SPRINGFIELD TECHNICAL COMMUNITY COLLEGE

ACADEMIC AFFAIRS

Course Number: MATH 376 Department: Mathematics

Course Title: Discrete Structures Semester: Spring Year: 2003

Objectives/Competencies

Course Objective	Competencies
1. Understand and employ the basics of LOGIC.	9. Define and construct propositions using unary and binary connectives. 10. Set up truth tables to establish tautologies, contingencies and contradictions. 11. Employ bit operators to manipulate string and numeric data. 12. Translate and symbolize universal and existential quantifiers.
2. Construct, define and apply FUNCTIONS.	1. Define basic set notation terminology and perform set operations. 2. Employ functions to examine relations between domains and ranges. 3. Determine the convergence and divergence of sequences and series. 4. Use exponential and factorial functions to determine function growth. 5. Use big-O notation to estimate the complexity of growth

Course Objective	Competencies
	functions.
3. Employ the concepts and theorems of NUMBER THEORY.	1. Use pseudocode to represent algorithms. 2. Perform search algorithms. 3. Determine the computational complexity of an algorithm. 4. Use prime and composite number in basic number theoretic algorithms. 5. Use the division algorithm. 6. Find the GCD and LCM. 7. Use the notation and operations of modular arithmetic. 8. Determine random number seeds and pseudorandomness. 9. Develop and analyze encryption models. 10. Perform binary operations. 11. Write computer code for Euclidean algorithm. 12. Solve problems using the Chinese Remainder Theorem. 13. Define basic matrix operations.
4. Apply MATHEMATICAL REASONING.	1. Define axioms and prove basic number theoretic theorems. 2. Use the rules of inference to determine the validity of arguments. 3. Recognize fallacies. 4. Use the direct proof and indirect to prove theorems. 5. Use mathematical induction to prove theorems for all *n*. 6. Demonstrate existence proofs. 7. Recognize the Halting problems and its consequences. 8. Use the well-ordering principle.

Course Objective	Competencies
	9. Set up recursion definitions. 10. Employ loops and iterations to solve recursion problems. 11. Determine program correctness.
5. Enumeric lists by applying COMBINATORICS.	1. State and use the sum rule and product rule for counting. 2. Use the inclusion/exclusion principles in counting. 3. Draw three diagrams for all possible outcomes. 4. Use Pascal's identity for counting problems. 5. State Vandermonde's Identity. 6. Expand binomials with the binomial theorem. 7. Solve discrete probability problems with combinations and permutations. 8. Do problems with conditional probability. 9. Find the expected value of an experiment. 10. Use Chebychev's inequality for appropriate discrete probability problems. 11. Compute the complexity of computations.
6. Examine and compare RELATIONS.	1. Set up mathematical models for compound interest, Fibonacci numbers and Tower of Hanoi problems. 2. Solve homogeneous linear recurrence relations. 3. Define binary relations. 4. Define and recognize reflexive, symmetric and transitive properties. 5. Define n-ary relations. 6. Set up and solve problems using directed graphs. 7. Define closures and paths in a given path problem.

Course Objective	Competencies
	8. Define qquivalence relations and partial orderings on a set.
7. Examine and solve problems with GRAPHS.	1. Define a simple graph. 2. Define a multigraph. 3. Set up graph models to represent real world problems. 4. Define a bipartite graph. 5. Define and utilize a local area network. 6. Set up an interconnected network with parallel processing. 7. Define a subgraph. 8. Use matrices to represent graphs. 9. Set up an adjoining matrix and an incidence matrix. 10. Define isomorphic. 11. Determine connectivity of a graph. 12. Find Euler paths of a given graph. 13. Find Hamilton paths of a given graph. 14. Define graph coloring.
8. Define and examine FINITE STATE MACHINES.	1. Define and use language and grammar parametrics. 2. Set up context-free and context-sensitive grammars. 3. Define a finite state machine with and without output. 4. Define a regular set. 5. Define the grammar of a regular set. 6. Define and examine the Turing Machine with its implications for computer output.