Linked lists
* Stacks, queues, and dequeues
* Heaps and priority queues
* Object-oriented programming and Java container classes
* Advanced sorting (Quicksort, Mergesort, Shellsort)
* Linear-time sorting (Counting Sort, Radix Sort)
* Recursion and trees
* Sorted containers: sets and multisets
* Hash tables
* Binary trees and the TreeMap class
* Associative containers: HashMap versus TreeMap
* Trees and tree traversal