

## *Spring 2014*

**Math 53:** A course in vector calculus, covering basic material such as partial derivatives, multiple integrals, and most importantly Green's Theorem, Gauss' Theorem, and Stokes' Theorem.

**Engineering 7:** A course teaching the MATLAB language and its applications to engineering at large along with its relevance to Linear Algebra. The class culminates in a large coding project that involves coding a robot to compete against other students in a class-wide competition.

**Math 54:** Basic Linear Algebra concepts, covering Vector Spaces, Inner Product Spaces, Eigenvectors and Eigenvalues, Linear Transformations, Differential Equations, and Fourier Series.

## *Fall 2014*

**Computer Science 61B:** A Computer Science course about Data Structures taught in the Java language. Topics include data structures, searches, traversals, Minimum Spanning Trees, Shortest Path Algorithms, and Dynamic Programming. Projects throughout the semester included an algorithm for programming the game "2048," creating a database similar to what one would find using SQL, recreating the game "KJumpingCube," and a project involving the creation of Directed Graphs and solving them using Dijkstra's algorithm and A\* algorithm.

**Computer Science 70:** A basic course in discrete mathematics probability theory with several applications to computer science, such as RSA encryption, modular arithmetic, proofs, Chinese Remainder Theorem, interpolation, and more.

## *Spring 2015*

**Math 110:** A rigorous proof based course in Linear Algebra, building upon what is learned in Math 54. Students cover the same topics and go beyond the familiar delving into Dual Spaces, Jordan Form, Adjoints, Markov Chains, and numerous proofs and intuition on how to prove properties of Linear Algebra.

**Statistics 134:** A more complex course in probability theory, beginning in discrete probability, moving onwards to calculus based theory. Strong emphasis is placed on distributions of events, in both recognizing a distribution given a situation and interpreting it as it presents itself.

**Industrial Engineering 160:** This is an introductory course to Industrial Engineering covering Linear Programming, Integer Programming, Nonlinear Programming, Minimum Cost Network Flows, Max Flow Algorithms, Shortest Path Algorithms, Minimum Spanning Trees, and the Branch and Bound Method. The course culminates in a coding project in the AMPL language, involving Linear Programming and modeling of a real world problem.

**Engineering 120:** This is a finance course, teaching how to discount cash flows, conversion of interest rates, value investments using Net Present Value, Internal Rate of Return, and the Payback Rule, as well as Bonds, both zero coupon and coupon, Stocks and the Dividend Discount Model, Annuities, Perpetuities, Market Risk and Individual Risk, Diversifying Portfolios, Identifying the Optimal Portfolio using the Sharpe Ratio, and many more financial topics.

### *Fall 2015*

**Industrial Engineering 115:** This is a course that is concerned with industrial and commercial databases. Throughout the year, we work closely with a local client, in our case the Sierra Nevada Torpedo Room in Berkeley, CA, to design a database in Microsoft Access to cater to their needs. The course covers EER Diagrams, Relational operators, logic, and many other organizational tools.

**Industrial Engineering 160:** This course covers nonlinear programming and optimization, focusing on search algorithms, minimum cost network flows, integer programming, branch and bound method, optimization techniques and more.

**Industrial Engineering 171:** This course teaches organizational behavior skills. Focusing on teaching the concepts of emotional intelligence, awareness to situations, and many case studies of real world occurrences, IEOR 171 lends itself to management skills with an engineering twist.

**Math 104:** This course is covers real analysis of calculus, sequences, series, and the real number line. Teaching many problem solving and analytical skills, the course is very proof intensive and teaches students how to think of math in a more generalized manner.

### *Programming Languages:*

Exposure to: Python & JavaScript

Experience in: Java, MATLAB, AMPL, HTML, and CSS