**CSE 421** Introduction to Algorithms

## Credits
3.0 (3 hrs lecture)

## Lead Instructor
Anup Rao

## Textbook

- *Algorithm Design*, Kleinberg

## Course Description
Techniques for design of efficient algorithms. Methods for showing lower bounds on computational complexity. Particular algorithms for sorting, searching, set manipulation, arithmetic, graph problems, pattern matching.

## Prerequisites
either CSE 312 or CSE 322; either CSE 326 or CSE 332.

## CE Major Status
Selected Elective

## Course Objectives
Learn basic techniques for design and analysis of algorithms, including correctness proofs. Learn a number of important basic algorithms. Learn how to prove that problems are NP-complete.

## ABET Outcomes
(a) an ability to apply knowledge of mathematics, science, and engineering

## Course Topics

- Main Techniques:
    - Design: Induction, Graph search, Divide and Conquer, Greedy, Dynamic Programming, Network Flow
    - Analysis: Asymptotic Analysis, Recurrences.
    - Intractability: Reduction.
- Typical Algorithm coverage:
    - depth- and breadth-first search
    - bi- and/or strongly connected components
    - shortest paths

- o min spanning trees
- o transitive closure
- o flows and matchings
- o Strassen's method
- o FFT
- o knapsack
- o edit distance/string matching
- o scheduling
- Intractablity:
  - o reduction
  - o P / NP
  - o verification/certificates/witnesses,
  - o nondeterminism,
  - o completeness
  - o Example problems:
    - ▪ SAT
    - ▪ 3-SAT
    - ▪ clique
    - ▪ vertex cover
    - ▪ 0-1 knapsack
    - ▪ partition
    - ▪ coloring