Instructor: Puneet Bhateja

Course Description: This course intends to provide a rigorous introduction to fundamental techniques in the design and analysis of algorithms. The course can be divided into five parts namely (1) Foundations, (2) Sorting and Order Statistics, (3) Advanced Design and Analysis Techniques, (4) Graph Algorithms and (5) Special Topics. In the Foundations part, we will overview asymptotic notation, divide and conquer techniques, solving recurrences, probabilistic analysis and randomized algorithms. In the Sorting and Order Statistics part, we will cover heapsort, quicksort, sorting in linear time and median order statistics. In the Advanced Design and Analysis Techniques part, we will cover dynamic programming, greedy algorithms and amortized analysis. In the Graph Algorithms part, we will cover depth first search, breadth-first search, bi-connectivity and strong connectivity, topological sort, minimum spanning trees, shortest paths and maximum flow. Finally, in special topics depending on the availability of time, we will cover the theory of NP-completeness and some standard string matching algorithms.

Outcomes and Objectives: Introduce students to some of the algorithm design techniques that are useful for solving problems in Computer Science.

Course Requirements: Every week there will be three hours of lecture and two hours of Laboratory. The problem sets will consist of mostly theoretical exercises. The theoretical exercises are intended to help students apply and master elementary concepts in the design and analysis of algorithms. The students are encouraged to implement the algorithms that they design and analyze during the labs.


Exams: All exams are cumulative
There will be 2 midterms (30 marks each) and a final exam (40 marks) consisting of problems from both topics covered in lectures.

Grading Criteria: TBD