University of Pune

Board of Studies in Mathematics

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

Syllabus of Mathematics

Introduction:

Savitribai Phule Pune University, Pune has decided to change the syllabi of various faculties from June, 2020. Taking into consideration the rapid changes in science and technology and new approaches in different areas of mathematics and related subjects Board of studies in Mathematics with concern of teachers of Mathematics from different colleges affiliated to Savitribai Phule Pune University, Pune has prepared the syllabus of S.Y.B.Sc. Computer Science Mathematics. To develop the syllabus the U.G.C. Model curriculum is followed.

Aims:

i) Give the students a sufficient knowledge of fundamental principles ,methods and a clear perception of innumerous power of mathematical ideas and tools and know how to use them by modeling ,solving and interpreting.

ii) Reflecting the broad nature of the subject and developing mathematical tools for continuing further study in various fields of science.

iii) Enhancing students overall development and to equip them with mathematical modeling abilities, problem solving skills , creative talent and power of communication necessary for various kinds of employment .

iv) Enabling students to develop a positive attitude towards mathematics as an interesting and valuable subject of study.

Objectives:

(i) A student should be able to recall basic facts about mathematics and should be able to display knowledge of conventions such as notations, terminology and recognize basic geometrical figures and graphical displays, state important facts resulting from their studies.

(ii) A student should get a relational understanding of mathematical concepts and concerned structures, and should be able to follow the patterns involved, mathematical reasoning.

(iii) A student should get adequate exposure to global and local concerns that explore them many aspects of Mathematical Sciences.

(iv) A student be able to apply their skills and knowledge, that is, translate information presented verbally into mathematical form, select and use appropriate mathematical formulae or techniques in order to process the information and draw the relevant conclusion.

(v) A student should be made aware of history of mathematics and hence of its past, present and future role as part of our culture.

*** Medium of Instruction: English**

*** Eligibility:** F.Y.B.Sc. Computer Science, as per University rules.

Structure of the course:

Semester - I			**Semester -II**	
Paper I	**MTC-231**	**Groups and Coding Theory**	**MTC-241**	**Computational Geometry**
Paper II	**MTC-232**	**Numerical Techniques**	**MTC-242**	**Operations Research**
Paper III	**MTC-233**	**Mathematics Practical: Python Programming Language-I**	**MTC-243**	**Mathematics Practical: Python Programming Language-II**

* **All three above courses are compulsory.**

*** External Students: Not allowed.**

*** Variation / Revaluation: Allowed for Paper- I and Paper-II.**

*** Qualifications for Teacher: M.Sc. Mathematics (with NET /SET as per existing rules)**

Equivalence of Previous syllabus along with new syllabus:

	Semester-III		**Semester-IV**	
	New Course	**Old Course**	**New Course**	**Old Course**
Paper I	**MTC-231: Groups and Coding Theory**	**MTC-211 : Applied Algebra**	**MTC-241: Computational Geometry**	**MTC-221: Computational Geometry**
Paper II	**MTC-232: Numerical Techniques**	**MTC-212: Numerical Analysis**	**MTC-242: Operations Research**	**MTC-222: Operations Research**

Paper III	**MTC-233: Mathematics Practical: Python Programming Language-I**	**MTC-213 : Mathematics Practical**	**MTC-243: Mathematics Practical: Python Programming Language-II**	**MTC–223: Mathematics Practical**

Semester III

MTC-231 : Groups and Coding Theory

Unit 1. Integers **[05 Lectures]**

1.1 Division Algorithm (without Proof)

1.2 G.C.D. using division algorithm and expressing it as linear combination

1.3 Euclid's lemma

1.4 Equivalence relation (revision), Congruence relation on set of integers, Equivalence class partition

Unit 2. Groups **[03 Lectures]**

2.1 Binary Operation

2.2 Group: Definition and Examples

2.3 Elementary Properties of Groups

Unit 3. Finite Groups and Subgroups **[10 Lectures]**

3.1 Order of a group, order of an element

3.2 Examples (Zn, +) and (U(n), *)

3.3 Subgroup definition, Finite subgroup test, subgroups of Zn

3.4 Generator, cyclic group, finding generators of Zn(Corollary 3,4 without proof)

3.5 Permutation group, definition, composition of two permutations, representation as product of disjoint cycles, inverse and order of a permutation, even/ odd permutation

3.6 Cosets: Definition, Examples and Properties, Lagrange Theorem(without Proof)

Unit 4. Groups and Coding Theory **[18 Lectures]**

4.1 Coding of Binary Information and Error detection

4.2 Decoding and Error Correction

4.3 Public Key Cryptography

Text Books:-

1. **Contemporary Abstract Algebra By J. A, Gallian (Seventh Edition)**
   Unit 1:Chapter 0, Unit 2: Chapter 2, Unit 3: Chapter 3 ,4, 5 and 7
2. **Discrete Mathematical Stuctures By Bernard Kolman, Robert C. Busby and Sharon Ross (6th Edition) Pearson Education Publication**
   Unit 4: Chapter 11

MTC-232 : Numerical Techniques

Unit 1: Algebraic and Transcendental Equation [04 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 [08 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 [08 Lectures]

4.1 Euler's Method

4.2 Euler's Modified Method

4.3 Runge-Kutta Methods

Text Book:-

1. A textbook of Computer Based Numerical and Statistical Techniques, by A. K. Jaiswal and Anju Khandelwal. 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 Books:-

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 and Company.
3. K.E. Atkinson; An Introduction to Numerical Analysis, Wiley Publications.
4. Balguruswamy; Numerical Analysis.

MTC-233: Mathematics Practical: Python Programming Language-I

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) Mehtod

5.4 Numerical Integration:
- 5.1.1 Trapezoidal Rule,
- 5.1.2 Simpson's 1/3rd Rule,
- 5.1.3 Simpson's 3/8th 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)

Semester - IV

MTC-241: Computational Geometry

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: [08 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 coordinate planes , an arbitrary plane

Unit 3. Projection **[08 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: **[08 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 (up to n = 3), equation of the curve in matrix form (upto n = 3)

Textbook:

1. D. F. Rogers, J. A. Adams, Mathematical elements for Computer graphics, Mc Graw 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 (4^{th} Edition)
2. Schaum Series, Computer Graphics.

MTC-242: Operations Research

Unit 1: Linear Programming Problem I **[12 Lectures]**

1.1 Introduction Definition and Examples
1.2 Problem solving using Graphical method
1.3 Theory of Linear Programming, Slack and surplus variables, Standard form of LPP, Some important definitions, Assumptions in LPP, Limitations of Linear programming, Applications of Linear programming, Advantages of Linear programming Techniques
1.4 Simplex method, Big- M-method

Unit 2: Linear Programming Problem II **[08 Lectures]**

2.1 Special cases of LPP : Alternative solution, Unbounded solution, Infeasible solution
2.2 Duality in Linear Programming, Primal to dual conversion, Examples

Unit 3: Assignment Models **[06 Lectures]**

3.1 Assignmment Model -Introduction
3.2 Hungerian method for Assignment problem

Unit 4: Transportation Models **[10 Lectures]**

4.1 Introduction, Tabular representation

4.2 Methods of IBFS (North-West rule, Matrix-minima, Vogel's Approximation), Algorithms

4.3 The Optimality Test of Transportation Model (MODI method only)

Text Book:-

Operation Research (12 th Edition), by S.D.Sharma.

Unit 1: Chapter 1: Sec. 1.1, 1.3-1, 1.3-2, 1.5, 1.6, 1.8, 1.9, 1.10, 1.11, 1.12,
Chapter 3: Sec. 3.1, 3.2, 3.3, 3. 4, 3.5-4,

Unit 2: Chapter 3: Sec. 3.8-1,3.8-2, Chapter 5: Sec. 5.1-1, 5.2-1,5.3,5.7-1, 5.7-2

Unit 3: Chapter 9: Sec. 9.1, 9.2, 9.4-1, 9.4-2, 9.5, 9.6, 9.7-1, 9.7-2

Unit 4: Chapter 10: 10.1, 10.2, 10.5, 10.8-1,10.9, 10.10

Reference Books:-

1. Operations Research by H. A. Taha
2. Operations Research by R. Panneerselvam, Prentice Hall of India.
3. Principles of Operations Research by H. M. Wagner, Prentice Hall of India.
4. Operations Research by Gupta and Hira.
5. Operation Research by J.K. Sharma

MTC-243: Mathematics Practical: Python Programming Language-II

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 PointList class, Integer point lists, Ordered Point sets, Extreme Points of a PointList, Random sets of Points not in 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

2.3 Polygon : 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 using matplotlib
Practical 7: Study of Graphical aspects of Three dimensional transformation matrix using matplotlib
Practical 8: Study of Graphical aspects of Three dimensional transformation matrix using matplotlib
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, Cambridge University Press, (2005)**
   Sections: 3
2. **Robert Johansson, Introduction to Scientific Computing in Python**
   Section: 1
3. **Jason Brownlee, Basics of Linear Algebra for Machine Learning, Discover the Mathematical Language of Data in Python**
   Sections: 2

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 and Associates Inc.
5. Jim Arlow, Interactive Computational Geometry in Python

Note:

(i) In paper -I , paper-II and paper-III, each course is of 50 marks (35 marks theory and 15 marks internal examination).

(ii) Paper III: Mathematics Practical - MTC-233 and MTC-243 is practical course and

is of 50 marks. Practicals shall be perforemed on computer.

Examination:

A) Pattern of examination: Paper- I, Paper-II and paper-III: Semesterwise

B) Pattern of question papers: For Paper -I and Paper-II

Q 1. Attempt any 05 out of 07 questions each of 01 marks. [05 Marks]
Q 2. Attempt any 02 out of 04 questions each of 05 marks. [10 Marks]
Q 3. Attempt any 02 out of 04 questions each of 05 marks. [10 Marks]
Q 4. Attempt any 02 out of 04 questions each of 10 marks. [10 Marks]

C) Instructions Regarding Practical:

Paper-III:Mathematics Practical:

(i) Mathematics Practical, external examiner shall be appointed by Savitribai Phule Pune University, Pune.
(ii) The minimum duration of parctical examination is 3 hours.
(iii) The semester examination is of 35 marks 15 marks are from internal evaluation (Journal, attendence and viva-voce or internal test etc.)
(iv) The slips for the questions on programming and problem solving using python shall be prepared and provided and these can be used at least for 3 years.

D) Standard of passing:

For Paper- I, Paper-II and Papaer -III: 14 Marks out of 35 and 06 marks out of 15 marks and total should be 20 marks for each course.