

Course Title and Code	CS211 - Concepts of Algorithms
------------------------------	---------------------------------------

I. Course Identification and General Information:

Course Title	Concepts of Algorithms	Course Code	CS211	Pre-requisite	CS181
Department	Computer Science	Course Level	5	Credit Hours	3(3+0)

II. Course Description/Topics: The following course topics will be covered.

- The Role of algorithms in computing - Algorithms as a technology - Analyzing and designing algorithms
- Growth of functions - Asymptotic notation - Standard notations and common functions
- Recurrences - The substitution method - The recursion-tree method - Proof of the master theorem
- Probabilistic analysis and randomized algorithms - The hiring problem - Indicator random variables - Randomized algorithms - Probabilistic analysis and further uses of indicator random variables
- Sorting: Bubble sort - Heap sort – Quick sort – Sorting in linear time
- Medians and Order Statistics - Minimum and maximum - Selection in expected linear time - Selection in worst-case linear time
- Hash tables – Direct-address tables - Hash functions - Open addressing - Perfect hashing
- Binary search trees – Querying a binary search tree Insertion and deletion – Randomly-built binary search trees
- Red-Black trees - Properties of red-black trees - Rotations - Insertion – Deletion
- Augmenting data structures - Dynamic order statistics - How to augment a data structure - Interval trees

III. Course Outcomes: Summary of the main learning outcomes for students enrolled in the course.

By the end of the course, one should be able to:

- Develop an ability to write a pseudo code for a given problem.
- Have facility in mapping a pseudo code into implementation of examples of algorithmic strategies from scratch, and applying them to specific problems.
- Implement basic numerical algorithms.
- Determine informally the time and space complexity of simple algorithms.
- Solve elementary recurrence relations, e.g., using some form of a Master Theorem.
- Use a heuristic approach to solve an appropriate problem.
- Describe the trade-offs between brute force and other strategies.
- Implement simple search algorithms and explain the differences in their time complexities.
- Be able to implement common quadratic and $O(N \log N)$ sorting algorithms.
- Be able to write efficient programs through the use of concise and efficient algorithms.
- Tackle any programming problem by breaking it into its component parts.
- Implement any algorithm using any programming language of one's choice.

IV. Required Text

- Cormen, T., C. Leiserson, R. Rivest and Clifford Stein, "Introduction to Algorithms", 2nd Edition MIT Press, 2002

V. Reference

- Goodrich, M.T. and R. Tamassia, "Algorithms Design, Foundations, Analysis and Internet Examples", John Wiley & Sons, 2002