Math 210
Final Exam Review Sheet

Part I: Logic

• Understand propositions and logical operators, and be able to build their truth tables.
• Be able to translate back and forth between English statements and symbolic logical propositions.
• Be able to prove the logical equivalence of a pair of propositions by building truth tables.
• Know how to use a 2-column proof to determine whether a given proposition is a tautology or to show that two propositions are equivalent.
• Be able to negate propositions using De Morgan's Laws and other equivalences.
• Understand predicates and quantifiers and be able to determine the truth value of statements involving predicates and/or quantifiers. Also be able to translate English statements into statements involving one or more predicates and quantifiers.
• Be able to negate statements involving predicates and quantifiers.
• Know the standard rules of inference for arguments with statements involving propositions and for arguments with statements involving predicates and quantifiers.

Part II: Proof Methods

• Understand how to use a *counterexample* to show that a statement is false.
• Understand the role of counterexamples and conjectures in proving and disproving statements.
• Understand the principle of mathematical induction and be able to prove theorems using mathematical induction.

Part III: Sets

• Understand the definitions of sets, elements, subsets, and proper subsets, as well as symbolic notation for these terms.
• Understand both roster and set builder notation, and be able to determine whether a given set is well defined.
• Understand the definition of set equality, the cardinality of a set, the universal set, the empty set, singleton sets, the power set of a set, and know how to form the Cartesian Product of two sets.
• Understand the definition of the basic set operations.
• Be able to determine which elements are in a set resulting from multiple set operations. Also be able to draw a Venn diagram representing such a set.
• Know the basic set identities (set equalities) and be able to prove them using membership tables, or two-column proofs.

Part IV: Functions, Sequences, Series, and Recursion

• Know the definition of a function, including the definition of the domain, co-domain, and range of a function.
• Understand and be able to apply the definitions of image and preimage (for both individual elements and sets of elements).
• Know the definition of one-to-one and onto, and be able to determine whether or not a function satisfies these definitions.
• Understand the definition of function addition, subtraction, multiplication, division and composition and be able to prove facts about the results of applying these operations to a pair of functions.
• Understand the definition of an inverse function and know the properties that a function must satisfy to have an inverse.
• Understand the definition of the graph of a function, and be able to draw the graph of a given function.
• Know the definition of the floor function and the ceiling function and be able to apply this definition to investigate the graphs and/or properties of related functions.
• Know the definition of a sequence along with standard sequence notation.
• Understand both explicitly and recursively defined sequences and be able to use them to write out the first few terms in a sequence.
• Be able to determine whether or not a sequence satisfies a given recurrence relation.
• Be able to find a recurrence relation that represents an explicitly defined sequence.
• Be able to find en explicitly defined sequence that is a solution to a given recurrence relation.
• Understand summation notation and be able to compute both single and double summations.

Part V: Graphs

• Understand the definitions of the different types of graph models and know how various types of graphs can be used to model real world applications.
• Understand the basic graph terminology and main results like the Handshaking Theorem.
• Know the main classes of simple graphs like K_n, C_n, W_n, Q_n, and $K_{m,n}$.
• Understand the definition of a bipartite graph and be able to prove whether or not a given graph is bipartite. Also be able to use bipartite graphs to solve matching problems.
• Understand how to form subgraphs of a graph, the union of two graphs, and how to find the complement of a simple graph.
• Understand how to represent both directed and undirected graphs using Adjacency Lists, Adjacency Matrices, and Incidence Matrices. Also know the definitions of a graph isomorphism and a graph invariant.
• Given a pair of graphs, be able to prove whether or not the given pair are isomorphic.
• Know the definitions of paths and circuits.
• Understand what it means for a graph to be connected, strongly connected, and weakly connected. Also be able to find the connected components (or strongly connected components) of a given graph.
• Understand the path length is a graph invariant, and be able to use this fact in graph isomorphism proofs.
• Understand the definition of Euler Paths, Euler Circuits, Hamilton Paths, and Hamilton Circuits. Also understand how these are used in graph models.
• Given a specific (di)graph, be able to determine whether or not the graph has an Euler Path, an Euler Circuit, a Hamilton Path or a Hamilton Circuit by using key theorems. Also be able to find and label them in a graph when they exist.
• Understand the definition of and be able to identify cut vertices and bridge edges in a given graph.
• Understand Dijkstra's algorithm and be able to use it to find the shortest path between two vertices in a weighted graph.
• Understand the Traveling Salesman problem and the issues which make it difficult to solve efficiently.
• Understand how to use a brute force algorithm in order to solve the TSP.
• Understand some of the practical applications of the shortest path problem and the TSP.
• Be able to find the vertex or edge chromatic number of a given graph.
• Be able to use graph vertex colorings to solve applications problems (like scheduling and radio frequency assignments).

Part VI: Trees

• Understand the definition of a tree, a forest, a simple circuit. Know the main theorems about both trees and rooted trees.
• Understand the definition of a rooted tree, parent vertices, children, descendants, ancestors, internal vertices, leaves, siblings, an m-ary tree, a full m-ary tree, and the level of a vertex in a rooted tree.
• Understand the proof of the theorem that states that a tree with n vertices has $n-1$ edges.
• Understand the proof of the theorem that states that there are at most m^h leaves in m-ary tree of height h.
• Understand how to carry out a "tournament sort"
• Be able to construct and use a decision tree to solve an application problem (e.g. a weighing problem).
• Understand the definition of a binary search tree and be able to construct a binary search tree for a given list of elements from a totally ordered set.
• Understand how to find the universal address system for a given rooted m-ary tree.
• Be able to find the preorder, inorder, and postorder traversal for a given m-ary tree.
• Be able to draw the tree of a computation in prenix, infix, or postfix form, and be able to evaluate the computation.
• Know the definition of a spanning tree for a connected graph and be able to find a spanning tree for a graph using either a depth first search or a breadth first search.
• Understand how to use Prim's Algorithm and Kruskal's Algorithm to find a minimum spanning tree for a given connected weighted graph.