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

(Autonomous)

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

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

Semester	Paper Code	Title of Paper	No. of Credits
I	CSMT1101	Graph Theory	2
	CSMT1102	Algebra	2
	CSMT1103	Mathematics Practical based on CSMT1101 & CSMT1102	2
II	CSMT1201	Discrete Mathematics	2
	CSMT1202	Calculus	2
	CSMT1203	Mathematics Practical based on CSMT1201 & CSMT1202	2

Choice Based Credit System Syllabus (2019 Pattern)

Class: F.Y.B.Sc. (Comp. Sci.) (Sem I) **Subject:** Mathematics
Course: Graph Theory **Course Code:**CSMT1101

A) Course Objectives:

1. To introduce graphs, their types and properties
2. To understand applications of graph theory in Computer science
3. To build the necessary skillset and analytical abilities for developing computer based solutions using mathematical concepts
4. Understand additional definitions related to graph theory and apply them in problem-solving.
5. Identify and analyse subgraphs within a larger graph structure.
6. Apply algorithms to solve shortest path problems in a given graph.
7. Apply graph theory concepts to solve problems relevant to computer science.

B) Course Outcome:

1. Understanding of algorithms and applications to computer science.
2. Calculate and interpret vertex degrees in different graph structures, showcasing proficiency in graph analysis.
3. Develop a comprehensive understanding of various definitions related to graph theory and apply them to analyze and solve problems.
4. Identify and analyze bridges within a graph, showcasing the ability to recognize critical components in network structures.
5. Apply algorithms to solve the Travelling Salesman Problem, demonstrating proficiency in solving optimization problems in graph theory.
6. Understand and apply concepts of flow and cuts in directed graphs, demonstrating proficiency in analyzing network flows.
7. Comprehend the matrix representation of graphs and utilize it for solving problems related to graph structures.

TOPICS/CONTENTS:

Unit 01: An Introduction to Graphs (12 lectures)

1.1 Definition of a Graph.
1.2 Graphs as a model
1.3 More definitions
1.4 Vertex Degrees
1.5 Subgraphs
1.6 Paths and Cycles
1.7 The Matrix Representation of Graphs
1.8 Fusion

Unit 02: Trees and Connectivity (12 lectures)

2.1 Definitions and Simple Properties
2.2 Bridges
2.3 Spanning Trees
2.4 Connector Problems
2.5 Shortest Path Problems
2.6 Cut Vertices and Connectivity

Unit 03: Euler Tours and Hamiltonian Cycles (7 lectures)

3.1 Euler Tours
3.2 The Chinese Postman Problem
3.3 Hamiltonian Graphs
3.4 Travelling Salesman Problem

Unit 04: Directed Graphs and Networks (5 lectures)

4.1 Definitions
4.2 Indegree and Outdegree
4.3 Flow and Cuts
4.4 Applications to Computer Science

Text Book:

John Clark and Derek Holtan , A First Look at Graph Theory, Allied Publishers
(Chapters: 1, 2, 3, 7 and 8)

Reference Books:

1. Kenneth Rosen, Discrete Mathematics and It's Applications, Tata McGraw Hill.
2. Narsingh Deo, Graph Theory with Application to Computer Science and Engineerng, Prentice Hall.

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	2
CO 2	3		2				
CO 3	2	3				1	3
CO 4	3		1				1
CO 5		3	2				
CO 6	3		1				3
CO 7	3					3	1

Justification for the mapping

PO1: Computer Knowledge

CO1: Understanding algorithms equips students with problem-solving prowess and strategic thinking, pivotal for unraveling complexities in computer science applications and fostering a profound grasp of the discipline.
CO2: Proficiently calculating and interpreting vertex degrees in diverse graph structures demonstrates students' mastery in graph analysis, a critical skill for solving real-world problems in computer science.
CO3: Developing a comprehensive understanding of graph theory definitions enables students to apply analytical skills for problem-solving in diverse computer science scenarios.
CO4: Proficiently identifying and analyzing bridges within a graph demonstrates students' capacity to discern crucial components in network structures, enhancing their expertise in computer science.
CO6: Mastering the concepts of flow and cuts in directed graphs showcases students' proficiency in analyzing network flows, a critical skill for solving complex problems in computer science.
CO7: Acquiring the ability to comprehend matrix representation of graphs empowers students to effectively utilize this powerful tool for problem-solving in diverse graph structures within the realm of computer science.

PO2: Design / Development of solution

CO1: Understanding algorithms is crucial for students in the design and development of solutions in computer science, enabling efficient problem-solving, optimized coding practices, and the creation of innovative and effective software solutions.

CO3: Developing a comprehensive understanding of graph theory definitions equips students to analyze and solve problems in the design and development of solutions, fostering effective representation and manipulation of complex relationships in various computational scenarios.
CO5: Applying algorithms to solve the Travelling Salesman Problem showcases students' proficiency in addressing optimization challenges within graph theory, a critical skill for designing and developing efficient solutions in various computational contexts.

PO3: Modern tool usage

CO2: Calculating and interpreting vertex degrees in various graph structures demonstrates students' proficiency in graph analysis, a crucial skill in modern tool usage for understanding and manipulating complex networks and relationships.
CO4:Identifying and analysing bridges within a graph demonstrates the student's ability to recognize critical components in network structures, showcasing essential skills for effective analysis and optimization in modern tool usage.
CO5:Applying algorithms to solve the Travelling Salesman Problem showcases the student's proficiency in utilizing modern tools for solving complex optimization problems in graph theory, a valuable skill for algorithmic analysis and practical problem-solving in diverse applications.
CO6: Understanding and applying concepts of flow and cuts in directed graphs demonstrates the student's proficiency in modern tool usage, showcasing their ability to analyze network flows effectively—a critical skill for optimizing transportation, communication, and resource allocation in various applications.

PO6: Individual and Team work

CO1: Understanding algorithms is essential for effective individual and team work in computer science, fostering collaborative problem-solving, streamlined workflows, and innovative solutions.
CO3:Developing a comprehensive understanding of graph theory definitions and their application fosters effective problem analysis and solving, enhancing individual and team capabilities in tackling diverse challenges collaboratively.
CO7:Comprehending matrix representation of graphs and utilizing it for problem-solving enhances individual and team capabilities, enabling collaborative analysis and efficient solutions in diverse graph-related challenges.

PO7: Innovation, employability and Entrepreneurial skills
CO1: Understanding algorithms in computer science fosters innovation, employability, and entrepreneurial skills by equipping individuals with the problem-solving and computational thinking necessary to drive technological advancements and succeed in dynamic, competitive professional landscapes.

CO3: Developing a comprehensive understanding of graph theory definitions and applying them enhances innovation, employability, and entrepreneurial skills, enabling individuals to address complex challenges and create solutions with a strategic and inventive mindset.

CO4:Identifying and analyzing bridges within a graph demonstrates the ability to recognize critical components in network structures, fostering innovation, employability, and entrepreneurial skills through a nuanced understanding of complex relationships and system optimization.

CO6: Identifying and analyzing bridges within a graph showcases the ability to recognize critical components in network structures, enhancing innovation, employability, and entrepreneurial skills by enabling strategic optimization and problem-solving in complex systems.

CO7:Comprehending matrix representation of graphs and utilizing it for problem-solving enhances innovation, employability, and entrepreneurial skills by providing a powerful analytical tool for optimizing solutions and fostering strategic thinking in diverse graph-related challenges.

Choice Based Credit System Syllabus (2019 Pattern)

Class:F.Y.B.Sc.(Comp. Sci.) (Sem I) **Subject:** Mathematics
Course: Algebra **Course Code:**CSMT1102

A) Course Objectives:

1. To understand properties and operations on sets and functions
2. To understand basic concepts of groups, integers, matrices
3. Define and analyse the Cartesian product of sets, demonstrating proficiency in set operations.
4. Understand the definition of a function and its components, including domain, co-domain, and range.
5. Define groups and analyse their properties, including subgroups, finite and infinite groups, and cyclic groups.
6. Understand congruence relations and their properties, including residue classes and examples.
7. Analyse applications of matrices and systems of linear equations to computer science.

B) Course Outcome:

1. Improves problem solving ability and understanding of differential algebraic structures in Mathematics.
2. Define functions and comprehend related terms such as domain, co-domain, and range.
3. Analyze composite functions and understand the concept of invertible functions.
4. Understand and apply the concepts of monoids and semigroups, providing relevant examples.
5. Apply the principles of mathematical induction, including the first and second principles, with examples.
6. Understand concepts related to relatively prime integers, Euclid's lemma, and its generalization.
7. Solve systems of linear equations using the Gauss elimination method, Gauss-Jordan elimination method, and L.U. decomposition method.

TOPICS/CONTENTS:

Unit 01: Set and Functions (5 lectures)

1.1 Definition of set, operations on sets, power set, Cartesian product of sets.

1.2 Definition of Function, Domain, Co-domain and the range of function, Injective, surjective and bijective functions, Composite function, invertible function.

Unit 02: Binary Operations and Groups (13 lectures)

2.1 Definition of binary operation, examples, properties of binary operation.

2.2 Definition of Monoid, semi group, examples.

2.3 Definition of Group and examples, subgroups, finite and infinite groups, cyclic groups.

2.4 Applications to Computer Science.

Unit 03: Integers (9 lectures)

3.1 Well ordering principle.

3.2 First and Second Principle of Mathematical Induction ,Examples.

3.3 Division Algorithm (Without Proof)

3.4 Divisibility and its Properties, prime numbers.

3.5 Definition G.C.D and L.C.M , Expressing G.C.D. of two integers as a linear combination of the two integers.

3.6 Euclidean Algorithm (Without Proof)

3.7 Relatively prime integers , Euclid Lemma and its generalization.

3.8 Congruence relations and its properties, Residue Classes: Definition, Examples, addition and multiplication modulo n and composition tables.

3.9 Euler's and Fermat's Theorems.(Without Proof). Examples.

3.10 Applications to Computer Science.

Unit 04: Matrices and System of linear Equations (9 lectures)

4.1 Revision: Elementary operations on matrices.

4.2 Echelon form of matrix.

4.3 System of linear Equations:

Gauss Elimination Method,

Gauss Jordan Elimination Method,

L.U. Decompositions Method.

4.4 Rank of matrix, Row rank, Column rank.

4.5 Applications to Computer Science.

Text Books: Unit 01: Set & Functions (Kenneth Rosen, Discrete Mathematics and It's Applications, Tata McGraw Hill). **Section : 2.1 to 2.3**

Unit 02: Binary Operations and Groups (J. B. Fraleigh, A. First Course in Abstract. Algebra, Third Ed., Narosa, New Delhi 1990).**Chapter 1 .Section: 1,2,4,5,6.**

Unit 03: Integers (David M. Burton, Elementary Number Theory**) Section: 1.1, 2.1 to 2.4, 4.2,5.2,7.2, 7.3**

Unit 04: Matrices and System of linear Equations Kwak & Hong, Linear Algebra**)**

Section: 1.1 to 1.4, 1.6 to 1.9. .

Reference Books:

1. 1. Discrete Mathematics Structure- Bernard Kolman, Robert Busby, Sharon Culter Ross, Nadeem-ur-Rehman , Pearson Education, 5th Edition.
2. Elements of Discrete Mathematics – C. L. Liu, Tata McGraw Hill.
3. J. B. Fraleigh, A. First Course in Abstract Algebra, 7th Edition,Pearson
4. H. Anton, C. Rorres, Elementary linear algebra with applications, Wiley 7th Edition,1994.

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			3	
CO 2	3						2
CO 3	3		3				2
CO 4	3	1			1	3	3
CO 5	3		2				
CO 6	3	1					2
CO 7	3	2				2	2

Justification for the mapping

PO1: Computer Knowledge

CO1: Enhancing problem-solving abilities and understanding various algebraic structures in mathematics equips individuals with foundational skills crucial for effective problem resolution, facilitating a robust grasp of mathematical principles within the context of computer knowledge.
CO2: Defining functions and understanding related terms like domain, co-domain, and range in computer knowledge sharpens analytical skills, providing a fundamental framework for effective problem-solving and algorithm development.
CO3: Analyzing composite functions and comprehending invertible functions in computer knowledge fosters a deeper understanding of functional relationships, promoting proficiency in algorithmic design and problem-solving within the field.
CO4: Understanding and applying the concepts of monoids and semigroups in computer knowledge fosters a foundational comprehension of abstract algebraic structures, paving the way for efficient algorithmic design and solution development with relevant real-world examples.
CO5: Applying the principles of mathematical induction, including the first and second principles, in computer knowledge cultivates a systematic approach to problem-solving, providing a robust foundation for algorithmic reasoning and effective solution development, as illustrated through examples.
CO6: Understanding concepts related to relatively prime integers, Euclid's lemma, and its generalization in computer knowledge enriches algorithmic thinking, fostering the ability to design efficient and reliable solutions, and lays the groundwork for applications in cryptography, number theory, and various computational domains.

CO7: Solving systems of linear equations using the Gauss elimination method, Gauss-Jordan elimination method, and L.U. decomposition method in computer knowledge enhances problem-solving skills, providing a foundation for diverse applications in computational mathematics, engineering, and scientific computing..

PO2: Design / Development of solution

CO1: Enhancing problem-solving skills and understanding diverse algebraic structures in mathematics contributes to effective solutions in design and development.
CO4: Applying concepts of monoids and semigroups, with relevant examples, enriches the design and development process.
CO6:Understanding concepts related to relatively prime integers, Euclid's lemma, and its generalization enhances problem-solving skills in design and development.
CO7: Solving systems of linear equations using various methods sharpens problem-solving skills crucial for design and development.

PO3: Modern tool usage

CO1: Enhancing problem-solving skills and grasping diverse algebraic structures is essential in utilizing modern tools effectively.
CO3:Analyzing composite functions and comprehending invertible functions is crucial for adept utilization of modern tools.
CO5: Applying mathematical induction principles with examples enhances problem-solving skills in the utilization of modern tools.

PO5: Ethics

CO4:Understanding and applying monoids and semigroups, with relevant examples, fosters mathematical reasoning and ethical decision-making in complex problem-solving contexts.

PO6: Individual and Team work

CO1: Enhancing problem-solving skills and grasping diverse algebraic structures in mathematics contributes to collaborative problem-solving in both individual and team contexts.
CO4:Mastering monoids and semigroup concepts, along with practical examples, equips students for collaborative problem-solving in both individual and team settings.
CO7: Proficiency in solving systems of linear equations using various methods enhances students' capabilities for collaborative problem-solving in both individual and team environments.

PO7: Innovation, employability and Entrepreneurial skills
CO2: Defining functions and understanding related terms fosters innovative thinking, enhancing employability by cultivating a solid foundation in mathematical concepts essential for entrepreneurial success.

CO3: Analyzing composite functions and grasping invertible functions fosters innovative thinking, enhancing employability by honing essential problem-solving skills in diverse entrepreneurial scenarios.

CO4:Applying monoid and semigroup concepts, with real-world examples, fosters innovative thinking and enhances employability in entrepreneurial ventures.

CO6:Grasping concepts related to relatively prime integers, Euclid's lemma, and their generalization fosters innovative and entrepreneurial thinking, enhancing employability skills.

CO7:Solving linear equations using advanced methods enhances students' analytical skills and problem-solving acumen, fostering innovation and employability in diverse entrepreneurial scenarios

Choice Based Credit System Syllabus (2019 Pattern)

Class: F.Y.B.Sc.(Comp. Sci.) (Sem I) **Subject:** Mathematics
Course: Mathematics Practical based on CSMT1101 & CSMT1102 **Course Code:** CSMT1103

A) Course Objectives:

1. Problem solving ability and understanding applications of Graph Theory.
2. Apply graph theory concepts to solve real-world problems.
3. Explore applications of graph theory in various fields.
4. Understand the concept of connected graphs.
5. Understand the concept of echelon form.
6. Apply divisibility and congruence to solve problems.
7. Improve skills to handle abstract algebraic structures such as integers, groups.

B) Course Outcome:

1. Leadstudentstoapplythesemathematicalconceptsinthestudyofcomputerscience.
2. Students will demonstrate the application of connected graphs in various scenarios.
3. Students will analyze properties of Eulerian and Hamiltonian paths and cycles.
4. Students will recognize the significance of Eulerian and Hamiltonian graphs in practical contexts.
5. Students will explore finite and infinite groups, including cyclic groups.
6. Students will define and analyze congruence relations.
7. Students will apply algebraic concepts to solve complex problems

Title of Experiments:

Graph Theory:

1. Graphs and Operations on Graphs.
2. Connected Graphs.
3. Eulerian and Hamiltonian Graphs.
4. Trees.
5. Directed Graphs.
6. Miscellaneous

Algebra:

1. Relations and functions.
2. Binary Operations
3. Groups
4. Divisibility and Congruence
5. Matrices and System of linear Equations
6. Miscellaneous

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			1				
CO 2		2				3	1
CO 3		3			1	3	
CO 4						2	1
CO 5	1					3	1
CO 6	3	3	1			2	
CO 7		3	3			2	1

Justification for the mapping

PO1: Computer Knowledge

CO5: Student will gain insights into finite and infinite groups, specifically exploring the properties and applications of cyclic groups, enhancing their understanding in computer knowledge.
CO6: Students will develop proficiency in defining and analyzing congruence relations, fostering a deeper understanding of modular arithmetic in computer knowledge.

PO2: Design / Development of solution

CO2: Students will showcase their ability to apply connected graphs effectively in real-world scenarios, enhancing problem-solving skills in design and development processes.
CO3:Students mastering the analysis of Eulerian and Hamiltonian paths and cycles gain essential problem-solving skills for effective design and development solutions.
CO6:Analyzing congruence relations enhances students' problem-solving skills and their ability to design and develop solutions.
CO7: Applying algebraic concepts enables students to navigate and address intricate problems, fostering effective solution development in the design process.

PO3: Modern tool usage

CO1: Student will develop modern tool usage skills by being able to calculate the distance between two points in 2D space using Python libraries and computational geometry techniques.
CO6:Analyzing congruence relations in modern tools enhances students' proficiency in applying mathematical concepts to real-world problems and tools.
CO7:Applying algebraic concepts in modern tools empowers students to solve intricate problems efficiently, enhancing their proficiency in modern tool usage.

PO5: Ethics

CO3:By examining properties of Eulerian and Hamiltonian paths and cycles, students develop ethical reasoning skills essential for responsible decision-making in various professional contexts.

PO6: Individual and Team work

CO2:Applying connected graphs in diverse scenarios enhances students' problem-solving skills and collaborative abilities for effective engagement in both individual and team contexts.

CO3: Analyzing Eulerian and Hamiltonian paths and cycles enhances students' problem-solving skills and teamwork proficiency.
CO4:Recognizing the importance of Eulerian and Hamiltonian graphs enhances collaborative problem-solving skills, contributing to both individual and team success.
CO5:Exploration of finite and infinite groups, specifically cyclic groups, fosters collaborative problem-solving skills in both individual and team contexts.
CO6:Applying algebraic concepts enhances problem-solving skills for effective collaboration in both individual and team settings.
CO7:Applying algebraic concepts enhances problem-solving skills, fostering collaboration in both individual and team environments.

PO7: Innovation, employability and Entrepreneurial skills

CO2: Applying connected graphs in diverse scenarios enhances students' innovation and employability, fostering entrepreneurial skills through practical problem-solving.

CO4: Understanding the concepts of linear programming and its applications in Python enhances students' innovation and employability skills by enabling them to solve complex optimization problems and make data-driven decisions, crucial for entrepreneurial success in a data-driven world.

CO5:Recognition of Eulerian and Hamiltonian graphs fosters innovative problem-solving skills, enhancing employability through practical applications in diverse entrepreneurial scenarios.

CO7:Applying algebraic concepts fosters innovative problem-solving skills, enhancing employability and entrepreneurial capabilities in students.