CS330 – Discrete Structures **Last Updated -** 02/19/02

Course Manager - Dr. Sanjiv Kapoor, Professor

3 credit hours; required for CS & CPE (or MATH230); 150 min. lecture each week

Current Catalog Description - Introduction to the use of formal mathematical structures to represent problems and computational processes. Topics covered include Boolean algebra, first-order logic, recursive structures, graphs, and abstract language models. Prerequisite: CS 106 or CS 200. (3-0-3)

Textbook

- Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, McGraw-Hill, 4th Edition, ©1999, ISBN-0072899050

References - other textbooks or materials

- none

Course Goals - Students should be able to:

- Illustrate by examples the basic terminology of functions, relations, and sets and demonstrate knowledge of their associated operations.
- Demonstrate in practical applications the use of basic counting principles of permutations, combinations, inclusion/exclusion principle and the pigeonhole methodology.
- Calculate probabilities of events and expectations of random variables for problems arising from games of chance.
- Establish and solve recurrence relations that arise in counting problems including the problem of determining the time complexity of recursively defined algorithms.
- Model logic statements arising in algorithm correctness and real-life situations and manipulate them using the formal methods of propositional and predicate logic.
- Outline basic proofs for theorems using the techniques of - direct proofs, proof by counterexample, proof by contraposition, proof by contradiction, mathematical induction.
- Relate the ideas of mathematical induction to recursion and recursively defined structures.
- Illustrate by example basic terminology of graph theory and model problems in computer science using graphs and trees.
- Deduce properties that establish particular graphs as Trees, Planar, Eulerian, and Hamiltonion.
- Illustrate the application of trees and graphs to data structures.
- Explain the basic concepts modeling computation including formal machines, languages, finite automata, Turing machines

Prerequisites by Topic

- Experience writing programs in any computer language.
- Pre-calculus

Major Topics Covered in Course

1. Sets, Functions and relations - sets, set operations, functions, summations, growth of functions, equivalence relations, countable and uncountable sets, examples of algorithm analysis	4.5 hours
2. Counting Methods – permutations, combinations, discrete probability, pigeonhole principle	6 hours
3. Advanced counting – inclusion-exclusion, recurrence relations, methods of solving recurrences, examples from computer sciences	6 hours
4. Introductory Logic – propositional logic, predicate logic, proof methodologies, examples of algorithm correctness	6 hours
5. Partially Ordered sets - trees, boolean algebra, example of minimizing circuits	4.5 hours
6. Introduction to Graphs - trees , connectivity, eulerian traversals, minimum spanning tree, planarity, Euler's formula, matching	7.5 hours

7. Formal machines and languages-an introduction - automaton, grammars and turing machines	6 hours
8. Introduction to Algebraic Topics (OPTIONAL) – rings, groups, semi-groups.	1.5 hours
Exam #1, Exam #2	3 hours
Final Exam	-
	45 hours

Laboratory projects (specify number of weeks on each)

- none

Estimate CSAB Category Content in Credit Hours

	CORE	ADVANCED		CORE	ADVANCED
Data Structures	.5		Computer Organization and Architecture	.5	
Algorithms	2		Concepts of Programming Languages	0	
Software Design	0				

Oral and Written Communications - Every student is required to submit at least __0___ written reports (not including exams, tests, quizzes, or commented programs) of typically __0___ pages and to make __0___ oral presentations of typically __0___ minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

Social and Ethical Issues - Please list the topics that address the social and ethical implications of computing covered in all course sections. Estimate the class time spent on each topic. In what ways are the students in this course graded on their understanding of these topics (e.g., test questions, essays, oral presentations, and so forth)?

- none

Theoretical Foundations - Please list the types of theoretical material covered, and estimate the time devoted to such coverage in contact (lecture and lab) hours.

- The entire course is basically theoretical material, please see "Major Topics Covered in Course" above

Problem Analysis - Please describe the problem analysis experiences common to all course sections.

- Deriving recurrence relations from recursive algorithms
- Modeling logic statements

Solution Design - Please describe the design experiences common to all course sections.

- none

Other Course Information

- Additional Suggested Course Assignments
  - 4-6 homework assignments
  - 2 exams (75 minutes each)
  - 1 final exam (120 minutes)
- Planned Course Enhancements
  - New course description - Introduction to the use of formal mathematical structures to represent problems and computational processes. Topics covered include sets, functions and relations, counting methods, recursive structures, logic, partially ordered sets, graphs, formal machines and languages. Prerequisite: CS 106 or CS 200. (3-0-3) (Fall 2002)
  - Replacement of programming assignments with additional quizzes or homework assignments (Fall 2002)