Course Outline:

- Introduction and Fibonacci Numbers.
- Divide and conquer: integer multiplication.
- Number theory and cryptography
  - Euclid’s GCD algorithm and Modular arithmetic.
  - Modular exponentiation, Factoring versus Primality testing.
  - Fermat’s little theorem and randomized primality testing.
  - RSA public-key cryptosystem
- Fast fourier transform.
- Interpolation of polynomials.
- Error correcting codes, and secret sharing.
- MIDTERM I
  - Graph Algorithms
    - Depth-first search.
    - Strongly connected components, 2SAT.
    - Breadth first search, Dijkstra’s algorithm.
    - Bellman-Ford Algorithm.
    - Minimum spanning trees
    - Union find
    - Huffman coding
- Dynamic programming
  - Longest common subsequence, chain matrix multiplication.
  - string matching, and other examples
- MIDTERM II
  - Linear programming
    - Problem definition and solution by improvement.
    - Reductions to linear programming.
    - network flows, maximum matching.
- NP-completeness
- Cook’s theorem, Satisfiability, Traveling Salesman problem.
- Techniques for proving NP-completeness.
- Branch and Bound
- Approximation algorithms
- Simulated annealing, go with the winners
- zero-knowledge protocols.