**Course description** (from the graduate catalog): Analyzes concepts in algorithm design, problem solving strategies, proof techniques, complexity analysis, upper and lower bounds, sorting and searching, graph algorithms, geometric algorithms, probabilistic algorithms, intractability and NP-completeness, transformations, and approximation algorithms.

Special emphasis will be placed on **problem solving**, unifying ideas, proof techniques, the “scientific method”, as well as striving for elegance, insights, and generalizability in developing algorithms and proofs.

**Prerequisites**: CS432 or equivalent


Supplemental reading: “*How to Solve It*”, by George Polya, Princeton University Press

Selected papers at: [http://www.cs.virginia.edu/~robins/CS_readings.html](http://www.cs.virginia.edu/~robins/CS_readings.html)

**Office hours**: Tues & Thur 4:45-7pm (after every class lecture), and other times by appointment

Also Email Q&A, and a running course-related blog

**Class structure**: two exams (midterm and final), several problem sets, with problems taken from the textbook and other sources, and a term project (involving implementing and demoing some algorithms and/or related concepts and ideas). Extra credit will be given throughout the semester for solving challenging problems.

We will cover as many selected topics from the following list as time permits:

**Fundamentals:**
- Review of asymptotics
- Review of basic data structures
- Review of basic algorithms

**Sorting and searching:**
- Review of classical sorting
- Interpolation Search
- Specialized sorting methods
- Deterministic $K^{th}$ selection
- Lower bounds on max & min
- Majority detection
- Meta algorithms

**Advanced data structures:**
- Skip lists
- Amortized analysis
- Fibonacci heaps
- Perfect hashing, cuckoo hashing

**Graph algorithms:**
- Lowest common ancestor
- Minimum spanning trees
- Shortest paths trees
- Radius-cost tradeoffs
- Steiner trees
- Minimum matchings
- Network flows
- Degree-constrained trees

**Numerical algorithms:**
- Linear programming
- Matrix multiplication
- Karatsuba’s algorithm

**Distributed algorithms:**
- Distributed models
- Asynchronous consensus impossibility
- Leader election in a ring
- Leader election in graphs
- Distributed MSTs
Computational geometry:
• Lower bounds
• Chan's convex hull algorithm
• Segment intersection
• Planar subdivision search
• Voronoi diagrams
• Nearest neighbors
• Geometric minimum spanning trees
• Delaunay triangulations
• Minimum density trees
• Minimum bounding box
• distance between convex polygons
• Smallest Enclosing Circle
• Triangulation of polygons
• Collinear subsets
• Probabilistic analysis

String matching:
• Knuth-Morris-Pratt
• Boyer-Moore
• Edit distance
• Longest increasing subsequence
• Smith-Waterman algorithm

NP-completeness:
• Polynomial time and intractability
• Space and time complexity
• Problem reductions
• NP-completeness of satisfiability
• Independent sets
• Graph colorability
• Travelling salesperson problem
• Approximation heuristics