Anekant Education Society's
Tuljaram Chaturchand College of Arts, Science and Commerce, Baramati

Autonomous

Course Structure for B.Sc. (Computer Science) Mathematics (2022 Pattern)

F. Y. B. Sc. (Computer Science) Mathematics

Semester	Course Code	Title of Course	No. of Credits	No. of Lectures
I	UCSMT111	Graph Theory	2	36
	UCSMT112	Matrix Algebra	2	36
	UCSMT113	Mathematics Practical based on UCSMT111 & UCSMT112	2	48
II	UCSMT121	Discrete Mathematics	2	36
	UCSMT122	Linear Algebra	2	36
	UCSMT123	Mathematics Practical based on UCSMT121 & UCSMT122	2	48

S. Y. B. Sc. (Computer Science) Mathematics

Semester	Course Code	Title of Course	No. of Credits	No. of Lectures
I	UCSMT231	Groups and Coding Theory	3	48
	UCSMT232	Numerical Techniques	3	48
	UCSMT233	Mathematics Practical Python Programming Language I	2	48
II	UCSMT241	Computational Geometry	3	48
	UCSMT242	Operation Research	3	48
	UCSMT243	Mathematics Practical Python Programming Language II	2	48

Equivalence of the Old Syllabus with New Syllabus:

Old Course		New Course	
CSMT1201	Discrete Mathematics	UCSMT121	Discrete Mathematics
CSMT1202	Calculus	UCSMT122	Linear Algebra
CSMT1203	Mathematics Practical based on CSMT1201 & CSMT1202	UCSMT123	Mathematics Practical based on UCSMT121 & UCSMT122

Choice Based Credit System Syllabus (2022 Pattern)

Class: F.Y.B.Sc.(Computer Science). (Sem II) **Subject:** Mathematics
Course: Discrete Mathematics **Course Code:** UCSMT121

A) Course Objectives:

1. Introduce concepts of mathematical logic for analyzing propositions and proving theorems.
2. To understand concepts of Lattices, Boolean algebra, Recurrence relation.
3. How to use and analyse recursive definitions.
4. Apply quantification and understand universal and existential quantifiers.
5. Represent relations using digraphs.
6. Apply the product rule, sum rule, and inclusion-exclusion principle in counting problems.
7. Understand boolean variables and functions.

B) Course Outcome:

1. Express logic sentences in terms of predicates, quantifiers and also evaluate Boolean functions and simplify expression using Boolean algebra.
2. Define and calculate transitive closure in relation to graphs
3. Apply quantifiers to create logical expressions.
4. Apply rules of inference, including direct and indirect methods.
5. Build and evaluate logical arguments using propositional logic.
6. Apply logic concepts to solve problems in computer science.
7. Analyze minterms, maxterms, disjunctive normal form, and conjunctive normal form.

TOPICS/CONTENTS

Unit 01: Logic (7 lectures)

1.1 Revision: Propositional Logic, Propositional Equivalences

1.2 Predicates and Quantifiers: Predicate, n-place Predicate or n-ary Predicate, Quantification and Quantifiers, Universal Quantifier, Existential Quantifier, Quantifiers with restricted domains, Logical Equivalences involving Quantifiers.

1.3 Rules of Inference: Argument in propositional Logic, Validity Argument (Direct and Indirect methods), Rules of Inference for Propositional Logic, Building Arguments.

Unit 02: Relation and Digraph (8 lectures)

2.1 Ordered pairs, Cartesian Product of sets

2.2 Relation, types of relation, equivalence relation, Partial Ordering relations.

2.3 Digraphs of relations ,matrix representation and composition of relations

2.4 Transitive Closure and Warshall’s Algorithm

Unit 03: Lattices and Boolean Algebra (6 lectures)

3.1 Lattices, Complemented Lattice, Bounded Lattice and Distributive Lattice.
3.2 Boolean Functions: Introduction, Boolean Variable, Boolean Function of degree n, Boolean identities, Definition of Boolean Algebra.
3.3 Representation of Boolean Functions: Minterm, Maxterm, Disjunctive normal form, Conjunctive normal form.

Unit 04: Counting Principles (7 lectures)

4.1 Cardinality of Set: Cardinality of finite Sets
4.2 Basics of Counting: The Product Rule, The Sum rule, The Inclusion-Exclusion Principle.
4.3 The Pigeonhole Principle: Statement, The Generalized Pigeonhole Principle, Its Applications.
4.4 Generalized Permutations and Combinations: Permutation and Combination with Repetitions, Permutation With Indistinguishable Objects.

Unit 05: Recurrence Relations (8 lectures)

5.1 Recurrence Relations: Introduction, Formation
5.2 Linear Recurrence Relations with constant coefficients
5.3 Homogeneous solutions.
5.4 Particular solutions
5.5 Total solutions

Text Book : Kenneth Rosen, Discrete Mathematics and its applications, McGraw Hill Education Pvt. Ltd. (7^{th} Edition).

Unit 1: Section 1.1to 1.5
Unit 4: Section 6.1 to 6.6
Unit 5: Section 8.2

Text Book : Bernard Kolman, Robert Busby, Sharon Culter Ross, Nadeem-ur-Rehman, Discrete Mathematics Structure, Pearson Education, 5^{th} Edition.

Unit 2: Section 4.2, 4.4, 4.5, 4.8
Unit 3: Section 7.3 to 7.6

Reference Books:

1. C. L. Liu., Elements of Discrete Mathematics, Tata McGraw Hill.

Mapping of Program Outcomes with Course Outcomes

Weightage: 1= weak or low relation, 2= moderate or partial relation, 3= strong or direct relation

	Programme Outcomes (POs)						
Course Outcomes	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7
CO 1	3					2	1
CO 2		3	3				3
CO 3			3				
CO 4		2	1				3
CO 5	1					1	
CO 6		3				1	2
CO 7	3		3			2	1

Justification for the mapping

PO1: Computer Knowledge

CO1: Expressing logic sentences in terms of predicates, quantifiers, and evaluating Boolean functions allows for precise representation and manipulation of logical statements, facilitating effective problem-solving and algorithmic design in computer science through the application of formal logic and Boolean algebra.
CO5: Enhances computational problem-solving skills by enabling systematic analysis and validation of logical structures in software development through the application of propositional logic.
CO7: Analyzing minterms, maxterms, disjunctive normal form (DNF), and conjunctive normal form (CNF) is essential in digital logic design for Boolean algebra manipulation, enabling efficient representation and simplification of logical expressions used in designing digital circuits.

PO2: Design / Development of solution

CO2: Transitive closure in graph theory is essential for determining the reachability between all pairs of vertices, aiding in the comprehensive analysis of connectivity within a graph during the design and development of a solution.
CO4: Applying rules of inference facilitates sound logical reasoning, enhancing the design and development of solutions by ensuring coherent and valid progression from premises to conclusions through direct and indirect methods.
CO6: Applying logic concepts in computer science facilitates the design and development of efficient solutions by ensuring systematic problem-solving, effective decision-making, and optimized algorithmic approaches.

PO3: Modern tool usage

CO2: Transitive closure in graph theory facilitates the determination of all possible paths between vertices, essential for efficient graph analysis and optimization in modern tools.
CO3: Applying quantifiers in modern tools enhances precision and efficiency by expressing logical conditions, enabling more robust and expressive representation of relationships within

data and programming environments.
CO4: Applying rules of inference, including direct and indirect methods, enhances logical reasoning and problem-solving abilities, facilitating effective decision-making and critical thinking in modern tool usage.
CO7: Analyzing minterms, maxterms, disjunctive normal form (DNF), and conjunctive normal form (CNF) is essential in modern tool usage for Boolean algebra simplification and optimization, facilitating efficient digital circuit design and logical expression representation.

PO6: Individual and Team work

CO1: Expressing logic sentences in terms of predicates, quantifiers, and evaluating Boolean functions using Boolean algebra fosters precise communication and efficient problem-solving, promoting both individual cognitive clarity and collaborative teamwork in logical reasoning and decision-making contexts.

CO5: Building and evaluating logical arguments using propositional logic enhances both individual and team problem-solving skills by fostering precise reasoning and collaborative decision-making based on a systematic and structured approach.
CO6: Applying logic concepts in computer science enhances problem-solving by systematically analyzing and structuring information, fostering precision and collaboration in both individual and team work.
CO7: Analyzing minterms, maxterms, disjunctive normal form (DNF), and conjunctive normal form (CNF) facilitates individual and team work by providing a systematic approach to Boolean algebra, aiding in logical circuit design, optimization, and simplification for efficient collaboration in digital system development.

PO7: Innovation, employability and Entrepreneurial skills
CO1: Expressing logic sentences in terms of predicates, quantifiers, and evaluating Boolean functions using Boolean algebra fosters critical thinking and problem-solving abilities, essential for innovation, employability, and entrepreneurial skills by developing a strong foundation in formal reasoning and computational decision-making.
CO2: Transitive closure in graphs enhances innovation, employability, and entrepreneurial skills by identifying and facilitating the exploration of indirect relationships, fostering a holistic understanding of interconnected concepts and opportunities within the dynamic landscape of innovation and entrepreneurship.

CO4: Applying rules of inference in innovation, employability, and entrepreneurial skills enhances logical reasoning and problem-solving, fostering critical thinking essential for effective decision-making and creative problem-solving in dynamic professional environments.

CO6: Applying logic concepts in computer science enhances innovation by fostering critical thinking, ensures employability through problem-solving proficiency, and cultivates entrepreneurial skills by enabling the development of robust and efficient solutions.

CO7: Analyzing minterms, maxterms, disjunctive normal form, and conjunctive normal form enhances innovation and entrepreneurial skills by fostering logical thinking, problem-solving, and structured representation, thereby empowering individuals with employability skills crucial for navigating complex business challenges.

Choice Based Credit System Syllabus (2022 Pattern)

Class: F.Y.B.Sc.(Computer Science). (Sem II) **Subject:** Mathematics
Course: Linear Algebra **Course Code:** UCSMT122

A) Course Objectives:

1. To understand properties and operations on System of Linear Equations.
2. To understand basic concepts of Determinants.
3. Understanding of how to translate a linear equation into a matrix.
4. Identify and analyze properties of vector spaces.
5. Understand the concepts of basis and dimensions in vector spaces.
6. Construct orthogonal bases using the Gram-Schmidt process.
7. Understand the use of eigenvalues and eigenvectors with complex matrices.

B) Course Outcome:

1. Student will apply the principles of Euclidean n-space, including geometric interpretations and applications.
2. Recognize and apply the concept of linear independence in vector spaces.
3. Analyze and understand the row space, column space, and null space of matrices, and relate them to linear transformations.
4. Apply techniques for determining eigenvalues and eigenvectors of matrices.
5. Apply the concept of inverse linear transformations and recognize their importance in various applications.
6. Analyze and calculate the kernel and range of linear transformations.
7. Apply inner products to analyze angles and Orthogonality in inner product spaces.

TOPICS/CONTENTS

Unit 01: Vector Spaces (10 lectures)

1.1 Vector Spaces and Subspaces
1.2 Solving $Ax = 0$ and $Ax = b$
1.3 Linearly Independence, Basis and Dimensions
1.4 The Four Fundamental subspaces
1.5 Graphs and Networks
1.6 Linear Transformations

Unit 02: Orthogonality (06 lectures)

2.1 Orthogonal Vectors and subspaces
2.2 Cosines and projections onto Lines
2.3 Projections and least squares
2.4 Orthogonal Bases and Gram-Schmidt
2.5 The Fast Fourier Transform

Unit 03: Eigen Values and Eigen Vectors (10 lectures)

3.1 Introduction
3.2 Diagonalization of a Matrix
3.3 Difference Equations and Powers A^{k}
3.4 Differential Equations and e^{At}
3.5 Complex Matrices
3.6 Similarity Transformations

Unit 04: Symmetric Matrices (04 lectures)

4.1 Diagonalization of Symmetric Matrices
4.2 Quadratic Forms

Unit 05: The Geometry of Vector Spaces (06 lectures)

5.1 Affine Combinations
5.2 Affine Independence
5.3 Convex Combinations

Text Book: Gilbert Strang, Linear Algebra and its applications (4^{th} Edition).

Unit 1: Section 2.1 to 2.6
Unit 2: Section 3.1 to 3.5
Unit 3: Section 5.1 to 5.6

Text Book: David C. Lay, Linear Algebra and its Applications, MacDonald Pearson Publication Fourth Edition.

Unit 4: Section 7.1, 7.2
Unit 5: Section 8.1 to 8.3

Reference Books:

1. Howard Anton and others , Elementary Linear Algebra with supplemental Applications ,Wiley Student Edition.
2. KantiBhushan Datta , Matrix and Linear Algebra (aided with MATLAB) , EasternEconomic Edition.
3. Devi Prasad , Elementary Linear Algebra , Narosa , Third Edition.

Mapping of Program Outcomes with Course Outcomes

Weightage: 1= weak or low relation, 2= moderate or partial relation, 3= strong or direct relation

	Programme Outcomes (POs)						
Course Outcomes	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7
CO 1			2			3	3
CO 2			1			3	
CO 3			3			2	2
CO 4		3	2			3	
CO 5		2				3	1
CO 6		2	3			2	
CO 7		1	2				3

Justification for the mapping

PO2: Design / Development of solution

CO4: Applying techniques for determining eigenvalues and eigenvectors in the design/development of a solution enables efficient linear transformation analysis, facilitating optimal system stability and performance optimization through eigen-based feature extraction and dimensionality reduction.
CO5: Applying the concept of inverse linear transformations is crucial in design and development, allowing for the effective solution of inverse problems, system calibration, and the reconstruction of original data, enhancing versatility in diverse applications such as image processing, signal recovery, and optimization.
CO6: Analyzing and calculating the kernel and range of linear transformations in the design/development of a solution provides insights into solution space, aiding in system optimization and constraint identification for enhanced problem-solving efficiency.
CO7: Applying inner products to analyze angles and orthogonality in inner product spaces during the design/development of a solution ensures precise vector alignment and efficient orthogonal decomposition, enhancing numerical stability and geometric understanding for optimal problem-solving strategies.

PO3: Modern tool usage

CO1: Student will apply the principles of Euclidean n-space, incorporating geometric interpretations and modern tool applications, fostering a comprehensive understanding that underpins practical problem-solving and analytical skills in diverse fields.
CO2: Student will recognize and apply the concept of linear independence in vector spaces within modern tools, enabling them to enhance computational efficiency, numerical stability, and analytical precision in various applications.
CO3: Student will analyze and understand the row space, column space, and null space of matrices, and relate them to linear transformations in modern tool uses, fostering a deep comprehension of linear algebra concepts essential for effective problem-solving and system analysis.
CO4: Student will apply techniques for determining eigenvalues and eigenvectors of matrices in modern tool uses, enhancing their ability to analyze dynamic systems, extract key features, and optimize solutions in diverse computational applications.

CO6: Student will analyze and calculate the kernel and range of linear transformations in modern tool uses, equipping them with essential skills to optimize system solutions, identify constraints, and enhance computational efficiency across a variety of applications.
CO7: Student will apply inner products to analyze angles and orthogonality in inner product spaces in modern tool uses, providing a foundation for precise vector manipulation, optimization, and geometric understanding in diverse computational applications.

PO6: Individual and Team work

CO1: Student will apply the principles of Euclidean n-space, integrating geometric interpretations and applications, fostering individual and teamwork skills essential for collaborative problem-solving and innovative solutions across diverse domains.

CO2: Student will recognize and apply the concept of linear independence in vector spaces, cultivating essential skills for both individual and teamwork scenarios, promoting collaborative problem-solving and effective communication in diverse applications.
CO3: Student will analyze and understand the row space, column space, and null space of matrices, and relate them to linear transformations, fostering a foundation for individual and teamwork problem-solving, system analysis, and efficient communication in diverse applications.
CO4: Student will apply techniques for determining eigen values and eigenvectors of matrices, enhancing individual and teamwork problem-solving capabilities, and enabling efficient analysis and optimization of systems across diverse collaborative applications.
CO5: Student will apply the concept of inverse linear transformations, recognizing their significance in various applications, thereby fostering individual and teamwork problem-solving skills essential for system calibration, data recovery, and optimization in collaborative projects.
CO6: Student will analyze and calculate the kernel and range of linear transformations, fostering individual and teamwork skills crucial for efficient problem-solving, system optimization, and collaborative decision-making across a range of applications.
PO7: Innovation, employability and Entrepreneurial skills
CO1: Student will apply the principles of Euclidean n-space, incorporating geometric interpretations and applications, to develop innovation, employability, and entrepreneurial skills, fostering a practical understanding essential for creative problem-solving in diverse contexts.

CO3: Student will analyze and understand the row space, column space, and null space of matrices, relating them to linear transformations, to enhance innovation, employability, and entrepreneurial skills, enabling practical problem-solving and strategic thinking in various professional contexts.

CO5: Student will apply the concept of inverse linear transformations, recognizing their importance in various applications, to cultivate innovation, employability, and entrepreneurial skills, empowering them with a versatile problem-solving toolkit for dynamic and creative solutions in diverse professional settings.

CO7: Student applying inner products to analyze angles and orthogonality in inner product spaces enhance innovation, employability, and entrepreneurial skills, fostering a practical foundation for precise measurements, optimization, and creative problem-solving in diverse professional contexts.

Choice Based Credit System Syllabus (2022 Pattern)

Class: F.Y.B.Sc.(Computer Science). (Sem II) **Subject:** Mathematics
Course: Mathematics Practical based on UCSMT121 & UCSMT122 **Course Code:** UCSMT123

A) Course Objectives:

1. Problem solving ability and understanding applications of discrete mathematics.
2. Solve system of linear equation using multiple methods.
3. To build the necessary skill set and analytical abilities for developing computer based solutions using mathematical concepts.
4. Apply logical reasoning to solve mathematical problems.
5. Utilize Maxima software to solve logic and lattice-related problems.
6. Enhance computational skills in linear algebra.
7. Apply Maxima software to solve problems in counting principles and recurrence relations.

B) Course Outcome:

1. Lead students to apply these mathematical concepts in the study of computer science
2. Students are able to apply logical reasoning to solve a variety of problems.
3. Understand and apply properties of lattices and relations to problem-solving.
4. Demonstrate proficiency in using computational tools for discrete mathematics.
5. Gain practical experience in using computational tools for solving discrete mathematics problems.
6. Utilize Maxima software proficiently to solve problems in vector spaces, eigenvalues, and eigenvectors.
7. Apply Maxima software effectively to solve problems related to orthogonality, symmetric matrices, and geometric interpretations.

Title of Experiments:

Discrete Mathematics:

1. Problems on Logic.
2. Problems on Lattices and Relation.
3. Problems on Counting Principles.
4. Problems on Recurrence Relation.
5. Problems on Logic and Lattices using maxima software.
6. Problems on Counting Principles and Recurrence Relations using maxima software.

Linear Algebra:

1. Problems on Vector Spaces
2. Problems on Eigen Values and Eigen Vectors
3. Problems on Orthogonality and Symmetric Matrices
4. Problems on The Geometry of vector spaces
5. Problems on Vector Spaces, Eigen Values and Eigen Vectors on Maxima Software.
6. Problems on Orthogonality, Symmetric Matrices and Geometry of Vector Spaces using Maxima Software.

Mapping of Program Outcomes with Course Outcomes

Weightage: 1= weak or low relation, 2= moderate or partial relation, 3= strong or direct relation

	Programme Outcomes (POs)						
Course Outcomes	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7
CO 1			2			3	1
CO 2		2					
CO 3		1	1			2	
CO 4	2		2			1	1
CO 5		3				3	
CO 6	2						2
CO 7	1		2			3	1

Justification for the mapping

PO1: Computer Knowledge

CO4: Proficiency in computational tools for discrete mathematics enhances problem-solving capabilities, algorithmic design, and logical reasoning skills essential for effective problem-solving in computer science.
CO6: Maxima software mastery enables efficient resolution of complex problems in vector spaces, eigenvalues, and eigenvectors, enhancing computational proficiency in mathematical applications.
CO7: Maxima software excels in solving problems related to orthogonality, symmetric matrices, and geometric interpretations due to its robust symbolic computation capabilities and extensive mathematical functions, providing efficient and accurate solutions for complex algebraic and geometric computations.

PO2: Design / Development of solution

CO2: Logical reasoning enhances students' problem-solving skills in design and development, fostering their ability to create effective and innovative solutions.
CO3: Utilizing lattice and relation properties enhances problem-solving by providing a structured framework that fosters systematic analysis and effective organization of information in the design and development of solutions.
CO5: Enhance problem-solving proficiency by applying computational tools to address discrete mathematics challenges, fostering practical skills crucial for solution design and development.

PO3: Modern tool usage

CO1: Fostering the application of mathematical concepts in computer science equips students with essential problem-solving skills and enhances their proficiency in modern tool usage, crucial for addressing complex computational challenges.
CO3: Understanding and applying properties of lattices and relations enhances problem-solving through modern tools by providing a structured framework for organizing and analyzing complex relationships, facilitating more efficient and systematic computational approaches.
CO4: Mastering computational tools in discrete mathematics enhances problem-solving

efficiency, fosters algorithmic thinking, and facilitates the seamless application of mathematical concepts to real-world problems in modern tool usage.
CO7: Maxima software facilitates efficient solutions to problems in orthogonality, symmetric matrices, and geometric interpretations through its robust symbolic computation capabilities and comprehensive mathematical functions, enhancing modern tool usage in diverse mathematical applications.

PO6: Individual and Team work

CO1: Applying mathematical concepts in computer science cultivates problem-solving skills, enhances analytical thinking, and promotes collaboration, essential for both individual and team success in the dynamic field of computer science.

CO3: Understanding properties of lattices and relations enhances problem-solving in both individual and team work by providing a structured framework to analyze and model complex relationships, facilitating more systematic and collaborative approaches to problem-solving.
CO4: Mastering computational tools in discrete mathematics enhances both individual and team effectiveness by enabling efficient problem-solving, analysis, and collaborative exploration of mathematical concepts in diverse applications.
CO5: Enhances problem-solving proficiency in discrete mathematics through hands-on application of computational tools, fostering both individual and collaborative skill development.
CO7: Utilizing Maxima software enhances efficiency in addressing problems of orthogonality, symmetric matrices, and geometric interpretations, fostering both individual and team productivity through its powerful symbolic computation capabilities and collaborative features.

PO7: Innovation, employability and Entrepreneurial skills
CO1: Empowering students to apply mathematical concepts in computer science cultivates innovation, enhances employability, and fosters entrepreneurial skills by providing a robust foundation for problem-solving and critical thinking in the rapidly evolving technological landscape.
CO4: Mastering computational tools for discrete mathematics enhances problem-solving acumen, fostering innovation by providing a robust foundation for algorithmic thinking, a critical skill for employability and entrepreneurial success in today's technology-driven landscape.

CO6: Maxima software proficiency enhances problem-solving capabilities in vector spaces, eigenvalues, and eigenvectors, fostering innovation, employability, and entrepreneurial skills through advanced mathematical modeling and analysis.

CO7: Applying Maxima software enhances innovation, employability, and entrepreneurial skills by efficiently tackling problems in orthogonality, symmetric matrices, and geometric interpretations through its powerful symbolic computation capabilities, fostering a deeper understanding and facilitating creative solutions in diverse fields.

****