CS 4521: Algorithms and Data Structures (4)

Catalog Description:
Asympototic analysis of algorithms. Methods for proving correctness. Implementation of algorithms. Survey of algorithms and data structures, such as: heaps and heapsort, quicksort, binary search trees, red-black trees, B-trees, hash tables, graph algorithms, dynamic programming, and greedy algorithms.

Textbook: Thomas Cormen, Charles Leiserson, Ronald Rivest and Clifford Stein, *Introduction to* Algorithms, 2nd Ed., MIT Press, 2001.

References:

Course Goals:
Advanced computer programming requires an understanding of both the objects on which programs work, that is, data structures, and the sequences of steps driving the programs themselves, that is, algorithms. In order for data structures and algorithms to be generally useful, they often must overcome memory and speed limitations of the underlying machine. Thus it is important for the programmer to be able to analyze algorithms for their efficiency in execution and use of space for data. This course will approach the study of data structures and algorithms with an eye toward both analyzing efficiency and developing working models using sound data abstraction and procedural abstraction practices.

Prerequisites by Course & Topic
CS 2511: Software Analysis and Design.- implementation experience, data abstraction, algorithm analysis
MA 3355: Discrete Math - proof techniques, recursion and mathematical induction, recursive algorithms, analysis of algorithms, assertions and loop invariants, complexity measures of algorithms, combinatorial counting techniques

Major Topics Covered in the Course
- Asymptotic Analysis and Correctness Proofs
- Algorithms and Data Structures

Class/Laboratory Schedule: Lecture: 3 hours per week, Laboratory: 1

Laboratory Projects
- Algorithms, Growth of Functions (2)
- Growth of Functions, Comparison of Sorts (2)
- Binary Heaps, Heapsort, and Quicksort (2)
- Binary Search Trees and Red-Black Trees (2)
- Red-Black Tree Implementation, B-Trees (2)
- B-Trees, Binomial Heaps (2)
- Binomial Heaps, Amortized Analysis (2)

Course Contribution to Program Objectives and Outcomes:
1. Further develop ability to understand, apply and prove precise mathematical claims about space and time requirements of algorithms, and about correctness of algorithms. (*a,b,c,d,e*)
2. Detailed knowledge of a number of standard algorithms, and data structures, including knowledge of time and space complexity, ability to prove (or understand proofs) of properties of data structures and associated algorithms, and also the ability to implement such algorithms.. (*a,b,c,d,e*)

Estimate CSAB Category Content

	CORE	ADVANCED
Data Structures		2
Algorithms		2
Software Design		

	CORE	ADVANCED
Computer Organization and Architecture		
Concept of Programming Languages		

Theoretical Content

- Asymptotic Analysis (3 weeks)
- Proofs of Correctness (1 week)
- Amortized Analysis (1 week)
- Algorithms and Data Structures (10 weeks)

Solution Design

Students implement a selection of standard and advanced algorithms and data structures.

Coordinator/Prepared by: H. Turner