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	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:

Semester	Old Course		New Course	
	F.Y.B.Sc.(Comp. Sci.)			
I	CSMT1101	Graph Theory	UCSMT111	Graph Theory
	CSMT1102	Algebra	UCSMT112	Matrix Algebra
	CSMT1103	Mathematics Practical based on CSMT1101 & CSMT1102	UCSMT113	Mathematics Practical based on UCSMT111 & UCSMT112
II	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
	S.Y.B.Sc.(Comp. Sci.)			
III	CSMT2301	Linear Algebra	UCSMT231	Groups and Coding Theory
	CSMT2302	Numerical Analysis	UCSMT232	Numerical Techniques
	CSMT2303	Mathematics Practical I	UCSMT233	Mathematics Practical Python Programming Language I
IV	CSMT2401	Computational Geometry	UCSMT241	Computational Geometry
	CSMT2402	Operations Research	UCSMT242	Operation Research
	CSMT2403	Mathematics Practical II	UCSMT243	Mathematics Practical Python Programming Language II

Choice Based Credit System Syllabus (2022 Pattern)

Class: S.Y.B.Sc.(Computer Science). (Sem IV)

Subject: Mathematics

Course: Computational Geometry **Course Code:** UCSMT241

A) Course Objectives

1. Provide students with a foundational understanding of two-dimensional transformations and their applications in computer graphics.
2. Introduce methods for representing points in two-dimensional space and emphasize the importance of coordinate systems.
3. Teach students the concept of transformation matrices and how they are used to perform various geometric transformations.
4. Instruct students on the midpoint transformation, emphasizing its application and significance in graphics algorithms.
5. Demonstrate the application of transformations to parallel and intersecting lines, emphasizing practical scenarios.
6. Provide students with the skills to apply multiple transformations sequentially, enabling them to achieve complex transformations.
7. Introduce solid body transformations, allowing students to extend their understanding to three-dimensional transformations.

B) Course Outcomes

1. Student will be able to apply knowledge of the fundamental problems withincomputational geometry and general techniques for solving problems.
2. Student will be able to construct algorithms for simple geometrical problems.
3. Understand the fundamental concepts of two-dimensional transformations incomputer graphics.
4. Perform rotations about axes parallel to coordinate axes and arbitrary lines.
5. Apply axonometric projections to three-dimensional objects.
6. Understand the representation of curves in computer graphics.
7. Perform reflections through coordinate planes, planes parallel to coordinate planes,and arbitrary planes.

TOPICS/CONTENT

Unit 1. Two dimensional transformations: **[12 Lectures]**

1.1 Introduction.
1.2 Representation of points.
1.3 Transformations and matrices.
1.4 Transformation of points.
1.5 Transformation of straight lines
1.6 Midpoint Transformation
1.7 Transformation of parallel lines
1.8 Transformation of intersecting lines
1.5 Transformation: rotations, reflections, scaling, shearing.
1.6 Combined transformations.
1.7 Transformation of a unit square.
1.8 Solid body transformations.

1.9 Translations and homogeneous coordinates.
1.10 Rotation about an arbitrary point.
1.11 Reflection through an arbitrary line.

Unit 2. Three dimensional transformations: **[10 Lectures]**

2.1 Introduction.
2.2 Three dimensional – Scaling, shearing, rotation, reflection, translation.
2.3 Multiple transformations.
2.4 Rotation about – an axis parallel to coordinate axes, an arbitrary line
2.5 Reflection through – coordinate planes, planes parallel to coordinateplanes, an arbitrary plane.

Unit 3. Projection **[12 Lectures]**

3.1 Orthographic projections.
3.2 Axonometric projections.
3.3 Oblique projections
3.4 Single point perspective projection

Unit 4. Plane and space Curves: **[14 Lectures]**

4.1 Introduction.
4.2 Curve representation.
4.3 Parametric curves.
4.4 Parametric representation of a circle and generation of circle.
4.5 Bezier Curves – Introduction, definition, properties (without proof), Curve fitting (upto n = 3), equation of the curve in matrix form (upto n = 3)

Textbook:

D. F. Rogers, J. A. Adams, Mathematical elements for Computer graphics,McGraw Hill Intnl Edition.

Unit 1: Chapter 2: Sec. 2-1 to 2.17
Unit 2: Chapter 3: Sec. 3.1 to 3.10,
Unit 3: Chapter 3: Sec. 3.12 to 3.14
Unit 4: Chapter 4: Sec. 4.1, 4.2, 4.5, Chapter 5: Sec. 5.1, 5.8

Reference books:

1. Computer Graphics with OpenGL, Donald Hearn, M. Pauline Baker, Warren Carithers,Pearson (4th Edition)
2. Schaum Series, Computer Graphics.

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

Justification for the mapping

PO1: Computer Knowledge

CO1: Student will be able to apply their knowledge of fundamental computational geometry problems and general problem-solving techniques to address real-world challenges in computer science and related fields.
CO2: Student will be able to construct algorithms for simple geometrical problems in computer knowledge to develop problem-solving skills and apply geometric concepts in practical programming tasks.
CO3: Student will gain a foundational understanding of two-dimensional transformations in computer graphics, essential for creating and manipulating graphical elements in various applications.
CO4: Student will gain proficiency in performing rotations around both coordinate axes and arbitrary lines, enhancing their 3D graphics and spatial manipulation skills in computer knowledge.
CO5: Student will acquire the ability to apply axonometric projections to three-dimensional objects, enhancing their proficiency in computer graphics and visualization.
CO6: Student will gain a comprehensive understanding of curve representation in computer graphics, enabling them to effectively create and manipulate visual elements in digital environments.
CO7: Student will gain the ability to perform reflections through various planes, enhancing their understanding of geometric transformations in computer graphics and 3D modeling.

PO2: Design / Development of solution

CO2: Student will be able to construct algorithms for simple geometrical problems to facilitate the design and development of efficient and accurate solutions in computer graphics and geometric modeling.

CO3: Student will be equipped to design and develop solutions in computer graphics by comprehending the fundamental concepts of two-dimensional transformations, enabling them to manipulate and transform graphical elements effectively.
CO7: Student will perform reflections through various planes to enhance their ability to visualize and manipulate 3D objects, a fundamental skill in design and solution development for computer graphics and engineering applications.

PO3: Modern tool usage

CO5: Student will apply axonometric projections to three-dimensional objects as a modern tool usage to enhance their ability to create accurate and visually appealing 3D representations in various fields, including computer graphics and engineering.

PO6: Individual and Team work

CO1: Student will develop the ability to apply their knowledge of computational geometry and problem-solving techniques to tackle fundamental problems both individually and as part of a team, enhancing their practical problem-solving skills and collaboration capabilities.
CO6: Student will understand the representation of curves in computer graphics to enable them to contribute effectively both individually and as part of a team in creating visually appealing and accurate graphical content.

PO7: Innovation, employability and Entrepreneurial skills

CO1: Student will apply knowledge of computational geometry and problem-solving techniques to enhance their innovation, employability, and entrepreneurial skills by tackling real-world challenges and developing creative solutions.

CO3: Understanding two-dimensional transformations in computer graphics enhances students' innovation, employability, and entrepreneurial skills by providing them with the foundational knowledge necessary for creating innovative visual content, which is valuable in various industries and entrepreneurial ventures.

Choice Based Credit System Syllabus (2022 Pattern)

Class: S.Y.B.Sc.(Computer Science). (Sem IV)

Subject: Mathematics

Course: Operations Research **Course Code:** UCSMT242

A) Course Objectives

1. Develop proficiency in representing real-world scenarios using LP models with two decision variables.
2. Develop skills in visually analyzing and optimizing solutions for two-variable LP models.
3. Develop the ability to model and optimize decision-making processes using linear programming.
4. Translate real-world problems into algebraic expressions suitable for linear programming analysis.
5. Provide a comprehensive understanding of the simplex method for solving linear programming problems.
6. Develop strategies for addressing and solving special cases efficiently.
7. Analyze decision-making under uncertainty using decision analysis techniques.

B) Course Outcomes

1. Student will be able to formulate and solve the linear programming problem usingdifferent methods.
2. Student will able to solve transportation and assignment problems.
3. Utilize the Big-M method for solving linear programming problems.
4. Recognize and address unbounded and infeasible solutions in linear programming.
5. Apply the modified distribution method to find optimal solutions for transportationproblems.
6. Apply the Hungarian method to solve assignment problems.
7. Interpret the relationship between primal and dual solutions.

TOPICS/CONTENT

Unit 1: Linear Programming Problem **[16 Lectures]**

1.1 Introduction and basic definitions.

1.2 Formulation of LPP.

1.3 Graphical method.

1.4 Simplex method.

1.5 Big-M method.

Unit 2: Special cases in LPP and concept of duality **[14 Lectures]**

2.1 Alternative solution.

2.2 Unbounded solution.

2.3 Infeasible solution.

2.4 Duality in Linear Programming.

2.5 Primal to dual conversion.

Unit 3: Transportation Models **[10 Lectures]**

3.1 Introduction and basic definitions.

3.2 Initial basic feasible solutions using North-West rule, Matrix-minima, Vogel'sApproximation.

3.3 Modified distribution method for optimal solution.

3.4 Special cases in transportation problems.

Unit 4: Assignment Models **[08 Lectures]**

4.1 Introduction and basic definitions.

4.2 Hungerian method for assignment problem.

4.3 Special cases in assignment problems.

Text Book:-

J. K. Sharma, Operations Research: Theory and Applications, Trinity Press, 6th Edition,2016.

Unit 1: Chapter 2: Sec. 2.1 to 2.8.
Chapter 3: Sec. 3.1 to 3.4.
Chapter 4: Sec. 4.1 to 4.4.
Unit 2: Chapter 4: Sec. 4.5 and 4.6.
Chapter 5: Sec. 5.1 and 5.2.
Unit 3: Chapter 9: Sec. 9.1 and 9.7.
Unit 4: Chapter 10: Sec. 10.1 to 10.4.

Reference Books:-

1. H. A. Taha, Operations Research: An introduction, Prentice Hall of India, 8th edition, 2006.
2. S. D. Sharma, Operations Research, Kedarnath Ramnath & Company, 1992.
3. D. S. Hira and Prem Kumar Gupta, Operations Research, S. Chand and Company Ltd., 2016.
4. R. Panneerselvam, Operations Research, PHI Learning Pvt. Ltd., 2nd edition, 2009.
5. H. M. Wagner, Principles of Operations Research, PHI Learning Pvt. Ltd., 2nd edition.
6. K. Rajagopal, Operations Research, PHI Learning Pvt. Ltd., Eastern Economy 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	3	3					3
CO 2	3	3	3			3	1
CO 3	3	2					
CO 4	3		2				
CO 5	3						3
CO 6	3					2	
CO 7	3	3					3

Justification for the mapping

PO1: Computer Knowledge

CO1: Student will gain the ability to formulate and solve linear programming problems using various methods, enhancing their computational knowledge and problem-solving skills.
CO2: Student will gain proficiency in solving transportation and assignment problems to enhance their computational skills and problem-solving capabilities in computer science and related fields.
CO3: Student will be equipped with the essential skill of applying the Big-M method to effectively tackle complex linear programming problems, enhancing their proficiency in computer-aided optimization.

CO4: Student will develop the ability to identify and resolve unbounded and infeasible solutions in linear programming, enhancing their problem-solving skills in computer-aided optimization.
CO5: Student will develop the ability to apply the modified distribution method to efficiently determine optimal solutions for transportation problems, enhancing their problem-solving skills in computer science and logistics.
CO6: Student will gain a comprehensive understanding of curve representation in computer graphics, enabling them to effectively create and manipulate visual elements in digital environments.
CO7: Student will interpret the relationship between primal and dual solutions to gain a deeper understanding of the duality concept in linear programming, a fundamental component of their computer knowledge.

PO2: Design / Development of solution

CO1: Student will be equipped to formulate and solve linear programming problems using various methods, enabling them to design and develop optimal solutions for real-world challenges in operations research and decision-making.
CO2: Student will be able to solve transportation and assignment problems, equipping them with essential skills for optimizing resource allocation and logistics in the design and development of practical solutions.
CO3: Student will be equipped with the ability to employ the Big-M method as a powerful tool in the design and development of solutions for complex linear programming problems, ensuring optimal problem-solving strategies.
CO7: Student will be able to interpret the relationship between primal and dual solutions to optimize and enhance the design and development of solutions in various problem-solving scenarios.

PO3: Modern tool usage

CO2: Student will be equipped to solve transportation and assignment problems efficiently using modern computational tools, enhancing their problem-solving skills in real-world logistics and allocation scenarios.
CO4: Students will be able to identify and resolve unbounded and infeasible solutions effectively in linear programming using modern computational tools.

PO6: Individual and Team work

CO2: Student will develop problem-solving skills through individual and team work, enabling them to effectively solve transportation and assignment problems, both independently and collaboratively.
CO6: Students will learn to apply the Hungarian method for efficient assignment problem solving, both independently and collaboratively, promoting problem-solving skills in real-world scenarios.

PO7: Innovation, employability and Entrepreneurial skills

CO1: Student will gain the ability to formulate and solve linear programming problems using various methods, enhancing their innovation, employability, and entrepreneurial skills by providing them with a valuable analytical and decision-making toolset for problem-solving in real-world applications.

CO2: Student who can solve transportation and assignment problems develop problem-solving and analytical skills, enhancing their innovation, employability, and entrepreneurial abilities by addressing real-world logistics and resource allocation challenges.

CO5: Student will employ the modified distribution method for transportation problems, fostering innovation and enhancing their employability and entrepreneurial skills through practical problem-solving and optimization techniques.

CO7: Understanding the relationship between primal and dual solutions fosters critical problem-solving abilities, enhancing students' innovation, employability, and entrepreneurial skills in optimizing complex business and engineering processes

Choice Based Credit System Syllabus (2022 Pattern)

Class: S.Y.B.Sc.(Computer Science). (Sem IV)

Subject: Mathematics

Course: Mathematics Practical Python Programming Language II **Course Code:** UCSMT243

A) Course Objectives

1. Install and configure numpy, matplotlib, and related packages, preparing the environment for advanced mathematical computations and visualizations.
2. Learn to plot basic and advanced functions (e.g., linear, quadratic, trigonometric functions) and customize visual aspects like colors, labels, and styles in two-dimensional graphs.
3. Understand and differentiate between various graph formats such as bar plots, scatter plots, histograms, and more specialized plots, analyzing their use cases and visual interpretation.
4. Learn to plot three-dimensional data, including points, lines, and surfaces, using matplotlib to represent complex geometrical objects in a 3D space.
5. Create contour plots and surface plots to analyze and represent multivariable functions in three dimensions, understanding the graphical representation of levels and surfaces.
6. Apply two-dimensional and three-dimensional transformations such as rotation and reflection to points, lines, and polygons, and visualize the results through plots.
7. Learn to implement and solve optimization problems using the Simplex method with Python libraries, focusing on practical applications and visualization of results.

B) Course Outcomes

1. Successfully install and utilize numpy and matplotlib packages for graphing.
2. Calculate the distance between two points in 2D space.
3. Understand the geometry of line segments and perform related calculations.
4. Understand the concepts of linear programming and its applications in Python.
5. Student will be able to design program in python.
6. Student will be able to solve problems in computational geometry and operationsresearch using python programming.
7. Apply three-dimensional rotation and reflection transformations.

TOPICS/CONTENT

Unit 1: 2D, 3D Graphs

1.1 Installation of numpy, matplotlib packages

1.2 Graphs plotting of functions such as ... etc.

1.3 Different formats of graphs.

1.3 Three-dimensional Points and Lines

1.4 Three-dimensional Contour Plots

1.5 Wireframes and Surface Plots

1.6 Graphs plotting of functions such as... etc.

Unit 2: Computational Geometry

1.1 Points: The distance between two points, Lists of Points - the Point List class, Integer point lists, Ordered Point sets, Extreme Points of a Point List, Random sets of Points notin general position

2.2 Points: Displaying Points and other geometrical objects, Lines, rays, and line segments,The geometry of line segments, Displaying lines, rays and line segments

2.3Polygon : Representing polygons in Python, Triangles, Signed area of a triangle, Triangles and the relationships of points to lines, is Collinear, is Left, is Left On, is Right,is Right On, Between

2.4 Two dimensional rotation and reflection

2.5 Three dimensional rotation and reflection

2.6 Generation of Bezier curve with given control points

Unit 3: Study of Operational Research in Python

3.1 Linear Programming in Python

3.2 Introduction to Simplex Method in Python.

Practicals:

Practical 1: Graph Plotting (Unit 1 – 1.1 – 1.3)

Practical 2: Graph Plotting (Unit 1 – 1.4 – 1.7)

Practical 3: Application to Computational Geometry (Unit 2 – 2.1) **Practical 4:** Application to Computational Geometry (Unit 2 – 2.2) **Practical 5:** Application to Computational Geometry (Unit 2 – 2.3)

Practical 6: Study of Graphical aspects of Two dimensional transformation matrix usingmatplotlib

Practical 7: Study of Graphical aspects of Three dimensional transformation matrix usingmatplotlib

Practical 8: Study of Graphical aspects of Three dimensional transformation matrix usingmatplotlib

Practical 9: Study of effect of concatenation of Two dimensional and Three dimensional transformations

Practical 10: Generation of Bezier curve using given control points**Practical 11:** Study of Operational Research in Python (Unit 3.1) **Practical 12:** Study of Operational Research in Python (Unit 3.2)

Text Books:-

1. Jaan Kiusalaas, Numerical Methods in Engineering with Python, CambridgeUniversity Press, (2005)
   Sections: 3
2. Robert Johansson, Introduction to Scientific Computing in PythonSection: 1
3. Jason Brownlee, Basics of Linear Algebra for Machine Learning, Discover theMathematical Language of Data in Python
   Sections: 2

Reference Books:-

1. Lambert K. A., Fundamentals of Python - First Programs, Cengage LearningIndia,2015.

2. Guzdial, M. J., Introduction to Computing and Programming in Python, Pearson India.

3. Perkovic, L., Introduction to Computing Using Python, 2/e, John Wiley, 2015.

4. Zelle, J., Python Programming: An Introduction to Computer Science,Franklin,Beedle and Associates Inc.

5. Jim Arlow, Interactive Computational Geometry in Python

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

Justification for the mapping

PO1: Computer Knowledge

CO1: Student will acquire essential skills in data visualization and graphing by successfully installing and utilizing the numpy and matplotlib packages in Python for effective graphical representation of data.

CO2: Student will learn to calculate the distance between two points in 2D space, enhancing their computational skills and problem-solving abilities in computer science and mathematics.

CO3: Student will acquire the ability to comprehend the geometry of line segments and apply mathematical calculations to solve practical problems in computer graphics and computational geometry.

CO4: Student will gain a deep understanding of linear programming concepts and their practical applications in Python, enhancing their computer knowledge for real-world problem-solving and optimization.

CO5: Student will acquire the ability to design and implement Python programs to solve complex computer-related problems and gain practical computer knowledge.

CO6: Student will gain the ability to effectively solve problems in computational geometry and operations research by applying Python programming skills, enhancing their computer knowledge for practical problem-solving in these domains.

CO7: Student will be able to apply three-dimensional rotation and reflection transformations to manipulate and visualize 3D objects in computer graphics and computer-aided design applications.

PO2: Design / Development of solution

CO1: Student will acquire the essential skills to efficiently employ numpy and matplotlib packages for data visualization and graphing, enabling effective solution design and development in various domains.
CO2: Student will gain the essential skill of calculating distances between two points in 2D space, a fundamental concept in designing and developing spatial solutions for applications such as mapping, computer graphics, and navigation systems.
CO3: Understanding the geometry of line segments and performing related calculations is essential for the design and development of geometric solutions, enabling students to create accurate and precise graphical representations and solve geometric problems effectively.
CO4: Student will understand the concepts of linear programming in Python, enabling them to design and develop efficient optimization solutions for real-world problems.
CO5: Students will be able to design and develop Python programs to create effective solutions for a wide range of computational problems.
CO6: Student will gain proficiency in computational geometry and operations research, equipping them to design and develop Python-based solutions to tackle real-world problems in geometric analysis and optimization.
CO7: Student will be able to apply three-dimensional rotation and reflection transformations to design and develop solutions for creating 3D graphical models and animations, enhancing their skills in computer graphics and visualization.

PO3: Modern tool usage

CO2: 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.
CO5: Student will be able to design programs in Python, demonstrating modern tool usage, for practical problem-solving and data analysis.
CO7: Student will be able to apply three-dimensional rotation and reflection transformations in modern tool usage, enhancing their ability to manipulate and visualize 3D objects in computer graphics and simulation applications.

PO6: Individual and Team work

CO3: Students will develop the ability to analyze line segments in geometric contexts and apply their knowledge to solve practical problems, both independently and collaboratively as part of a team.
CO6: Students will acquire the ability to apply computational geometry and operations research principles using Python for both individual problem-solving and collaborative team projects, enabling them to tackle real-world challenges effectively.
CO7: Student will master the application of three-dimensional rotation and reflection transformations, enabling them to contribute effectively to both individual and team-based projects involving 3D spatial data manipulation and visualization.

PO7: Innovation, employability and Entrepreneurial skills

CO2: Student will acquire problem-solving and spatial analysis skills, enhancing their innovation and employability by being able to calculate distances between points in 2D space, which is essential for various applications in fields such as engineering, geography, and data analysis, and can be a valuable entrepreneurial skill.

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.

CO7: Students will develop practical 3D rotation and reflection skills, essential for innovation and employability in fields like computer graphics and 3D modeling, fostering entrepreneurial opportunities in emerging technologies.