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
III	UCSMT231	Groups and Coding Theory	3	48
	UCSMT232	Numerical Techniques	3	48
	UCSMT233	Mathematics Practical Python Programming Language I	2	48
IV	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

SYLLABUS (CBCS) FOR S. Y. B. Sc.(COMPUTER SCIENCE) MATHEMATICS
(w.e.f. June, 2023)

Academic Year 2023-24

Class : S.Y.B.Sc. (Comp. Sci.) (Semester – III)

Course Code: UCSMT 231 Title of the Course: Groups and Coding Theory

Course: I Credit: 3 No. of lectures: 48

A) Learning Objectives

- To introduce concept of relation.
- To introduce basic algebraic properties of groups.
- Use algebraic techniques to construct efficient codes.

B) Learning Outcomes

- Student will be able to recognize a set with given operation is group or not.
- Student will be able to use public key cryptography.

TOPICS/CONTENT

Unit 1: Integers [12 Lectures]

1.1 Division algorithm.
1.2 G.C.D. and Euclidean algorithm.
1.3 Euclid's lemma.
1.4 Equivalence relation (revision), Congruence relation on set of integers.
1.5 Equivalence class and partitions.

Unit 2: Groups [08 Lectures]

2.1 Binary Operation
2.2 Group: Definition and Examples
2.3 Elementary Properties of Groups

Unit 3: Finite Groups and Subgroups [16 Lectures]

3.1 Basic terminologies.
3.2 Subgroup test.
3.3 Cyclic groups.
3.4 Properties of cyclic groups.
3.5 Classification of subgroups of cyclic groups.
3.6 Permutation groups.
3.7 Properties of permutation groups.
3.8 Cosets.
3.9 Properties of cosets.
3.10 Lagrange theorem.

Unit 4: Groups and Coding Theory [12 Lectures]

4.1 Coding of Binary Information and Error detection
4.2 Decoding and Error Correction
4.3 Public Key Cryptography

Text Book:

1. J. A. Gallian, Contemporary Abstract Algebra, Narosa, 7th Edition
   Unit 1: Chapter 0
   Unit 2: Chapter 2
   Unit 3: Chapters 3, 4, 5 and 7
2. Bernard Kolman, Robert C. Busby and Sharon Ross, Discrete Mathematical Structures, Pearson Education Publication, 6th Edition.
   Unit 4: Chapter 11

Reference Book:

1. N. S. Gopalakrishnan, University Algebra, New Age International (P) Ltd, Publishers, 2nd Edition (1986).
2. P. B. Bhattacharya, S. K. Jain, S. R. Nagpaul, Basic Abstract Algebra, Cambridge University Press, 2nd Edition (1994).
3. I. N. Herstein, Topics in Algebra, Wiley, 2nd Edition.
4. J. H. van Lint, Introduction to Coding Theory, Springer.

Academic Year 2023-24

Class: S.Y.B.Sc. (Comp. Sci.) (Semester – III)

Course Code: UCSMT 232 Title of the Course: Numerical Technique

Course: II Credit: 3 No. of lectures: 48

A) Learning Objectives

- To introduce calculus of finite difference.
- To introduce methods in numerical integration.
- To solve ordinary differential equations using numerical techniques.

B) Learning Outcomes

- Student will be able to solve integration using numerical techniques.
- Student will be able to apply numerical method for solving ordinary differential equations.

TOPICS/CONTENT

Unit 1: Algebraic and Transcendental Equation [08 Lectures]

1.1 Introduction to Errors
1.2 False Position Method
1.3 Newton-Raphson Method

Unit 2: Calculus of Finite Differences and Interpolation [16 Lectures]

2.1 Differences
2.2 Forward Differences
2.3 Backward Differences
2.4 Central Differences
2.5 Other Differences (δ, μ operators)
2.6 Properties of Operators
2.7 Relation between Operators
2.8 Newton's Gregory Formula for Forward Interpolation
2.9 Newton's Gregory Formula for Backward Interpolation
2.10 Lagrange's Interpolation Formula
2.11 Divided Difference
2.12 Newton's Divided Difference Formula

Unit 3: Numerical Integration [12 Lectures]

3.1 General Quadrature Formula
3.2 Trapezoidal Rule
3.3 Simpson's one-Third Rule
3.4 Simpson's Three-Eight Rule

Unit 4: Numerical Solution of Ordinary Differential Equation [12 Lectures]

4.1 Euler's Method
4.2 Euler's Modified Method
4.3 Runge-Kutta Methods

Text Book:

1. A. K.Jaiswal and Anju Khandelwal, A textbook of Computer Based Numerical and Statistical Techniques, New Age International Publishers.

Unit 1: Chapter 2: Sec. 2.1, 2.5, 2.7

Unit 2: Chapter 3: Sec. 3.1, 3.2, 3.4, 3.5, Chapter 4: Sec. 4.1, 4.2, 4.3, Chapter 5: Sec. 5.1, 5.2, 5.4, 5.5

Unit 3: Chapter 6: Sec. 6.1, 6.3, 6.4, 6.5, 6.6, 6.7

Unit 4: Chapter 7: Sec. 7.1, 7.4, 7.5, 7.6

Reference Book:

1. S.S. Sastry; Introductory Methods of Numerical Analysis, 3rd edition,Prentice Hall of India, 1999.
2. H.C. Saxena; Finite differences and Numerical Analysis, S. Chand andCompany.
3. K.E. Atkinson; An Introduction to Numerical Analysis, Wiley Publications.
4. Balgurusamy; Numerical Analysis.

Academic Year 2023-24

Class: S.Y.B.Sc. (Comp. Sci.) (Semester – III)

Course Code: UCSMT 233**Title of the Course:** Mathematics Practical: Python Programming I

Course: III **Credit: 2** **No. of lectures: 48**

A) Learning Objectives

- To introduce basic concepts in python programming.
- To solve problems in linear algebra using python programming.
- To solve problems in numerical techniques using python programming.

B) Learning Outcomes

- Student will be able to design program in python.
- Student will be able to solve problems in linear algebra and numerical techniques using python programming.

TOPICS/CONTENT

Unit 1: Introduction to Python

1.1 Installation of Python
1.2 Values and types: int, float and str,
1.3 Variables: assignment statements, printing variable values, types of variables.
1.4 Operators, operands and precedence:+, -, /, *, **, % PEMDAS(Rules of precedence)
1.5 String operations: + : Concatenation, * : Repetition
1.6 Boolean operator:
    1.6.1 Comparison operators: ==, !=, >, =, <=
    1.6.2 Logical operators: and, or, not
1.7 Mathematical functions from math, cmath modules.
1.8 Keyboard input: input() statement

Unit 2: String, list, tuple

2.1 Strings:
    2.1.1 Length (Len function)
    2.1.2 String traversal: Using while statement, Using for statement
    2.1.3 String slice
    2.1.4 Comparison operators (>, <, ==)
2.2 Lists:
    2.2.1 List operations
    2.2.2 Use of range function
    2.2.3 Accessing list elements
    2.2.4 List membership and for loop
    2.2.5 List operations
    2.2.6 Updating list: addition, removal or updating of elements of a list
2.3 Tuples:
    2.3.1 Defining a tuple,
    2.3.2 Index operator,
    2.3.3 Slice operator,
    2.3.4 Tuple assignment,
    2.3.5 Tuple as a return value

Unit 3: Iterations and Conditional statements

3.1 Conditional and alternative statements, Chained and Nested Conditionals: if, if-else,if-elif-else, nested if, nested if-else

3.2 Looping statements such as while, for etc, Tables using while.

3.3 Functions:

3.3.1 Calling functions: type, id

3.3.2 Type conversion: int, float, str

3.3.3 Composition of functions

3.3.4 User defined functions, Parameters and arguments

Unit 4: Linear Algebra

4.1 Matrix construct, eye(n), zeros(n,m) matrices

4.2 Addition, Subtraction, Multiplication of matrices, powers and invers of a matrix.

4.3 Accessing Rows and Columns, Deleting and Inserting Rows and Columns

4.4 Determinant, reduced row echelon form, nullspace, columnspace, Rank

4.5 Solving systems of linear equations (Gauss Elimination Method, Gauss Jordan Method,LU- decomposition Method)

4.6 Eigenvalues, Eigenvectors, and Diagonalization

Unit 5: Numerical methods in Python

5.1 Roots of Equations

5.2 Newton-Raphson Method

5.3 False Position (Regula Falsi) Method

5.4 Numerical Integration:

5.1.1 Trapezoidal Rule,

5.1.2 Simpson's $1/3^{rd}$ Rule,

5.1.3 Simpson's $3/8^{th}$ Rule

Text Books:-

1. Downey, A. et al., How to think like a Computer Scientist: Learning with Python,John Wiley, 2015.
Sections: 1, 2, 3

2. Robert Johansson, Introduction to Scientific Computing in Python
Section: 4

Reference Books:-

1. Lambert K. A., Fundamentals of Python - First Programs, Cengage Learning India, 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 & Associates Inc.

5. Sandro Tosi, Matplotlib for Python Developers, Packt Publishing Ltd.(2009)

Practicals:

Practical 1: Introduction to Python, Python Data Types-I (Unit 1)
Practical 2: Python Data Types- II (Unit 2)
Practical 3: Control statements in Python-I (Unit 3- 3.1, 3.2)
Practical 4: Control statements in Python-II (Unit 3- 3.3)
Practical 5: Application: Matrices (Unit 4 – 4.1-4.3)
Practical 6: Application: Determinants, system of Linear Equations (Unit 4- 4.4, 4.5)
Practical 7: Application: System of equations (Unit 4- 4.5)
Practical 8: Application: Eigenvalues, Eigenvectors (Unit 4 – 4.6)
Practical 9: Application: Eigenvalues, Eigenvectors (Unit 4 – 4.6)
Practical 10: Application: Roots of equations (Unit 5 – 5.1)
Practical 11: Application: Numerical integration (Unit 5 – 5.2, 5.3)
Practical 12: Application: Numerical integration (Unit 5 – 5.4)