UNIVERSITY OF PUNE, PUNE.

SYLLABUS FOR M.TECH.
(INDUSTRIAL MATHEMATICS WITH COMPUTER APPLICATIONS)
(Colleges Affiliated to Pune University)
FOR THE YEAR I
(SEMESTER I, II)
(With effect from June 2013)

Introduction:

University of Pune has decided to change the syllabi of various faculties from June, 2013. 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 consent of teachers of Mathematics from different colleges affiliated to University of Pune has prepared the syllabus of M.TECH. (INDUSTRIAL MATHEMATICS WITH COMPUTER APPLICATIONS**).**

To develop the syllabus the U.G.C. Model curriculum is followed.

Aims:

i) Give the students sufficient knowledge of fundamental principles, methods and a clear perception of the innumerous power of mathematical ideas and tools and knowledge of 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 skill, 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 with mathematical reasoning.
(iii) A student should get adequate exposure to global and local concerns so as to explore many aspects of Mathematical Sciences.
(iv) Students should 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.
(vi) A student should be able to write necessary algorithms and programs in different languages as per the need of the industry.

Eligibility: B.Sc. with any Science stream with Mathematics subject up to second year or equivalent examination.

Structure of the course:

Sr. No.	Paper	Theory	Oral	Internal	Total
1	MIM 101 (Real Analysis)	80 Marks	-	20 Marks	100 Marks
2	MIM 102 (Linear Algebra and Computational Geometry)	80 Marks	-	20 Marks	100 Marks
3	MIM 103 (Discrete Mathematical Structures)	80 Marks		20 Marks	100 Marks
4	MIM 104 (C Programming)	80 Marks		20 Marks	100 Marks
5	MIM 105 (Elements of Information Technology)	80 Marks		20 Marks	100 Marks

All 5 above courses are compulsory.

Medium of Instruction: English

Examination:

A) Pattern of examination: Semester.

B) Standard of passing : 40 Marks out of 100 marks for each papers.
But for qualifying in all the subjects, a student should obtain minimum 32 marks out of 80 in the theory examination and overall total marks for theory and internal should be minimum 40.

C)Pattern of question papers:

Q1. Attempt any 08 out of 10 questions each of 02 marks. [16 Marks]
Q2.
A) Attempt any 01 out of 02 questions each of 06 marks. [6 Marks].
B) Attempt any 02 out of 03 questions each of 05 marks. [10 Marks].
Q.3.
A) Attempt any 01 out of 02 questions each of 06 marks. [6 Marks].
B) Attempt any 02 out of 03 questions each of 05 marks. [10 Marks].
Q4.
A) Attempt any 01 out of 02 questions each of 06 marks. [6 Marks].
B) Attempt any 02 out of 03 questions each of 05 marks. [10 Marks].

Q.5. Attempt any 02 out of 03 questions each of 08 marks. [16 Marks].

OR

Q.5.
A) Attempt any 01 out of 02 questions each of 06 marks. [6 Marks].
B) Attempt any 02 out of 03 questions each of 05 marks. [10 Marks].

D) External Students: Not allowed.

E)Verification/Revaluation: Allowed for all theory papers.

F) Class Improvement scheme is applicable.

Equivalence of subject
For M.Tech 1st year (SEM I & II)

Sr. No.	Subject Code (OLD Course)	Subject Name (OLD Course)	Equivalent Subject Code (NEW Course) (2013-14)	Equivalent Subject Name (NEW Course) (2013-14)
Semester -I				
1.	MIM-101	Real Analysis	MIM-101	Real Analysis
2.	MIM-102	Algebra I	MIM-202	Algebra
3.	MIM-103	Discrete Mathematical Structure		
4.	MIM-104	C Programming	MIM-104	C Programming
5.	MIM-105	Elements of Information Technology	MIM-105	Elements of Information Technology
6.	MIM-106	Lab work	MIM-106	Lab work
Semester -II				
7.	MIM-201	Real and complex Analysis	MIM-201	Complex Analysis
8.	MIM-202	Algebra II	MIM-102	Linear Algebra and Computational Geometry
9.	MIM-203	Discrete Mathematical Structure – II	MIM-103	Discrete Mathematical Structure
10.	MIM-204	Database Fundamentals	MIM-305	Database Fundamentals
11.	MIM-205	Data Structure Using C	MIM-205	Data Structure Using C
12.	MIM-206	Lab work	MIM-206	Lab work

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

M.Sc. Computer Science (with NET /SET as per existing rules) (For core Computer science subjects)

M.Tech .**(**INDUSTRIAL MATHEMATICSWITH COMPUTER APPLICATIONS) (with NET /SET as per existing rules)

M.C.A.(Science Faculty).(with NET /SET as per existing rules) (For core Computer science subjects)

SEMESTER I

MIM- 101 Real Analysis

MIM- 102 Linear Algebra and Computational Geometry

MIM- 103 Discrete Mathematical Structure

MIM- 104 C Programming

MIM- 105 Elements of Information Technology

MIM- 106 Lab work (Assignment List)

SEMESTER II

MIM- 201 Complex Analysis

MIM- 202 Algebra I

MIM- 203 Numerical Analysis

MIM- 204 C++

MIM- 205 Data Structure Using C

MIM- 206 Lab work (Assignment List)

MIM -101 : Real Analysis

1. Metric Spaces and its Topology:

1.1 Metric Spaces Definition and Examples, k-cells, convex sets, open closed ball, properties

1.2 Definition : Neighborhood, limit point, isolated points, closed sets, interior points, open sets, perfect sets bounded sets, dense sets, examples and properties

1.3 Definition: Open cover, compact sets, examples and properties. Theorem of Weierstrass

1.4 Connected sets, definition of separated sets, connected sets and properties

2. Numerical Sequences and series

2.1 Convergent Sequences, Definition and Examples Properties

2.2 Subsequences: Definition and properties

2.3 Cauchy Sequences: Definition, Examples and properties, Definition of complete metric space, examples, Definition of Monotonic Sequences and its properties

2.4 Upper and lower limits, Definition, examples and properties

2.5 Convergence of some special sequences

2.6 Series: Definition, examples and properties, series of non-negative terms, Cauchy's condensation test and examples

2.7 The Number e

2.8 Root and ratio tests, examples

2.9 Power series, Definition, radius of Convergence, examples and properties

2.10 Summation by parts, absolute convergence

3. Continuity:

3.1 Limits of functions: Definition, examples and properties

3.2 Continuous functions, Definition, examples and properties,

3.3 Continuity and Compactness

3.3.1 Bounded Set: Definition

3.3.2 Continuous image of a compact set is compact and related properties

3.3.3 Definition of Uniform Continuity and related properties

3.4 Continuity and Connectedness: continuous image of connected set is connected and related properties

3.5 Discontinuities, Definition, examples

3.6 Monotonic functions, Definition examples and properties

4. Differentiation:

4.1 Derivative of a real function, Definition examples and properties

4.2 Mean Value Theorem

4.3 Continuity of derivatives,

4.4 Taylor's theorem

4.5 Differentiation of a vector valued function

5. Riemann Stieljes Integral:

5.1 Definition and existence of the integral, related properties

5.2 Properties of the integral

5.3 Integration and differentiation

5.4 Integration of vector valued functions

6. Sequences and series of function:

6.1 Discussion of main problem- with examples
6.2 Uniform convergence: Definition and properties
6.3 Uniform convergence: and continuity
6.4 Uniform convergence: and integration
6.5 Uniform convergence: and differentiation

Text Book: Walter Rudin: Principles of Real Analysis, (3rd Edition, Tata McGraw Hill Publication)

Art. 2.15 to 2.42, 2.45 to 2.47, Art. 3.1 to 3.46, Art. 4.1 to 4.18 4.19 (Statement only), 4.22 to 4.28, 4.29 (Statement only), 5.1 to 5.12, 5.15 to 5.19, 6.1 to 6.15, 6.20, to 6.25, Art 7.1 to 7.17.

MIM 102:- Linear Algebra and Computational Geometry

1. **Vector Spaces**
   1.1 Definitions & Examples
   1.2 Simple properties of Vector Spaces
   1.3 Subspaces: Definitions, Examples, Necessary and Sufficient conditions
   1.4 Sum, Intersection of Subspaces
   1.5 Quotient Space
   1.6 Linear Span: Definitions & Properties
   1.7 Linear Dependence & Independence: Definitions, examples & properties
   1.8 Basis and dimension of a vector space, Dimension of subspaces, Dimension of a quotient space
   1.9 Coordinates relative to a basis, coordinate vector, coordinate matrix

2. **Linear Transformations**
   2.1 Definitions, Examples
   2.2 Simple properties
   2.3 Representation of a linear transformation as a matrix, change of basis
   2.4 Rank-Nullity theorem
   2.5 Algebra of linear transformation

3**. Eigen values & Eigenvectors of a Linear Transformation**
   3.1 Definitions and Examples
   3.2 Eigen values & Eigenvectors of a square matrix
   3.3 Properties, Cayley Hamilton theorem
   3.4 Diagonalization

4. **Inner Product Spaces**
   4.1 Definitions & Examples, properties
   4.2 Cauchy-Schwarz inequality
   4.3 Orthonormal vectors, Orthogonal Complements
   4.4 Orthonormal sets and bases
   4.5 Gram Schmidt orthogonalization process

5. **Two-dimensional Transformations**
   5.1 Representation of Points, Transformations and Matrices, Transformation of Points
   5.2 Rotation, Reflection, Scaling, Combined Transformations, Transformation of the Unit Square, Solid Body Transformation,
   5.3 Translations and Homogeneous Coordinates, Rotation About an Arbitrary Point, Reflection Through an Arbitrary Line,
   5.4 Projection - A Geometric Interpretation of Homogeneous Coordinates, Overall Scaling, Points at Infinity, Transformation Conventions.

6. **Three Dimensional Transformations**
   6.1 Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation.
   6.2 Multiple Transformations,
   6.3 Rotations about an Axis Parallel to a coordinate axis, Rotation about an Arbitrary Axis in

Space,

6.4 Reflection through an Arbitrary Plane. Affine and Perspective Geometry,

6.5 Orthographic Projections, Axonometric Projections, Oblique Projections, Perspective Transformations.

6.6 Techniques for generating perspective views, Vanishing points,

7 **Plane Curves**

7.1. Curve representation, non-parametric curves, parametric curves,

7.2. Parametric representation of a circle, parametric representation of an Ellipse, parametric representation of a parabola, parametric representation of a Hyperbola.

7.3. A procedure for using conic sections. The general conic equations.

8 **Space Curves**

8.1 Bezier curves introduction, definition, properties (without proofs),

8.2 Curve fitting (up to n = 3), equation of the curve in matrix form (up to n = 3). B-spline curves- introduction, definition, properties (without proof).

Reference Book:

Linear Algebra By David Lay: Mathematical Elements of Computer Graphics, Second Edition by D.F. Rogers, J. Alan Adams, McGraw-Hill Publishing Company.

MIM 103 Discrete Mathematical Structures

Formal Logic

1.1 Logic:

Introduction, Proposition, Simple proposition, Compound proposition, Truth value, Prepositional Calculus, operators, Conjunction, Disjunction, Conditional statement, Bi conditional statement, converse, contra positive and In-verse,

1.2 Predicates and Quantifiers: Introduction, Universal quantifier, existential quantifier, counter example, negating quantifiers, nested quantifier, order of quantifiers, truth value of quantifier.

1.3 Methods of proof: Introduction, theorem, proof, rules of inference, argument, valid argument, invalid argument, direct method of proof, indirect method of proof, rules of inference for quantified statements.

2. Counting:

The Basic of Counting, the Pigeonhole Principle, Permutations and Combinations, Binomial Coefficient. Inclusion -Exclusion and Applications of Inclusion-Exclusion.

Graph Theory

1. Graph: Definition, Vertex, Edge, Terminal vertices , self loop, incidence, adjacency finite, In finite graphs degree of a vertex. Isolated vertex, pendant vertex, Null graph, Hand shaking Lemma, Regular graph, complete graph, Bipartite graph, Complete bipartite graph. (Theorem 1.1)

2. Isomorphism, Examples, Subgraph.

3. Operations on graphs: Union, Intersection, ring sum, sum of 2 graphs, fusion, Deletion of a vertex (edge), Decomposition of a graph.

4. Connected graph: walk path, circuit, component (Theorem 2.1, 2.2, 2.3)

5. Euler graph: Definition examples, Chinese postman problem, Fleury’s algorithm. Arbitrarily Traceable graph. (Theorem 2.4, 2.6)

6. Trees: Definition, Pendant vertex in a tree, Distance and Centres in a tree. Rooted and binary trees, Spanning trees, rank nullity, Fundamental circuit, Fundamental cutset, vertex connectivity, edge connectivity, spanning tree, weighted graph, Kruskal's algorithm. (Theorem 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 6.7, 3.9, 3.11)

7. Planner graph: Introduction, Kuratowski's two graphs (K5, K3), Euler's theorem, problems (Theorem 5.1, 5.2, 5.6)

8. Matrix Representation: Incidence matrix, adjacency matrix, properties.

9. Directed graphs: Definition, Incident out of a vertex, incident into a vertex, indegree, outdegree, isolated vertex, pendant vertex, Types of digraphs, Simple Asymmetric, Symmetric, complete, Complete symmetric digraph, complete asymmetric digraph, Arborescence, Definition.

10. Graph theoretic algorithms: Dijkstra's algorithm, Warshall Floyd algorithm, Depth first

search on a graph (Theorem 11.5, 11.6).

11. Networks: Flows and Cuts, Network, sink, source, capacity, Flow,
12. Maximal Flow, f-saturated, f- unsaturated. Ford and Fulkerson Algorithm

Section 8.1 and 8.2. Theorem 8.1, Theorem 8.2 (statement only) [Chapter-8 of Graph Theory by John Clark and Allan Holton]

Reference Book:-

1. K.H. Rosen : Discrete Mathematics and its Applications (TATA McGraw-HILL), 5th Edition
Chapter 1 Section 1.1, 1.2, 1.3, 1.4, 1.5, 1.6.
Chapter 4 Section 4.1, 4.2, 4.3, 4.4.
Chapter 6 Section 6.5, 6.6
3. N. Deo : Graph Theory with Applications to Comp. Sc. and Engineering. PHI Publication.
3. John Clark and Allan Holton : Graph Theory.

MIM-104: C Programming

1. Programming languages (1 Lecture)
1.1 Machine language
1.2 Assembly language
1.3 High level languages
1.4 Compilers and Interpreters

2. Introduction to C (1 Lecture)
2.1 History
2.2 Structure of a C program
2.3 Functions as building blocks
2.4 Application Areas
2.5 C Program development life cycle

3. C Tokens (8 Lectures)
3.1 Keywords
3.2 Identifiers
3.3 Variables
3.4 Constants character, integer, oat, string, escape sequences
3.5 Data types built-in and user defined
3.6 Operators and Expressions: Operator types (arithmetic, relational, logical, assignment, bitwise, conditional, other operators), precedence and associativity rules.

4. Input and Output (1 Lecture)
4.1 Character input and output
4.2 String input and output
4.3 Formatted input and output

5. Control Structures (5 Lectures)
5.1 Decision making structures: If, if-else, switch
5.2 Loop Control structures: While, do-while, for
5.3 Nested structures
5.4 break and continue

6. Functions in C (6 Lectures)
6.1 What is a function?
6.2 Advantages of Functions
6.3 Standard library functions
6.4User defined functions: Declaration, definition, function call, parameter passing (by value), return keyword,
6.5 Scope of variables, storage classes
6.6 Recursion

7. Arrays (4 Lectures)
7.1 Array declaration, initialization
7.2 Types one, two and multidimensional
7.3 Passing arrays to functions

8. Pointers (6 Lectures)

8.1 Pointer declaration, initialization
8.2 Dereferencing pointers
8.3 Pointer arithmetic
8.4 Pointer to pointer
8.5 Arrays and pointers
8.6 Functions and pointers passing pointers to functions, function returning pointers, pointer to function
8.7 Dynamic memory allocation

9. Strings (3 Lectures)
9.1 Declaration and initialization
9.2 Standard library functions
9.3 Strings and pointers
9.4 Array of strings.

10. Structures and Unions (4 Lectures)
10.1 Creating structures
10.2 Accessing structure members (dot Operator)
10.3 Array of structures
10.4 Passing structures to functions
10.5 Nested structures
10.6 Pointers and structures
10.7 Unions
10.8 Difference between structures and unions

11. C Preprocessor (2 Lectures)
11.1 Format of Preprocessor directive
11.2 File Inclusion directive
11.3 Macro substitution, nested macro, argument macro
11.4 Conditional compilation

12. Command Line Arguments (1 Lecture)
12.1. Accessing command line arguments

13. File Handling (3 Lectures)
13.1 Streams
13.2 Types of Files
13.3 Operations on files

13.4 Random access to files

References:

1. Kernighan and Ritchie : The C Programming language
2. Forouzan and Gilberg : Structured Programming approach using C, Thomson learning publications
3. Herbert Schildt : Complete C Reference

MIM-105 Elements of Information Technology

1: Introduction

1.1 Concept of Information Technology and its applications
1.2 What is a computer?
1.3 Basic structure of a computer
1.4 Characteristics of computers

2. Data Representation

2.1 Representation of data
2.2 Types of number systems
2.3 Need for binary systems
2.4 Representation of characters
3.4.1 The ASCII code
3.4.2 The EBCDIC code

3: Input/output units

3.1 Introduction to input/output units
3.2 VDU
3.3 Printer
3.3.1 Inkjet
3.3.2 Laser
3.4 A plotter
3.5 Input Methods
3.5.1Magnetic ink character reorganization
3.5.2 OMR
3.5.3 OCR
3.5.4 bar Coding

4: Computer memory organization

4.1 Memory Cell
4.2 Memory Organization
4.3 Physical devices used to construct memory
4.4 magnetic surface reading
4.5 magnetic Hard disk
4.6 CDROM/DVDROM

5: Processor

5.1 CPU
5.2 CU
5.3 ALU
5.3.1 Instruction Set
5.3.2 Different types of registers used in CPU (MAR, MBR, PC, A, I,I/O)
5.6 Processors speed
5.7 Types of processors
5.7.1 CISC

5.7.2 RISC
5.7.3 EPIC
5.7.4 Multi core processor

6: Main Memory
- 6.1 storage evaluation criteria
- 6.2 Why More bits?
- 6.3 Fix and variable word length memory
- 6.4 Types of memory chip
- 6.5 cache memory

7: Introduction of Computer architecture
- 7.1 interconnection of units
- 7.2 processor to memory communication
- 7.3 I/O to processor communication

Reference book:-

1. Fundamentals of computer by V Rajaraman, fourth edition, Prentice-Hall of India Pvt, Ltd
2. Fundamentals of computer by P K Sinha, sixth edition, BPB publication

---

MIM 106 Lab Work

Assignments List

1. Write Simple C Programs (Using operators only) Area of Triangle, Circle, Simple and Compound Interest, Celsius to Fahrenheit

2. Roots of Quadratic Equations.

3. Write a C program to accept a decimal number and convert it to Binary, Octal and Hexadecimal equivalent

4. Write a menu driven program to check if a given number is perfect / prime/ palindrome.

5. Computing sin x and cos x series.

6. Write a menu driven program to multiply, subtract and find transpose of the given matrices.

7. Display the single digit sum of the given number recursively.

8. String Manipulations using pointers

a. String length

b. Display substring from a given position and up to the given number of characters

c. Concatenate two strings

d. Uppercase to Lowercase

e. String compare Without using Standard Library functions

9. Write a C program to Insert and Delete an element in an array using Pointers.

10. Write a C program to accept information of n students having elds: Rollno, Name, Class, Grade(A/B/C). Display the information of those students who have A grade.

11. Write a program to add 2 matrices of size m x n using dynamic memory allocation.

12 Write a C program to create an **le** and count the number of words, lines and characters in the **le**.

13. Write a C program to encrypt /decrypt the contents of an **le** using command line arguments.

MIM 201 Complex Analysis

1 Complex Numbers

Sums and Products
Basic Algebraic Properties
Further Properties
Vectors and Moduli
Complex Conjugates
Exponential Form
Products and Powers in Exponential Form
Arguments of Products and Quotients
Roots of Complex Numbers
Examples
Regions in the Complex Plane

2 Analytic Functions

Functions of a Complex Variable
Mappings
Limits
Theorems on Limits
Limits Involving the Point at Infinity
Continuity
Derivatives
Differentiation Formulas
Cauchy–Riemann Equations
Sufficient Conditions for Differentiability
Analytic Functions
Harmonic Functions
Uniquely Determined Analytic Functions

3 Elementary Functions

Trigonometric Functions
Hyperbolic Functions
Inverse Trigonometric and Hyperbolic Functions

4 Integrals

Derivatives of Functions w(t)
Definite Integrals of Functions w(t)
Contours
Contour Integrals
Some Examples
Upper Bounds for Moduli of Contour Integrals
Anti-derivatives
Proof of the Theorem
Cauchy–Goursat Theorem (without proof)
Simply Connected Domains (only definitions)
Cauchy Integral Formula
An Extension of the Cauchy Integral Formula

Morera Theorem
Some Consequences of the Extension
Liouville's Theorem and the Fundamental Theorem of Algebra
Maximum Modulus Principle

5 Series

Convergence of Sequences
Convergence of Series
Taylor Series
Proof of Taylor's Theorem
Examples
Laurent Series
Proof of Laurent's Theorem
Examples

6 Residues and Poles

Isolated Singular Points
Residues
Cauchy's Residue Theorem
Residue at Infinity
The Three Types of Isolated Singular Points
Residues at Poles
Examples
Zeros of Analytic Functions
Zeros and Poles
Behavior of Functions Near Isolated Singular Points viii contents

7 Applications of Residues

Evaluation of Improper Integrals

Reference Books

Churchill & Brown: Complex variables and its Applications(Eight Edition, Tata McGrawHill)

S. Ponnuswamy: Foundations of Complex Analysis (Narosa Publication, Fourth Reprint)

MIM 202-ALGEBRA – I

1:- Groups

1.1 Definitions and Examples
1.2 Simple properties of Groups based on axioms
1.3 Order of an Element – Definition, properties and Examples

2:- Subgroups
2.1 Subgroups
2.2 Definition and Examples
2.3. NAS conditions for a Subgroups
2.4. Properties of Subgroups

3:- Cyclic groups
3.1 Cyclic groups
3.2 Definitions and Examples
3.3 Properties

4:- Permutation groups
4.1 Definition and Examples; (Permutation as composition of function)
4.2 Definition of S_n and discussion of S_3 in detail
4.3 Cycles, Transpositions
4.4 Every Permutation is a product of disjoint cycles (without proof)
4.5 Even and odd permutations, order of a permutation
4.6 Alternating group A_n.
4.7 $\frac{S_n}{A_n} \sim \{-1,1\}$

5:- Homomorphism & Isomorphism

5.1 Definitions and Examples
5.2 Simple Properties
5.3 Isomorphism - Definition and Examples
5.4 Fundamental theorem of homomorphism & application
5.5 Cayley's theorem

6:- Cosets & Langrange theorem
6.1 Cosets - Definition, Examples & Properties
6.2 Lagrange's theorem and its corollaries

7:- Normal Subgroups

7.1 Definition and Examples
7.2 Properties of Normal Subgroups
7.3 Simple Groups, An is Simple for n = 5 (without proof)
7.4 Factor Group, Definition and Examples.
7.5 Properties of Factor groups

8:- Sylow's theorems

8.1. Class Equations
8.1.1. Conjugate of an element-Definition & Examples
8.1.2. Conjugacy relation is and equivalence relation, Conjugacy Class
8.1.3. Normaliser, Centraliser, Center of a group.
8.1.4. Class equation
8.1.5. 'a' belongs to Z(G) iff N(a) = G
8.1.6. Centre of a p-group is nontrivial.
8.1.7. Every group of order p-square is abelian.
8.2. Cauchy's theorem (Statements only)
8.3. Sylow's theorems (without proofs) - only problems.

9:- Rings

9.1. Definitions & Examples
9.2. Simple Properties of Rings.
9.3. Commutative ring, ring with unity, integral domain, field, skew field definitions, examples and interrelationships between them.
9.4. Subrings - Definition, Examples, Properties.
9.5. Characteristic of an integral domain.

10:- Ideals & Factor Rings

10.1. Definitions & Examples
10.2. Properties of ideals
10.3. Prime Ideals, Maximal Ideals.
10.4 Quotient rings

11:- Homomorphism & Isomorphism of rings
11.1. Definitions & Examples
11.2. Properties of ring homomorphisms
11.3. Fundamental theorem of ring homomorphisms & its applications.

12:- Polynomial Rings

12.1. Definitions & Examples
12.2. Properties
12.3. Polynomial ring F[x] over a field F.
12.4. F[x] is a Euclidean Ring.
12.5. Irreducible polynomials over a field

Text Books:
1) Contemporary Abstract Algebra by Joseph Gallian (Fourth Edition, Narosa Publication)
2) J.B. Fraleigh – Abstract Algebra, 5th edition
3) S. Gopala Krishnan, University Algebra

MIM 203 Numerical Analysis

1. Iterative solutions of Nonlinear Equations: Bisection Method, Fixed-Point iteration, Newton's method, Secant method, Acceleration of convergence, Newton's method for two nonlinear equations, Polynomial equation methods.

2. Polynomial Interpolation: The Lagrange interpolation polynomial, Divided difference interpolation, Aitken's Algorithm, Finite difference formulas, Choice of nodes and non convergence of polynomial interpolation.

3. Systems of Linear equations, Gauss elimination with partial pivoting, Error analysis, Matrix factorization methods (Doolittle reduction, Crout reduction), Iterative refinement, Iterative techniques, Gauss-Seidel iteration

4. Numerical Calculus: Numerical differentiation, Forward difference Quotient, Central difference quotient, Interpolatory quadrature (order of methods), Newton-Cotes methods, Error estimates for trapezoidal rule and Simpson's rule.

5. Numerical solution of Differential Equations : Euler's method, Analysis of Euler's method, Order of Euler's method, Runge-Kutta method, One step modified and midpoint methods, Runge-Kutta methods for systems of equations.

6. The Eigen value problem: Power method, Eigenvalues of symmetric matrices, Jacobi method,.

Reference Books:

1. John H. Mathews : Numerical Methods for Mathematics, Science and Engineering (Prentice-Hall) 2nd Edition. Sections : 1.3, 2.1 to 2.7, 3.4 to 3.7, 4.2 to 4.4, 6.1 to 6.2, 7.1 to 7.4, 9.2 to 9.7, 11.1 to 11.2

2. K. E. Atkinson : An introduction to Numerical Analysis (John Wiley Sons).

3. S. Sastry: Numerical Analysis

MIM 204- OBJECT ORIENTED PROGRAMMING WITH C++

1. 1. Introduction

1. Concept, Benefits and Application of OOP
2. Structure of C++ Programming
3. Tokens, expressions and control structures, keywords, Identifiers, data types & operators in C++.

2. Functions in C++

1. Function Prototyping
2. Call by value, Call by reference
3. Return by reference
4. Inline Functions
5. Default arguments
6. Function overloading
7. Friend and Virtual functions

3. Class and Objects

1. Introduction to classes and creating objects
2. Friend classes
3. Static class members
4. Nested classes
5. Local classes
6. Memory allocation for objects
7. Array to objects
8. Objects as function arguments
9. Constructors and destructors

4. Inheritance, Pointers, Virtual functions and Polymorphism

1. Single, Multilevel, Multiple, Hierarchical and Hybrid Inheritance
2. Virtual base classes
3. Abstract classes
4. Pointer to objects, pointer to derived class
5. Operator overloading

5. I/O System Basics

1. C++ streams, C++ stream classes
2. Formatted I/O, Unformatted I/O operations
3. Overloading <<and >>, creating own inserters
4. Extractor and manipulator functions

6. File I/O and Array Based I/O

1. Classes for file stream operations
2. Opening and closing of file, detecting EOF
3. Random access, I/O status
4. Array based class, Array based I/O stream, random access within the array
5. Dynamic arrays

6. Custom extractors and inserters

7. Templates and Exception Handling

1. Generic functions
2. Templates, class Templates, functions Templates
3. Member function templates, template arguments
4. Exception handling function templates, template arguments
5. Exception handling fundamentals, exception handling options
6. Catching all exceptions, restricting exceptions and rethrowing exceptions.

Reference books:

1. Object Oriented Programming with C++ - E. BALAGURUSWAMY
2. C++ the Complete Reference - HERBERT SCHILDT
3. A Treatise on Object Oriented Prog. Using C++ - B. CHANDRA
4. Serial communication - A C++ developers guide - NELSON

MIM 205: Data Structures using C

1. Introduction

1.1 Data, Data types, Abstract Data Type

1.2 Data Structures

1.3 Linear & Nonlinear data structures

1.4 Algorithm Analysis

2. Arrays

2.1 Arrays as ADT

2.2 1-D, 2-D, Multidimensional Arrays

2.3 Applications

2.4 Polynomial Representation in one variable (Using array of structure)

3. Stacks

3.1 ADT, Push and Pop operations

3.2 Stack implementation using array

3.3 Stack applications

3.3.1 Infix to postfix conversion of expression

3.3.2 Expression evaluation

3.3.3 Recursion

4. Queues

ADT, Insert and Delete operations

Queue implementation using array

Types –Priority Queue, Circular queue, Dequeue

4.4 Queue applications:

4.4.1 CPU Scheduling Algorithms

FCFS, Round Robin algorithm

5. Linked List Concept, Operations: Insert, Delete, Traversal

Static implementation using arrays

Dynamic implementation

Doubly Linked list

Circular list

Linked list applications:

Stacks and Queues as Linked Lists

Merging of two linked lists

6. Trees

6.1 Terminology and Concepts

6.2 Binary Tree Representation

6.2.1 Static implementation using arrays

6.2.2 Linked representation

6.2.3 Binary Search Tree

6.2.4 Operations on Binary search tree -Insert, Delete

6.2.5 Tree Traversals

6.3 Representing General Trees as binary tree

7. Searching and Sorting

Searching ,Concept and need ,Techniques

Linear search, Binary search, Indexed sequential search

Sorting ,Concept and Need ,Performance criteria

Techniques

Comparison Based-(Bubble, Quick, Insertion, Merge)

Linear order sorting-(Counting)

8. Graphs

8.1 Terminology and concepts

8.2 Graph Representation: Adjacency matrix, Adjacency list, Adjacency multilist

8.3 Traversals: Depth first and Breadth first

Reference Books:

1. Tanenbaum, Langsam, Augenstein : “Data structures using C”, PHI 1994

2. D. Samanta : “Classic Data Structures”, PHI 2002

MIM 206: Lab Work

Assignment based on object oriented programming with C++ and Data structure using C.
