

E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)

Approved by AICTE, New Delhi | Affiliated to Anna University, Chennai
Accredited by NAAC with 'A' Grade | Accredited by NBA (CSE, EEE, MECH)
NAGAPATTINAM – 611 002



B.E. COMPUTER SCIENCE AND ENGINEERING

Third Year – Fifth Semester

Course Code	Course Name	L	T	P	C	Maximum Marks		
						CA	ES	Total
Theory Course								
1701MA501	Discrete Mathematics	3	2	0	4	40	60	100
1702CS501	Object Oriented Analysis and Design	3	0	0	3	40	60	100
1702CS502	Theory of Computation	3	2	0	4	40	60	100
1702CS503	Computer Graphics and Multimedia	3	0	0	3	40	60	100
1703CS002	Elective I-Software Testing	3	0	0	3	40	60	100
1703CS006	Elective II- Mobile Computing	3	0	0	3	40	60	100
Laboratory Course								
1702CS551	Computer Graphics and Multimedia Lab	0	0	2	1	50	50	100
1704CS552	CASE Tools Lab (Mini Project I)	0	0	2	1	50	50	100
1704GE553	Life Skills: Aptitude I	0	0	2	1	100	-	100
1704GE554	Technical Seminar	0	0	2	1	100	-	100

L – Lecture | T – Tutorial | P – Practical | CA – Continuous Assessment | ES – End Semester

1701MA501	DISCRETE MATHEMATICS	L	T	P	C
		3	2	0	4

PREREQUISITE:

Engineering Mathematics I
Engineering Mathematics II
Engineering Mathematics III

COURSE OBJECTIVES:

1. Develop ability to analyze the mathematical Logic
2. Explore the concepts of counting principles and graph theory
3. To familiarize the students in understanding algebraic systems and relations

UNIT I LOGIC AND PROOFS

12 Hours

Propositional Logic – Propositional equivalences-Predicates and quantifiers-Nested Quantifiers- Rules of inference-introduction to Proofs-Proof Methods and strategy

UNIT II COMBINATORICS

12 Hours

Mathematical inductions-Strong induction and well ordering-The basics of counting-The pigeonhole principle – Permutations and combinations-Recurrence relations-Solving Linear recurrence relations-generating functions.

Unit III GRAPHS

12 Hours

Graphs and graph models-Graph terminology and special types of graphs-Representing graphs and graph isomorphism-connectivity-Euler and Hamilton paths

UNIT IV ALGEBRAIC STRUCTURES

12 Hours

Algebraic systems-Semi groups and monoids-Groups-Subgroups and homomorphisms-Cosets and Lagrange's theorem- Ring & Fields (Definitions and examples)

UNIT V LATTICES AND BOOLEAN ALGEBRA

12 Hours

Partial ordering-Posets-Lattices as Posets- Properties of lattices-Lattices as Algebraic systems-Sublattices – direct product and Homomorphism-Some Special lattices- Boolean Algebra

TOTAL: 60 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

- 1 Modeling Computation and Languages
- 2 Matrix representation of Graphs

REFERENCES:

1. Ralph. P. Grimaldi, "Discrete and Combinatorial Mathematics: An Applied Introduction", Fourth Edition, Pearson Education Asia, Delhi, (2002).
2. Kenneth H. Rosen, "Discrete Mathematics and its Applications", Special Indian edition, Tata McGraw-Hill Pub. Co. Ltd., New Delhi, (2007).
3. Trembly J. P. and Manohar R., "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw-Hill Pub. Co. Ltd., New Delhi, 30th Re-print (2007).
4. Thomas Koshy, "Discrete Mathematics with Applications", Elsevier Publications, (2006).
5. Seymour Lipschutz and Mark Lipson, "Discrete Mathematics", Schaum's Outlines, Tata McGraw-Hill Pub. Co. Ltd., New Delhi, Second edition, (2007).
6. nptel.ac.in/courses/111105035, www.nptelvideos.in/2012/11/Mathematics.html
7. www.learnerstv.com/Free-maths-video-lectures-ltv348-page1.html

1702CS501	OBJECT ORIENTED ANALYSIS AND DESIGN	L	T	P	C
		3	0	0	3

PREREQUISITE:

1. Software Engineering
2. Programming Concepts

COURSE OBJECTIVES:

1. To develop background knowledge as well as core expertise in object oriented System.
2. To provide the importance of the software design process.
3. Learn the basics of OO analysis and design skills the UML design diagrams.

UNIT I UML DIAGRAMS 9 Hours

Introduction to OOAD – Unified Process - UML diagrams – Use Case – Class Diagrams– Interaction Diagrams – State Diagrams – Activity Diagrams – Package, component and Deployment Diagrams

UNIT II DESIGN PATTERNS 9 Hours

Object oriented design methodology – Common base class - **GRASP**: Designing objects with responsibilities – Patterns– Creator – Information expert – Low coupling –Controller – High cohesion – Designing for visibility - Applying GoF design patterns – Adapter – Singleton – Factory – Strategy – Composite - Facade and observer patterns

UNIT III APPLYING DESIGN PATTERNS 9 Hours

System sequence diagrams - Relationship between sequence diagrams and use cases Logical architecture and UML package diagram – UML class diagrams - UML interaction diagrams - Finding conceptual classes and description classes – Associations – Attributes – Domain model refinement – Finding conceptual class Hierarchies - Aggregation and Composition.

UNIT IV IMPLEMENTATION AND APPLICATION 9 Hours

Mapping design to code – Forward Engineering – Reverse Engineering - Test driven development – Refactoring – UML tools and UML as blueprint - UML state machine diagrams and modeling - UML deployment and component diagrams

UNIT V CODING AND TESTING 9 Hours

Mapping design to code – Testing: Issues in OO Testing – Class Testing – OO Integration Testing – GUI Testing – OO System Testing.

TOTAL: 45 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

1. Advanced Design Patterns.
2. Developing SRS Documents.

REFERENCES:

1. Craig Larman, "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development", Third Edition, Pearson Education, 2015.
2. MichealBlaha, James Rumbaugh, "Object-Oriented Modeling and Design with UML", Second Edition, Prentice Hall of India Private Limited, 2007
3. Mike O'Docherty, "Object-Oriented Analysis & Design: Understanding System Development with UML 2.0", John Wiley & Sons, 2005.
4. James W- Cooper, Addison-Wesley, "Java Design Patterns – A Tutorial", 2000.
5. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design patterns: Elements of Reusable object-oriented software", Addison-Wesley, 2009
6. <http://nptel.ac.in/courses/106105153/>

1702CS502	THEORY OF COMPUTATION	L	T	P	C
		3	2	0	4

PREREQUISITE:

1. C Programming.
2. Engineering Mathematics III

COURSE OBJECTIVES:

1. Introduce students to the mathematical foundations of computation including automata theory; the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability.
2. Enhance/develop students' ability to understand and conduct mathematical proofs for computation and algorithms.
3. Be able to construct Turing machines and Post machines.
4. Understand the notions of decidability and undecidability of problems, Halting problem.

UNIT I FINITE AUTOMATA

12 Hours

Introduction- Basic Mathematical Notation and techniques- Finite State systems –Basic Definitions – Finite Automaton – DFA & NFA – Finite Automaton with ϵ -moves – Regular Languages- Regular Expression – Equivalence of NFA and DFA – Equivalence of NFA's with and without ϵ -moves – Equivalence of finite Automaton and regular expressions –Minimization of DFA- - Pumping Lemma for Regular sets – Problems based on Pumping Lemma.

UNIT II GRAMMARS

12 Hours

Grammar Introduction– Types of Grammar - Context Free Grammars and Languages– Derivations and Languages – Ambiguity- Relationship between derivation and derivation trees – Simplification of CFG – Elimination of Useless symbols - Unit productions - Null productions – Greiback Normal form – Chomsky normal form – Problems related to CNF and GNF.

UNIT III PUSHDOWN AUTOMATA

12 Hours

Pushdown Automata- Definitions – Moves – Instantaneous descriptions – Deterministic pushdown automata- Equivalence of Pushdown automata and CFL - pumping lemma for CFL – problems based on pumping Lemma.

UNIT IV TURING MACHINE

12 Hours

Turing Machines- Introduction – Formal definition of Turing machines – Instantaneous descriptions- Turing Machine as Acceptors – Turing Machine as Transducers Computable Languages and functions – Turing Machine constructions – Modifications of Turing Machines.

UNIT V COMPUTATIONAL COMPLEXITY

12 Hours

Undecidability- Basic definitions- Decidable and undecidable problems - Properties of Recursive and Recursively enumerable languages – Introduction to Computational Complexity: Definitions-Time and Space complexity of TMs – complexity classes – introduction to NP-Hardness and NP-Completeness

TOTAL: 60 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

1. Introduction to Infinite Automata Theory
2. Advanced theory of computation.

REFERENCES:

1. Hopcroft J.E., Motwani R. and Ullman J.D, "Introduction to Automata Theory, Languages and Computations", Third Edition, Pearson Education, 2008.
2. Adesh K. Pandey "An introduction to automata theory and formal languages", S.K. Kataria & Sons
3. John C Martin, Introduction to Languages and the Theory of Computation, TMH, 2007
4. Michael Sipser, Introduction To Theory of Computation, Cengage Publishers, 2013
5. Dexter C. Kozen, Automata and Computability, Springer 1999
6. <http://nptel.ac.in/downloads/106106049/>

1702CS503	COMPUTER GRAPHICS AND MULTIMEDIA	L	T	P	C
		3	0	0	3

PREREQUISITE:

1. Programming in C
2. Programming Paradigms
3. Design and analysis of Algorithms

COURSE OBJECTIVES:

1. To know the Basic devices of graphics
2. To know the algorithm for displaying two dimensional output primitives for raster graphics system
3. To know the basic concepts of how to represent the 3D objects and colour models
4. To know the basic concepts of multimedia and advanced multimedia system concepts

UNIT I BASIC OF COMPUTER GRAPHICS 7 Hours

Basic of Computer Graphics, Applications of computer graphics, Display devices, Random and Raster scan systems, Graphics input devices, Graphics software and standards

UNIT II GRAPHICS PRIMITIVES 9 Hours

Points, lines, circles and ellipses as primitives, scan conversion algorithms for primitives, Fill area primitives including scan-line polygon filling, inside-outside test, boundary and flood-fill, character generation, line attributes, area-fill attributes, character attributes.

UNIT III 2D TRANSFORMATION AND VIEWING 10 Hours

Transformations (translation, rotation, scaling), matrix representation, homogeneous coordinates, composite transformations, reflection and shearing, viewing pipeline and coordinates system, window-to-viewport transformation, clipping including point clipping, line clipping (cohen-sutherland, liang- bersky, NLN), polygon clipping, Color models: properties of light, XYZ, RGB, YIQ and CMY color models

UNIT IV 3D CONCEPTS, TRANSFORMATION AND VIEWING 10 Hours

3D display methods, polygon surfaces, tables, equations, meshes, curved lies and surfaces, quadric surfaces, spline representation, cubic spline interpolation methods, Bazier curves and surfaces, B-spline curves and surfaces, 3D scaling, rotation and translation, composite transformation, viewing pipeline and coordinates, parallel and perspective transformation

UNIT V MULTIMEDIA 9 Hours

Introduction to Multimedia: Concepts, uses of multimedia, hypertext and hypermedia; Image, video and audio standards. Audio: digital audio, MIDI, processing sound, sampling, compression. Video: MPEG compression standards, compression through spatial and temporal redundancy, inter-frame and intraframe compression. Animation: types, techniques, key frame animation, utility, morphing. Virtual Reality concepts.

TOTAL: 45 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

1. Visible surface detection concepts
2. Back-face detection, depth buffer method, illumination, light sources, illumination methods

REFERENCES:

1. Computer Graphics, D.Hearn And P.Baker - Pearson Education - C Version
2. Computer Graphics, with OpenGL Hearn and Baker, - Pearson
3. Computer Graphics, Sinha&Udai, - TMH
4. Computer Graphics, Foley and van Dam - Person Education
5. Mukherjee, Fundamentals of Computer graphics & Multimedia, PHI

1703CS502

SOFTWARE TESTING

L	T	P	C
3	0	0	3

Course Objectives:

1. To study the various test design strategies.
2. To understand the levels of testing and defect classes.
3. To learn the testing and debugging policies.
4. To study about quality assurance plan.

Unit I INTRODUCTION 9 Hours

Software testing fundamentals– Minimizing Risks – Writing a policy – Building a structured approach – Developing a test strategy – Building the software testing process – Software testing guidelines –Customizing the software testing process.

Unit II ORGANIZATION AND DEVELOPMENT OF TESTING APPROACH 9 Hours

Overview of the software testing process – Organizing for testing – Developing Test plan – Profile the software project – Understand project risk – Testing technique – Unit testing and analysis – Build and Inspect Test Plan.

Unit III VERIFICATION AND VALIDATION 9 Hours

Verification Testing – Requirement phase Testing – Design phase testing – Programming phase testing – Test during requirement, Design and Programming Phase – Guidelines – Validation Testing – Build test data – Execute Results – Record Test Results.

Unit IV IMPLEMENTATION 9 Hours

Acceptance Testing and Operational Testing – Acceptance Testing – Define, Develop and Execute – Preoperational Testing – Test and Monitor – Post-Operational Testing – Develop and Test – Post Implementation Analysis – Workbenches – Procedures.

Unit V SOFTWARE QUALITY CONSIDERATIONS 9 Hours

Quality management – Quality assurance plan- SCM support functions- SCM Tools- Establishing standards – Guidelines- Basic inspection principles- Principles of software defect prevention- Process changes for defectPrevention –Defect prevention considerations.

Total: 45 Hours

Further Reading:

1. The SEI process capability maturity model
2. Reliability measures

References:

1. William E Perry, *Effective Methods for Software Testing*, John Wiley & Sons, USA, 2008
2. Watts S. Humphrey, *Managing the software process*, Addison Wesley, 201
3. Ian Sommerville, *Software Engineering*, Addison-Wesley, 8th edition, 2006.
4. Steve McConnell, *Code Complete*, Second Edition, Microsoft Press.
5. Richard E. Fairley, *Software Engineering Concepts*, McGraw- Hill, 1985

1703CS506

MOBILE COMPUTING

L	T	P	C
3	0	0	3

Course Objectives:

1. To know the components and structure of mobile application development frameworks for Android and Windows OS based Mobiles.
2. To learn how to work with various mobile application development frameworks.
3. To be familiar with the capabilities and limitations of mobile devices.

Unit I OVERVIEW and GSM ARCHITECTURE 8 Hours

Mobile Computing, Mobile Computing Architecture, Mobile Devices, Mobile System Networks

GSM and Other 2G Architectures: GSM, Radio Interfaces of GSM, Protocols of GSM, Localization, Call Handling, Handover, Security, New Data Services, General Packet Radio Service.

Unit II WIRELESS MEDIUM ACCESS CONTROL, CDMA, 3G, AND 4G COMMUNICATION 9 Hours

Multiplexing, Controlling the Medium Access, , Frequency Hopping Spread Spectrum , Coding Methods , Code Division Multiple Access, IMT -2000 3G Wireless Communication Standard, WCDMA 3G Communication Standards, I-mode, OFDM, Long-term Evolution, WiMaxRel 1.0 IEEE 802.16e, 4G Networks.

Unit III MOBILE IP NETWORK LAYER & MOBILE TRANSPORT LAYER 9 Hours

Mobile IP Network Layer: IP and Mobile IP Network Layer, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, Dynamic Host Configuration Protocol. Mobile Transport Layer: Conventional TCP/IP Transport Layer Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Methods for Mobile TCP –layer Transmission, TCP over 2.5G/3G Mobile Networks.

Unit IV DATABASES AND DATA DISSEMINATION AND BROADCASTING SYSTEMS 8 Hours

Databases: Database Hoarding Techniques, Data Caching, Client- Server Computing and Adaptation, Transaction Models, Query Processing, Data Recovery Process, Issues Relating to Quality Of Service. Data Dissemination and Broadcasting Systems: Communication Asymmetry, Classification of Data-Delivery Mechanisms, Data Dissemination Broadcast Models, Selective Tuning and Indexing Techniques.

Unit V MOBILE SYNCHRONIZATION AND MOBILE DEVICES 11 Hours

Mobile Synchronization in Mobile Computing Systems: Synchronization, Synchronization Software for Mobile Devices, Synchronization Protocols, Sync- Synchronization Language for Mobile Computing, Sync4J Synchronized Multimedia Markup Language (SMIL). Mobile Devices: Server and Management-Mobile Agent, Application Server, Gateways, Portals, Service Discovery, Device Management, Mobile File Systems, Security.

Total: 45 Hours

REFERENCE BOOKS:

1. Jochen H. Schiller, "Mobile Communications," Pearson Education, Second Edition, 2004
2. Asoke Talukder, Roopa Yavagal, "Mobile Computing," Tata McGraw Hill, Second Edition 2010.

1702CS551	COMPUTER GRAPHICS AND MULTIMEDIA LAB	L	T	P	C
		0	0	2	1

PREREQUISITE :

1. Programming in C
2. Programming Paradigms

COURSE OBJECTIVES:

1. To explore the various multimedia editing tools like Photoshop/EQV, audacity, Garageband, iMovie and OpenCV
2. To outline the structure media processing tools

LIST OF EXPERIMENTS:

1. To study the various graphics commands in C language.
2. Develop the DDA Line drawing algorithm using C language
3. Develop the Bresenham's Line drawing algorithm using C language
4. Develop the Bresenham's Circle drawing algorithm using C language
5. Develop the C program for to display different types of lines
6. Perform the following 2D Transformation operation Translation , Rotation and Scaling
7. Perform the Line Clipping Algorithm
8. Perform the Polygone clipping algorithm
9. Procedure to draw the fan blades and to give proper animation using flash
10. Procedure to simulate a ball hitting another ball using flash
11. Procedure to prepare a cover page for the book in your subject area. plan your own design using Photoshop.
12. Design a banner using coral Draw

TOTAL:45 HOURS

ADDITIONAL EXPERIMENTS/ INNOVATIVE EXPERIMENTS:

1. Basic Graphics games in C language

REFERENCES:

1. spoken-tutorial.org
2. <http://www.cosc.canterbury.ac.nz/people/mukundan/cogr/DDA.html>
3. www.doc.ic.ac.uk/~dfg/graphics/GraphicsSlides01.pdf
4. Donald Hearn, Pauline Baker M., "Computer Graphics", 2nd Edition, Prentice Hall, 1994.
5. Tay Vaughan , "Multimedia", 5th Edition, Tata McGraw Hill, 2001

1704CS552	CASE TOOLS LAB (MINI PROJECT I)	L	T	P	C
		0	0	2	1

PREREQUISITE :

1. Software Engineering
2. Programming Concepts

COURSE OBJECTIVES:

1. To highlight the importance of object-oriented analysis and design and its limitations.
2. To show how we apply the process of object-oriented analysis and design to software development.
3. To provide the necessary knowledge and skills in using object oriented CASE tools.

LIST OF EXPERIMENTS:

1. To develop a problem statement and Statement of Work.
2. Develop an IEEE standard SRS document. Also develop risk management and project plan
3. Identify Use Cases and develop the Use Case model.
4. Identify the business activities and develop an UML Activity diagram.
5. Identify the conceptual classes and develop a domain model with UML Class diagram.
6. Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams
7. Draw the State Chart diagram.
8. Identify the User Interface, Domain objects, and Technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation and patterns
9. Draw Component and Deployment diagrams.
10. Practice forward engineering and reverse engineering

TOTAL : 45 HOURS

ADDITIONAL EXPERIMENTS/ INNOVATIVE EXPERIMENTS:

1. Exam Registration.
2. Library Management System.

REFERENCES:

1. Manual Prepared by the course instructor
2. Craig Larman, "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development", Third Edition, Pearson Education, 2015.
3. <http://www.seminaronly.com/computer%20science/itwin-seminar-report-ppt-pdf.php>

1704GE551	LIFE SKILLS: APTITUDE – 1	L	T	P	C
		0	0	2	1

Course Objectives:

1. The students should be made to:
2. To brush up problem solving skill and to improve intellectual skill of the students
3. To be able to critically evaluate various real life situations by resorting to Analysis Of key issues and factors
4. To be able to demonstrate various principles involved in solving mathematical problems and thereby reducing the time taken for performing job functions.
5. To enhance analytical ability of students
6. To augment logical and critical thinking of Student

UNIT I INTRODUCTION TO NUMBER SYSTEM, BASIC SHORTCUTS OF ADDITION, MULTIPLICATION, DIVISION 6 Hours

Classification of numbers – Types of Numbers - Divisibility rules - Finding the units digit - Finding remainders in divisions involving higher powers - LCM and HCF Models - Fractions and Digits – Square, Square roots – Cube, Cube roots – Shortcuts of addition, multiplication, Division.

UNIT II RATIO AND PROPORTION, AVERAGES 6 Hours

Definition of Ratio - Properties of Ratios - Comparison of Ratios - Problems on Ratios - Compound Ratio - Problems on Proportion, Mean proportional and Continued Proportion Definition of Average - Rules of Average - Problems on Average - Problems on Weighted Average - Finding average using assumed mean method.

UNIT III PERCENTAGES, PROFIT AND LOSS 6 Hours

Introduction Percentage - Converting a percentage into decimals - Converting a Decimal into a percentage - Percentage equivalent of fractions - Problems on percentages - Problems on Profit and Loss percentage- Relation between Cost Price and Selling price - Discount and Marked Price - Two different articles sold at same Cost Price - Two different articles sold at same Selling Price - Gain% / Loss% on Selling Price.

UNIT IV CODING AND DECODING, DIRECTION SENSE 6 Hours

Coding using same set of letters - Coding using different set of letters - Coding into a number - Problems on R-model - Solving problems by drawing the paths - Finding the net distance travelled - Finding the direction - Problems on clocks - Problems on shadows - Problems on direction sense using symbols and notations.

UNIT V NUMBER AND LETTER SERIES NUMBER AND LETTER ANALOGIES, ODD MAN OUT 6 Hours

Difference series - Product series - Squares series - Cubes series - Alternate series - Combination series - Miscellaneous series - Place values of letters - Definition of Analogy - Problems on number analogy - Problems on letter analogy - Problems on verbal analogy - Problems on number Odd man out - Problems on letter Odd man out - Problems on verbal Odd man out

TOTAL 30 Hours

REFERENCES:

1. Arun Sharma, 'How to Prepare for Quantitative Aptitude for the CAT', 7th edition, McGraw Hills publication, 2016.
2. Arun Sharma, 'How to Prepare for Logical Reasoning for CAT', 4th edition, McGraw Hills publication, 2017.
3. R S Agarwal, 'A modern approach to Logical reasoning', revised edition, S.Chand publication, 2017.
4. R S Agarwal, 'Quantitative Aptitude for Competitive Examinations', revised edition, S.Chand publication, 2017.
5. Rajesh Verma, "Fast Track Objective Arithmetic", 3rd edition, Arihant publication, 2018.
6. B.S. Sijwali and InduSijwali, "A New Approach to REASONING Verbal & Non-Verbal", 2nd edition, Arihant publication, 2014.

ASSESSMENT PATTERN :

1. Two tests will be conducted (25 * 2) - 50 marks
2. Five assignments will be conducted (5*10) - 50 Marks.

1704GE353

TECHNICAL SEMINAR I

L	T	P	C
0	0	2	1

PREREQUISITE :

COURSE OBJECTIVES:

1. To develop self-learning skills of utilizing various technical resources to make a technical presentation.
2. To promote the technical presentation and communication skills.
3. To impart the knowledge on intonation, word and sentence stress for improving communicative competence, identifying and overcoming problem sounds.
4. To promote the ability for Interacting and sharing attitude.
5. To encourage the commitment-attitude to complete tasks.

The students are expected to make two presentations on advanced topics (recent trends) related to II year subjects. A faculty guide is to be allotted and he / she will guide and monitor the progress of the student and maintain attendance also. Students are encouraged to use various teaching aids such as powerpoint presentation and demonstrative models.

TOTAL: 30 HOURS

EVALUATION SCHEME:

Continuous Assessment (100 Marks)

Distribution of Marks for Continuous Assessment	Marks
Presentation I	40
Report	10
Presentation II	40
Report	10
Total	100