CS 3350: STANDARD DATA STRUCTURES AND ALGORITHMS

Semester Hours: 3.0
Contact Hours: 3

Coordinator: Winnie Rex

Text: Data Abstraction and Problem Solving With C++, 6/E

Author(s): FRANK CARRANO & TIMOTHY HENRY

Year: 2013

SPECIFIC COURSE INFORMATION

Catalog Description:
Advanced programming concepts. Adaptation and use of standard class libraries and generic algorithms. Prerequisite: Math 2220 or MATH 3220 and grade of C or better in CS 2020.

Course type: REQUIRED

SPECIFIC COURSE GOALS

- I can solve computational problems using recursion
- I can implement and apply stacks, queues, trees, and other custom data structures
- I can create generic functions and classes
- I understand algorithmic complexity (e.g. Big “O” notation)
- I can understand the relationship between data structures and algorithms
- I can understand the design tradeoffs (e.g., code complexity and performance) in data structures and algorithms

STUDENT OUTCOMES ADDRESSED BY THIS COURSE

- B.1 Analyze a given problem, and identify and define the computing requirements appropriate to its solution
- B.2 Use current techniques, skills, and tools in computing practice
- B.3 Apply mathematical foundations, algorithmic principles, and computer science theory as appropriate in modeling and solving real-world problems
- B.6 Communicate effectively with a range of audiences using oral, written, and electronic mediums
LIST OF TOPICS COVERED

- C++ review (arrays, classes)
- Recursive functions
- Advanced C++ topics (pointers, copy constructors, overloading functions, abstract classes, base classes, derived classes and friends)
- Exception handling
- Templates (function and class templates)
- Adaptation and use of Standard Template Library data structures (iterators, vectors, strings, dequeues, heap trees, hash tables)
- Use of Standard Template Library Algorithms (sorting, shuffling, permuting)
- Sorting algorithms
- Big-O notation