



University of New Mexico Computer Science Department
CS 261 - Mathematical Foundations of Computer Science
Summer 2005

Taught by:

Joel Castellanos

FEC 319

e-mail joel@cs.unm.edu

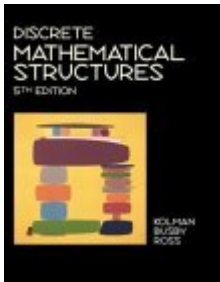
Course Description (from UNM Catalog):

This course is an introduction to the formal mathematical concepts of computer science for the beginning student. Topics include elementary logic, set theory, relations, deduction, induction, algorithmic processes, graph theory, and models of computation.

Prerequisites:

CS-152L (Computer Programming Fundamentals) and Math-162L (Calculus I).

Course Textbook:



Discrete Mathematical Structures, 5th Edition. By Kolman, Busby, and Ross

Course Information, Assignments, and Lecture Schedule

All course information, assignments, lecture notes, and practice exams can be accessed from the web through the instructor's homepage:

<http://cs.unm.edu/~joel/>

Grading:

Homework problem sets will be assigned Mondays, Tuesdays, and Wednesdays.

Homework will be neither turned in nor graded. Students are highly encouraged to work together on the homework. There will be a quiz each Monday (Tuesday on the week of July 4) covering the homework that was due the previous week. There will be a total of 7 quizzes. Each quiz will count as 10% of the final course grade with the lowest two quiz grades dropped. The final exam will be a two hour compressive exam and count as 50% of the final course grade.

CS 261 - Tentative Syllabus:

Date	Sections	Topics
6/6	1.1, 1.2	Sets, Ven diagrams, Infinite Sets, Operations on Sets
6/7	1.3, 1.4	Sequences, Regular Expressions, binary, hexadecimal
6/8	1.5	Matrices: Addition, Multiplication, Boolean Operations
6/9	1.6, 2.1, 2.2	Intro to Mathematical Structures (abstract operations, and commutative, associative, and closure properties), Propositions, Logical and Conditional Statements.
6/13	2.3, 2.4	Methods of Proof (Rules of Inference, Indirect Method, Proof by Contradiction, and Induction).
6/14	3.1, 3.2	Permutations, and Combinations
6/15	3.3, 3.4	Pigeonhole Principle and Probability (Sample Spaces, Probability using one die and several dice, the Shell Game Problem.
6/16	3.5	Recurrence Relations
6/20	4.1, 4.2, 4.3	Product sets, set partitions, Relations, Matrix of a Relation, and Digraph of a Relation.
6/21	4.3,4.4	Paths in Relations and Digraphs, Symmetric, Asymmetric, Antisymmetric, and Transitive Relations,
6/22	4.5	Equivalence Relations and Equivalence Classes
6/23	4.6	Computer Representations of Relations and Digraphs
6/27	5.1, 5.2	Function properties (onto, one-to-one, invertible), Special Computer Science Functions (Characteristic, floor, ceiling, exponential, Boolean, Hashing)
6/28	5.3	Growth of Functions: Rules for determining big-Oh, and big-theta
6/29	5.4	Permutation Functions: cyclic, transposition codes
6/30	6.1	Partially Ordered Sets, Hasse Diagrams, Topological Sorting, and Isomorphism.
7/5	6.2	Extremal Elements of Partially Ordered Sets
7/6	6.3	Lattices
7/7	6.4, 6.5	Finite Boolean Algebras: Substitution Rule, Boolean Polynomials, and logical Diagrams (<i>and</i> , <i>or</i> , and <i>inverter</i> gates)
7/11	6.6	Circuit Design
7/12	7.1, 7.2	Trees: Vertices, Leaves, Siblings, Descendants, Height, and Binary Positional Trees
7/13	7.3, 7.4	Tree Searching, Undirected Trees, and Spanning Trees of Connected Relations
7/14	7.5	Minimal Spanning Trees: Prim's Algorithm, Kruskal's Algorithm
7/18	8.1	Graphs: Vertex Degree, Simple Circuits, Connected, and Quotient
7/19	8.2, 8.3	Euler Paths and Circuits, and Hamiltonian Paths and Circuits
7/20	8.4	Transport Networks: A Maximum Flow Algorithm
7/21	8.5, 8.6	Matching Problems and Graph Coloring
7/25	10.1, 10.2	Languages: Derivation Trees, Context-free Grammars, Context Sensitive Grammars, Backus-Naur form notation, Syntax Diagrams, and Regular Grammars.
7/26	10.3, 10.4, 10.5, 10.6	Finite-State Machines: State Diagrams, Moore Machines, Monoids, Machines and Regular Languages, and Simplification of Machines.
7/27		Review
7/28		Final Exam (2 hour, Comprehensive)