COURSE OUTLINE

Course Number	Course Title	Credits
COS 204	**Discrete Mathematical Structures**	**4**
Hours: Lecture/Lab/Other	**Co- or Pre-requisite**	**Implementation Semester & Year**
4 hours lecture	**Pre-requisite: MAT 151**	**Spring 2022**

Catalog description:

Discrete mathematics is primarily intended for computer science majors. The course covers a wide variety of topics serving as the mathematical framework for the design and analysis of algorithms. Topics include induction and recursion, relations, functions, sets, propositional logic, Boolean algebra, grammars, tree structures, permutations and combinations and finite state machines.

General Education Category:
Not GenEd

Course coordinator:

Meimei Gao, 609-570-3483, gaom@mccc.edu

Required texts & Other materials:
Rosen, Discrete Mathematics and Its Applications with CONNECT, 8th Edition, ISBN10: 1259731251, ISBN13: 9781259731259
CONNECT is the required software for this course. The e-text is included with CONNECT access. Students can register and access the materials through the Blackboard link.

Course Student Learning Outcomes (SLO):

Upon successful completion of this course the student will be able to:

1. Use mathematical tools of logic and induction [Supports ILG # 2, 11; PLO #1, 2, 3]
2. Describe and integrate definitions and theorems concerning sets, functions, and relations. [Supports ILG # 2, 11; PLO #1, 2, 3]
3. Create and understand a formal proof [Supports ILG # 2, 11; PLO #1, 2, 3]
4. Use graphs to represent relations and to describe a system [Supports ILG # 2, 11; PLO #1, 2, 3]

Course-specific Institutional Learning Goals (ILG):

Institutional Learning Goal 2. Mathematics. Students will use appropriate mathematical and statistical concepts and operations to interpret data and to solve problems.
Institutional Learning Goal 11. Critical Thinking: Students will use critical thinking skills understand, analyze, or apply information or solve problems.

Program Learning Outcomes for Computer Science AS (PLO)

1. Apply the fundamental concepts and techniques of computation, algorithms, and software design to a specific problem in a variety of applied fields;
2. Provide detailed specifications, analyze the problem, and design a solution that functions as desired, has satisfactory performance, is reliable and maintainable, and meets desired criteria;
3. Apply a firm understanding in areas of mathematics and science.

Units of study in detail – Unit Student Learning Outcomes:

Unit I **Logic and Proof [**Supports Course SLO #1, 3]

Learning Objectives

The student will be able to:

- Apply formal methods of symbolic propositional and predicate logic.
- Apply the rules and techniques needed for determining whether a given argument is valid.
- Use propositional logic to prove theorems.
- Use deduction and induction as methods of proof.

Unit II **Sets, Functions, Sequences, Sums and Matrices [**Supports Course SLO #2]

Learning Objectives

The student will be able to:

- Perform the common operations on sets such as union, intersection, and difference.
- Visualize set operations in terms of Venn diagrams.
- Use algebraic laws to manipulate and simplify expressions involving sets and operations on sets.
- Define a sequence with a recursive or explicit formula.
- Describe recursive algorithms such as factorial and the Fibonacci sequence.
- Compute the sum, product, and transpose of matrices.

Unit III **Counting [**Supports Course SLO #1, 2, 3]

Learning Objectives

The student will be able to:

- Use combinatorial analysis techniques to count the number of solutions.
- Compute the number of permutations of a set of objects.
- Compute the number of combinations of a set of objects.

Unit IV **Relations and Digraphs [**Supports Course SLO #2, 4]

Learning Objectives

The student will be able to:

- Describe relations and their properties.
- Represent relations with matrices and graphs.
- Define reflexive, irreflexive, symmetric, asymmetric, antisymmetric and transitive relations.
- Define an equivalence relation.

Unit V **Trees and Graphs [**Supports Course SLO #4]

Learning Objectives

The student will be able to:

- Traverse trees with preorder, inorder, and postorder algorithms.
- Represent a graph as adjacency matrix, adjacency list.
- Represent finite-state machines with state transition tables and digraphs.

Evaluation of student learning:

Specific methods for evaluating student progress through the course is up to the discretion of the instructor. Below is an example:

Assignments = 30% of the grade
Tests = 30% of the grade
Final Exam = 40% of the grade