PAAVAI ENGINEERING COLLEGE, NAMAKKAL – 637 018

(AUTONOMOUS)

B.TECH INFORMATION TECHNOLOGY

CURRICULUM

REGULATION 2015

SEMESTER III

Course Code	Course Title	L	T	P	C
MA15301	Transforms and Boundary Value Problems	3	2	0	4
IT15301	Object Oriented Programming with C++	3	0	0	3
IT15302	Database Management Systems	3	0	0	3
IT15303	Design and Analysis of Algorithms	3	0	0	3
EC15308	Principles of Communication	3	0	0	3
EC15307	Digital Principles and System Design	3	0	0	3
IT15306	Database Management System Laboratory	0	0	4	2
IT15307	Object Oriented Programming with C++ Laboratory	0	0	4	2
EC15309	Digital Laboratory	0	0	4	2
EN15301	Business English Course Laboratory	0	0	2	1

SEMESTER IV

Course Code	Course Title	L	T	P	C
IT15401	Computer Architecture	3	0	0	3
IT15402	Software Engineering	3	0	0	3
IT15403	Operating System	3	0	0	3
EC15408	Microprocessor and Microcontroller	3	0	0	3
MA15404	Numerical Methods	3	0	0	3
IT15404	Java Programming and Applications	3	0	0	3
IT15405	Operating System Laboratory	0	0	4	2
EC15409	Microprocessor and Microcontroller Laboratory	0	0	4	2
IT15406	Java Programming Laboratory	0	0	4	2

MA15301 TRANSFORMS AND BOUNDARY VALUE PROBLEMS 3 2 0 4

COURSE OBJECTIVES

- To introduce Fourier series analysis which are common for many engineering applications apart from solving boundary value problems.
- To acquaint the student with Fourier transform techniques used in wide variety of situations.
- To introduce the effective mathematical tools for the solutions of partial differential equations that model several physical processes .
- To develop Z transform techniques for discrete time systems.

UNIT I FOURIER SERIES 15

Dirichlet's conditions – General Fourier series – Odd and even functions – Half range sine series – Half range cosine series – Complex form of Fourier Series – Parseval's identify – Harmonic Analysis.

UNIT II FOURIER TRANSFORMS 15

Fourier integral theorem (without proof) – Fourier transform pair – Sine and Cosine transforms – Properties – Transforms of simple functions – Convolution theorem – Parseval's identity.

UNIT III PARTIAL DIFFERENTIAL EQUATIONS 15

Formation of partial differential equations – Lagrange's linear equation – Solutions of standard types of first order partial differential equations - Linear partial differential equations of second and higher order with constant coefficients.

UNIT IV APPLICATIONS OF PARTIAL DIFFERENTIAL EQUATIONS 15

Solutions of one dimensional wave equation – One dimensional equation of heat conduction – Steady state solution of two-dimensional equation of heat conduction (Insulated edges excluded).

UNIT V Z TRANSFORMS AND DIFFERENCE EQUATIONS 15

Z-transforms - Elementary properties – Inverse Z-transform – Convolution theorem –Formation of difference equations – Solution of difference equations using Z-transform.

TOTAL: 75 PERIODS

COURSE OUTCOMES

At the end of the course, the students would

- have gained a well-founded knowledge of fourier series, their different possible forms and the frequently needed practical harmonic analysis that an engineer may have to make from discrete data.
- have grasped the concept of expression of a function, under certain conditions, as a double integral leading to identification of transform pair and specialization on fourier transform pair, their properties.

- have obtained capacity to formulate and identify certain boundary value problems encountered in engineering practices, decide on applicability of the fourier series method of solution, solve them and interpret the results.
- be capable of mathematically formulating certain practical problems in terms of partial differential equations, solve them and physically interpret the results.
- have learnt the basics of z – transform in its applicability to discretely varying functions, gained the skill to formulate certain problems in terms of difference equations and solve them using the z – transform technique bringing out the elegance of the procedure involved.

TEXT BOOKS

1. Veerarajan T., "Transforms and Partial Differential Equations", Tata McGraw Hill Education Pvt. Ltd., New Delhi, Second reprint, 2012.
2. Grewal B.S., "Higher Engineering Mathematics", 42nd Edition, Khanna Publishers, Delhi, 2012.
3. Narayanan S., Manicavachagom Pillay.T.K and Ramanaiah.G "Advanced Mathematics for Engineering Students" Vol. II & III, S.Viswanathan Publishers Pvt Ltd. 1998.

REFERENCES

1. Bali.N.P and Manish Goyal, “A Textbook of Engineering Mathematic”, 7th Edition, Laxmi Publications(P) Ltd. (2009)
2. Ramana.B.V., “Higher Engineering Mathematics”, Tata Mc-GrawHill Publishing Company limited, New Delhi (2010).
3. Glyn James, “Advanced Modern Engineering Mathematics”, 3rd Edition, Pearson Education (2007).
4. Erwin Kreyszig, “Advanced Engineering Mathematics”, 8th edition, Wiley India (2007).
5. Ray Wylie C and Barrett.L.C, "Advanced Engineering Mathematics" Tata McGraw Hill Education Pvt Ltd, Sixth Edition, New Delhi, 2012.
6. Datta K.B., "Mathematical Methods of Science and Engineering", Cengage Learning India Pvt Ltd, Delhi, 2013.

IT15301 OBJECT ORIENTED PROGRAMMING WITH C++ 3 0 0 3

COURSE OBJECTIVES

- To learn the basic concepts of Object Oriented Programming.
- To learn the basics of C++ language.
- To illustrates solution of different problems using C++.
- To apply the Object oriented concepts in generic programming.
- To know about master of OOP using C++.

UNIT I PRINCIPLES OF OOP 9

Programming Paradigms- Basic concepts and benefits of OOP- Structure of C++ program – Applications of C++ - Tokens- Keywords- Identifiers-constants- Data types - Basic, User defined ,Derived - Dynamic initialization -Reference variables- Scope resolution operator-Member dereferencing operators- memory management operators- Type casting- Function Prototyping- call by value, call by reference- Inline function- Default arguments – Function overloading.

UNIT II CLASSES AND OBJECTS 9

Class specification- Access qualifiers - Static data members and member functions - Array of objects- Objects as function arguments-Friend functions- Returning objects- Local classes - Constructors – Parameterized constructors- Overloaded Constructors- Constructors with default arguments-Copy constructors- Dynamic constructors-Dynamic initialization using constructors- Destructors - Operator Overloading: Operator function – Overloading unary and binary operator-Overloading the operator using friend function- Stream operator overloading -Type Conversion.

UNIT III INHERITANCE 9

Basic Principle – Use of Inheritance-Defining Derived classes- Single Inheritance-Protected Data with private inheritance- Multiple Inheritance- Multi level inheritance- Hierarchical Inheritance- Hybrid Inheritance-Multipath inheritance- Need for virtual function- Pointer to derived class objects- Definition of virtual functions- Array of pointer to base class objects- Abstract classes- Virtual destructors – Dynamic Binding -Virtual Base Class- Constructors in derived and base class- Pointers- pointers to objects – this pointer.

UNIT IV STREAMS AND FILE HANDLING 9

Stream classes- - Stream classes- Formatted and unformatted data -Formatted I/O- I/O Manipulators- User defined manipulators- File handling -File pointer and manipulation- File open and close- Sequential and random access.

UNIT V GENERIC PROGRAMMING WITH TEMPLATES 9

Function templates, overloaded function templates, user defined template arguments, class templates - Exception Handling: Exception handling mechanism, multiple catch, nested try, re-throwing the exception – Namespaces – std namespace- Standard Template Library.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- identify and apply object oriented concepts like abstraction, encapsulation, modularity, hierarchy, typing, concurrency and persistence.
- estimate various metrics specific to object oriented development.
- apply arrays, pointers and functions to write a C++ program.
- create and use data type, expression and functions in C++.
- use inheritance and templates in C++ program.
- perform efficient exception handling.

TEXT BOOKS

1. E.Balagurusamy, "Object Oriented Programming with C++", Tata McGraw Hill, Sixth Edition, 2013.
2. B.Trivedi, "Programming with ANSI C++", Oxford University Press, 2007.

REFERENCES

1. K.R.Venugopal, Rajkumar, T.Ravishankar, "Mastering C++ ",Tata McGraw Hill, 2007.
2. Robert Lafore, "Object Oriented Programming in Turbo C++", Galgotia Publications, 2006.
3. Bjarne Stroustrup, "The C++ Programming Language", Pearson Education, Fourth Edition, 2013.
4. K.S. Easwarakumar, " Object Oriented Data Structures Using C++", Vikas Publication House Pvt Ltd, First Edition, 2000.

WEB LINKS

1. http://www.tutorialspoint.com/cplusplus/cpp_object_oriented.htm.
2. https://www3.ntu.edu.sg/home/ehchua/programming/cpp/cp3_OOP.html.

IT15302 DATABASE MANAGEMENT SYSTEMS 3 0 0 3

COURSE OBJECTIVES

- To learn the fundamentals of Database Management Systems.
- To make the students understand the relational model.
- To familiarize the students with ER diagrams.
- To expose the students to SQL.
- To familiarize the students with the different types of databases.

UNIT I INTRODUCTION 9

Purpose of Database System — Views of data – Data Models – Database Languages – Database System Architecture – Database users and Administrator – Entity–Relationship model (E-R model) – E-R Diagrams -- Introduction to relational databases.

UNIT II RELATIONAL MODEL 9

The relational Model – The catalog- Types– Keys - Relational Algebra – Domain Relational Calculus – Tuple Relational Calculus - Fundamental operations – Additiona l/O operations- SQL fundamentals - Integrity – Triggers - Security – Advanced SQL features –Embedded SQL– Dynamic SQL- Missing Information– Views – Introduction to Distributed Databases and Client/Server Databases.

UNIT III DATABASE DESIGN 9

Functional Dependencies – Non-loss Decomposition – Functional Dependencies – First, Second, Third Normal Forms, Dependency Preservation – Boyce/ Code Normal Form-Multi-valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form.

UNIT IV TRANSACTIONS 9

Transaction Concepts - Transaction Recovery – ACID Properties – System Recovery –Media Recovery – Two Phase Commit - Save Points – SQL Facilities for recovery –Concurrency – Need for Concurrency – Locking Protocols – Two Phase Locking –Intent Locking – Deadlock- Serializability – Recovery Isolation Levels – SQL Facilities for Concurrency.

UNIT V IMPLEMENTATION TECHNIQUES 9

Overview of Physical Storage Media – Magnetic Disks – RAID – Tertiary storage – File Organization – Organization of Records in Files – Indexing and Hashing –Ordered Indices – B+ tree Index Files – B tree Index Files – Static Hashing – Dynamic Hashing –Query Processing Overview – Catalog Information for Cost Estimation – Selection Operation – Sorting – Join Operation – Database Tuning.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- describe basic concepts of database system.
- design a data model and schemas in RDBMS.
- analyze functional dependencies for designing a robust database.
- apply SQL for business related problems.
- implement transactions, Concurrency control, and be able to do database recovery.

TEXT BOOKS

1. Abraham Silberschatz, Henry F. Korth and S. Sudharshan, "Database System Concepts", Sixth Edition, Tata Mc Graw Hill, 2011.
2. C.J.Date, A.Kannan, S.Swamynathan, "An Introduction to Database Systems",Eighth Edition, Pearson Education, 2006.

REFERENCES

1. Elmasri R. and Shamakant B. Navathe, "Fundamentals of Database Systems", 6th Edition, Addision Wesley , 2011.
2. Atul Kahate, "Introduction to Database Management Systems", Pearson Education, New Delhi, 2006.
3. Raghu Ramakrishnan, "Database Management Systems", Fourth Edition, Tata Mc Graw Hill, 2010.
4. G.K.Gupta, "Database Management Systems", Tata Mc Graw Hill, 2011.
5. Hector Garcia-Molina, Jeff Ullman, and Jennifer Widom, "Database Systems: The Complete Book", Pearson Education, Second Edition, 2008.

WEB LINKS

1. http://www.cs.helsinki.fi/u/laine/tikape/k03/material03.html
2. http://infolab.stanford.edu/~ullman/dscb.html

IT15303 DESIGN AND ANALYSIS OF ALGORITHMS 3 0 0 3

COURSE OBJECTIVES

- To study the principles of algorithm design.
- To know the importance of computational complexity of the algorithm.
- To become familiar with Dynamic programming, divide and conquer, branch and bound and backtracking techniques.
- To understand the limitations of Algorithm power.
- To study about Notions of P, NP, NPC, and NP-hard.

UNIT I INTRODUCTION 9

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving – Important Problem Types – Fundamentals of the Analysis of Algorithm Efficiency – Analysis Framework – Asymptotic Notations and its properties – Mathematical analysis for Recursive and Non-recursive algorithms.

UNIT II BRUTE FORCE AND DIVIDE-AND-CONQUER 9

Brute Force - Closest-Pair and Convex-Hull Problems-Exhaustive Search - Traveling Salesman Problem - Knapsack Problem - Assignment problem. Divide and conquer methodology – Merge sort – Quick sort – Binary search – Multiplication of Large Integers – Strassen's Matrix Multiplication-Closest-Pair and Convex-Hull Problems.

UNIT III DYNAMIC PROGRAMMING AND GREEDY TECHNIQUE 9

Computing a Binomial Coefficient – Warshall's and Floyd's algorithm – Optimal Binary Search Trees – Knapsack Problem and Memory functions. Greedy Technique– Prim's algorithm- Kruskal's Algorithm-Dijkstra's Algorithm-Huffman Trees.

UNIT IV ITERATIVE IMPROVEMENT 9

The Simplex Method-The Maximum-Flow Problem – Maxim Matching in Bipartite Graphs- the Stable marriage Problem.

UNIT V LIMITATIONS OF ALGORITHM POWER 9

Limitations of Algorithm Power-Lower-Bound Arguments-Decision Trees-P, NP and NP-Complete Problems--Coping with the Limitations – Backtracking - n-Queens problem – Hamiltonian Circuit Problem – Subset Sum Problem-Branch and Bound - Assignment problem – Knapsack Problem – Traveling Salesman Problem.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- discuss the significance of algorithms in problem solving process.
- analyze asymptotic runtime complexity of algorithms.
- describe and apply dynamic programming and divide and conquer algorithms.

- design efficient algorithms for new situations, using as building blocks the techniques learned.
- apply algorithm design techniques to solve certain NP-complete problems.

TEXT BOOK

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education, 2012.

REFERENCES

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Third Edition, PHI Learning Private Limited, 2012.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, Reprint 2006.
3. Donald E. Knuth, "The Art of Computer Programming", Volumes 1& 3 Pearson Education, 2009.
4. Steven S. Skiena, "The Algorithm Design Manual", Second Edition, Springer, 2008.

WEB LINKS

1. http://nptel.ac.in/
2. http://freevideolectures.com/Course/2281/Design-and-Analysis-of-Algorithms

EC15308 PRINCIPLES OF COMMUNICATION 3 0 0 3

COURSE OBJECTIVES

- To have understanding about different types of AM and FM Communication systems.
- To gain knowledge in different digital modulation techniques for digital transmission.
- To have knowledge of base band transmission ISI and distortion free base band transmission.
- To know the spread spectrum modulation techniques and different multiple access methods.
- To gain knowledge about Satellite and Optical Communication.

UNIT I ANALOG COMMUNICATION 9

Principles of amplitude modulation, AM envelope, frequency spectrum and bandwidth, modulation index and percent modulation, AM power distribution, Angle modulation - FM and PM waveforms, phase deviation and modulation index, frequency deviation and percent modulation.

UNIT II DIGITAL COMMUNICATION 9

Introduction, Shannon limit for information capacity, digital amplitude modulation, frequency shift keying, FSK bit rate and baud, FSK transmitter, BW consideration of FSK, FSK receiver, phase shift keying – binary phase shift keying – QPSK, Quadrature Amplitude modulation.

UNIT III DIGITAL TRANSMISSION 9

Introduction, Pulse modulation, PCM – PCM sampling, sampling rate, signal to quantization noise rate, delta modulation, adaptive delta modulation, differential pulse code modulation, pulse transmission – Inter symbol interference, eye patterns.

UNIT IV SPREAD SPECTRUM AND MULTIPLE ACCESS TECHNIQUES 9

Introduction, Pseudo-noise sequence, DS spread spectrum with coherent binary PSK, processing gain, FH spread spectrum, multiple access techniques – wireless communication, TDMA and CDMA in wireless communication systems, source coding of speech for wireless communications.

UNIT V SATELLITE AND OPTICAL COMMUNICATION 9

Satellite Communication Systems-Keplers Law, LEO and GEO Orbits, Link model-Optical Communication Systems-Elements of Optical Fiber Transmission link, Types, Losses, Sources and Detectors.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student will be able to

- understand about different types of am and fm communication systems .
- know about different digital modulation techniques for digital transmission.
- gain knowledge in base band transmission is distortion free base band transmission.
- know about spread spectrum modulation techniques and different multiple access methods.
- gain knowledge in satellite and optical communication.

TEXT BOOKS

1. Wayne Tomasi, “Advanced Electronic Communication Systems”, Pearson Education, 2007.
2. Simon Haykin, “Communication Systems”, 4th Edition, John Wiley &Sons., 2001.

REFERENCES

1. H.Taub,D L Schilling ,G Saha ,”Principles of Communication”3/e,2007.
2. B.P.Lathi,”Modern Analog And Digital Communication systems”, 3/e, Oxford University Press, 2007
3. Blake, “Electronic Communication Systems”, Thomson Delmar Publications, 2002.
4. Martin S.Roden, “Analog and Digital Communication System”, 3rd Edition, PHI, 2002.
5. B.Sklar,”Digital Communication Fundamentals and Applications”2/e Pearson Education 2007.

WEB LINKS

1. https://www.youtube.com/watch?v=TPm0XSPxld8
2. www.nptel.ac.in/courses/106105080/pdf/M2L5.pdf

EC15307 DIGITAL PRINCIPLES AND SYSTEM DESIGN 3 0 0 3

COURSE OBJECTIVES

- To learn the basic concepts of Boolean algebra and logic gates.
- To know about the analysis and design procedure for combinational circuits.
- To familiarize the students with memory devices.
- To know about the analysis and design procedure for Asynchronous and synchronous sequential circuits.

UNIT I BOOLEAN ALGEBRA AND LOGIC GATES 9

Review of binary number systems - Binary arithmetic – Binary codes – Boolean algebra and theorems - Boolean functions – Simplifications of Boolean functions using Karnaugh map and tabulation methods – Implementation of Boolean functions using logic gates.

UNIT II COMBINATIONAL LOGIC 9

Combinational circuits - Analysis and design procedures - Circuits for arithmetic operations and Code conversion - Introduction to Hardware Description Language (HDL).

UNIT III DESIGN WITH MSI DEVICES 9

Encoder - decoder - Multiplexer - demultiplexer - HDL for combinational circuits - Memory and programmable logic.

UNIT IV SYNCHRONOUS SEQUENTIAL LOGIC 9

Sequential circuits – Flip flops – Analysis and design procedures - State reduction and state assignment - Shift registers – Counters – HDL for Sequential Circuits.

UNIT V ASYNCHRONOUS SEQUENTIAL LOGIC 9

Analysis and design of asynchronous sequential circuits - Reduction of state and flow tables - Race-free state assignment – Hazards - ASM Chart.

TOTAL: 45 PERIODS

COURSE OUTCOMES

After completion of the course, the students will be able to

- understand the basic concept of Boolean algebra and logic gates.
- design the combinational logic circuits.
- design MSI devices.
- know about memory and programmable logic.
- design the synchronous and asynchronous sequential logic circuits.

TEXT BOOK

1. M.Morris Mano, “Digital Design”, 3rd edition, Pearson Education, 2007.

REFERENCES

1. Charles H.Roth, Jr. “Fundamentals of Logic Design”, 4th Edition, Jaico Publishing House, Cengage Earning, 5th ed, 2005.
2. Donald D.Givone, “Digital Principles and Design”, Tata McGraw-Hill, 2007.

WEB LINKS

1. http://nptel.ac.in/video.php?subjectid=117106086
2. http://www.electronics-tutorials.ws/combination/comb_1.html

IT15306 DATA BASE MANAGEMENT SYSTEM LAB 0 0 4 2

COURSE OBJECTIVES

- To learn to create and use a database.
- To be exposed to different types of database applications.
- To develop conceptual understanding of database management system.
- To understand how a real world problem can be mapped to schemas.
- To develop understanding of different applications and constructs of SQL PL/SQL.

LIST OF EXPERIMENTS

1. Data Definition, Table Creation, Constraints,
2. Insert, Select Commands, Update & Delete Commands.
3. Nested Queries & Join Queries
4. Views
5. High level programming language extensions (Control structures, Procedures and Functions).
6. Front end tools
7. Forms
8. Triggers
9. Menu Design
10. Reports.
11. Database Design and implementation (Mini Project).
    a) Personal Information System.
    b) Web Based User Identification System.
    c) Timetable Management System.
    d) Hotel Management System

TOTAL: 60 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- design and implement a database schema for a given problem-domain.
- populate and query a database.
- create and maintain tables using PL /SQL.
- prepare forms and reports.

IT15307 OBJECT ORIENTED PROGRAMMING WITH C++ LABORATORY 0 0 4 2

COURSE OBJECTIVES

- To know the fundamental knowledge of object oriented programming.
- To develop skills required to become a proficient C++ programmer.
- To transforming the physical problem domain into a hierarchy of objects.
- To apply OOP to solve simple engineering problems.
- To development of solution for complex problems in the real world.

LIST OF EXPERIMENTS

1. Write C++ Programs using Classes and Objects.
2. Design C++ classes with static members, methods with default arguments, friend functions.
3. Develop C++ Programs using Operator Overloading.
4. Develop C++ Programs using constructor, destructor, and copy constructor.
5. Develop C++ Programs Overload the new and delete operators.
6. Develop C++ Programs using Inheritance, Polymorphism and its types.
7. Develop C++ Programs using Arrays and Pointers.
8. Develop C++ Programs using Dynamic memory allocation.
9. Develop C++ Programs using Templates and Exceptions.
10. Develop C++ Programs using Sequential and Random access files.

TOTAL: 60 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- design an object oriented program using classes and objects.
- apply inheritance to reuse the C++ code.
- apply polymorphism to extend the code and reduce the complexity of the program.
- implement files and streams in C++ programs.
- implement exception handling in projects using generic types.

EC15309 **DIGITAL LABORATORY** **0 0 4 2**

COURSE OBJECTIVES

- To understand the concept of Boolean theorems.
- To understand the concept combinational circuits using digital logic gates.
- To design & implement the concept of combinational circuits using MSI devices.
- To design & implement the sequential logic circuits.
- To simulate combinational & sequential logic circuits using VHDL/Verilog.

LIST OF EXPERIMENTS

1. Verification of Boolean theorems using digital logic gates
2. Design and implementation of code converters (i) BCD to Excess-3 code and Excess-3 code to BCD, (ii) Binary to Gray code and Gray code to Binary.
3. Design and implementation of 4-bit binary adder / subtractor using basic gates and MSI devices
4. Design and implementation of parity generator / checker using basic gates and MSI devices
5. Design and implementation of magnitude comparator
6. Design and implementation of application using multiplexers/ Demultiplexers
7. Design and implementation of Shift registers
8. Design and implementation of Synchronous and Asynchronous counters
9. Simulation of combinational circuits using Hardware Description Language

TOTAL: 60 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- design and implement the combinational and sequential circuits.
- simulate combinational and sequential circuits using VHDL/Verilog HDL.
- know about the Boolean theorems using logic gates.
- design various synchronous and asynchronous sequential circuits.
- Acquired knowledge about internal circuitry and logic behind any digital system.

EN15301 BUSINESS ENGLISH COURSE LABORATORY 0 0 2 1

COURSE OBJECTIVES

- To develop the reading skills and get familiarized in skimming and scanning.
- To install the communication concepts to enhance the students conversational skills through various practice sessions.
- To inculcate the receptive skills i.e. Listening and Reading and get well versed in the Productive skills.
- To assist the students in improving their vocabulary and comprehension of grammar.
- To familiarize the students with a variety of business correspondence.

UNIT I READING & VOCABULARY

Understanding short, real world notices, messages - Detailed comprehension of factual material- Skimming & Scanning skills - Interpreting visual information - Reading for detailed factual information - Reading for gist and specific information - Reading for grammatical accuracy and understanding of text structure - Reading and information transfer.

UNIT II WRITING

Re-arranging appointments - Asking for permission - Giving instructions - Apologizing and offering compensation - Making or altering reservations - Dealing with requests - Giving information about a product.

UNIT III LISTENING

Listening for short telephonic conversation - Listening for short conversation or monologue - Listening for specific information - Listening for conversation- Interview, Discussion.

UNIT IV SPEAKING

Conversation between the interlocutor and each candidate - General interaction and social language - A mini presentation by each candidate on a business theme - Organising a larger unit of discourse - giving information and expressing opinions - two way conversation between candidates followed by further prompting from the interlocutor- Expressing opinions- agreeing and disagreeing

TOTAL: 30 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- enrich the vocabulary reading and develop their pronunciation skills.
- speak effectively in English in all occasions.
- prepare flawless reports and proposals.

TEXT BOOKS

1. Cambridge BEC Preliminary, Self Study Edition, Cambridge University Press, New York, 2012.
2. Whitby, Norman. Business Benchmark, Pre-intermediate to intermediate, Business Preliminary, Shree Maitrey Printech Pvt. Ltd., Noida, 2014.

REFERENCES

1. Raman, Meenakshi & Sangeetha Sharma. Technical Communication: Principles and Practice. Oxford University Press, New Delhi. 2011.
2. Rizvi, Ashraf. M. Effective Technical Communication. Tata McGraw-Hill, New Delhi. 2005.
3. Rutherford, Andrea. J Basic Communication Skills for Technology. Pearson, New Delhi. 2001.

WEB LINKS

1. http://www.cambridge.org/us/cambridgeenglish/catalog/cambridge-english-exams-ielts/business-benchmark

IT15401 COMPUTER ARCHITECTURE 3 0 0 3

COURSE OBJECTIVES

- To make students understand the basic structure and operation of digital computer.
- To familiarize the students with arithmetic and logic unit and implementation of fixed point and floating-point arithmetic operations.
- To expose the students to the concept of pipelining.
- To understand the concept of virtual and catch memory.
- To expose the students with different ways of communicating with I/O devices and standard I/O interfaces.

UNIT I BASIC STRUCTURE OF COMPUTERS 9

Functional units – Basic operational concepts – Bus structures – Performance and metrics – Instructions and instruction sequencing – Hardware – Software Interface –Instruction set architecture – Addressing modes – RISC – CISC.

UNIT II BASIC PROCESSING UNIT & ALU OPERATIONS 9

Fundamental concepts – Execution of a complete instruction – Multiple bus organization– Hardwired control – Micro programmed control – ALU-Addition and subtraction–Multiplication–Division.

UNIT III PIPELINING & PARALLELISM 9

Basic concepts – Data hazards – Instruction hazards – Structural Hazards-Influence on instruction sets – Data path and control considerations – Performance considerations – Exception handling- Instruction-level-parallelism – Parallel processing challenges – Flynn's classification – Hardware multithreading - Hardware support for exposing parallelism

UNIT IV MEMORY SYSTEM 9

Basic concepts – Semiconductor RAM – ROM – Speed – Size and cost – Cache memories – Improving cache performance – Virtual memory – Memory management requirements – Associative memories – Secondary storage devices.

UNIT V I/O ORGANIZATION 9

Accessing I/O devices – Programmed Input/ Output -Interrupts – Direct Memory Access– Buses – Interface circuits – Standard I/O Interfaces (PCI, SCSI, USB), I/O devices and processors.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- understand instruction and addressing modes.
- design arithmetic and logic unit.
- design and analyses pipelined control units.
- evaluate performance of memory systems.
- understand parallel processing architectures.

TEXT BOOKS

1. David A. Patterson and John L. Hennessey, "Computer organization and design', Morgan Kauffman / Elsevier, Fifth edition, 2014.
2. Carl Hamacher, ZvonkoVranesic and SafwatZaky, "Computer Organization", Fifth Edition, Tata McGraw Hill, 2002.

REFERENCES

1. V.Carl Hamacher, Zvonko G. Varanesic and Safat G. Zaky, "Computer Organisation", VITH edition, Mc Graw-Hill Inc, 2012.
2. William Stallings "Computer Organization and Architecture" , Seventh Edition , Pearson Education, 2006.
3. Vincent P. Heuring, Harry F. Jordan, "Computer System Architecture", Second Edition, Pearson Education, 2005.
4. William Stallings, "Computer Organization and Architecture – Designing for Performance", Sixth Edition, Pearson Education, 2003.
5. John P. Hayes, "Computer Architecture and Organization", Third Edition, TataMcGraw Hill, 1998.

WEB LINKS

1. http://nptel.ac.in/
2. http://www.cis.upenn.edu/~milom/cse240-Fall05/

IT15402 SOFTWARE ENGINEERING 3 0 0 3

COURSE OBJECTIVES

- To learn Software life cycle models and system engineering process for developing a system from scratch.
- To study the Software project management concepts.
- To know about Prototyping techniques for requirement engineering process and to analyze data, functional and behavioral model.
- To become familiar with Design levels of software engineering.
- To study Software testing and technical metrics for verifying and validating the software.

UNIT I SOFTWARE PROCESS 9

The Evolving role of Software – Software – The changing Nature of Software – Legacy Software —A generic view of process– A layered Technology – A Process Framework – The Capability Maturity Model Integration (CMMI) – Process Assessment – Personal and Team Process Models. Product and Process. Process Models – The Waterfall Model – Incremental Process Models – Incremental Model – The RAD Model – Evolutionary Process Models – Prototyping – The Spiral Model – The Concurrent Development Model – Specialized Process Models – the Unified Process.

UNIT II SOFTWARE REQUIREMENTS 9

Software Engineering Practice – communication Practice – Planning practice modeling practice– Construction Practice –Deployment. Requirements Engineering - Requirements Engineering tasks – Initiating the requirements Engineering Process-Eliciting Requirements – Developing Use cases – Building the Analysis Models –Elements of the Analysis Model – Analysis pattern – Negotiating Requirements – Validating Requirements.

UNIT III REQUIREMENTS ANALYSIS 9

Requirements Analysis – Analysis Modeling approaches – data modeling concepts – Object oriented Analysis – Scenario based modeling – Flow oriented Modeling – Class based modeling – creating a behavior model.

UNIT IV SOFTWARE DESIGN AND SOFTWARE TESTING 9

Design Engineering – Design process -Design Quality-Design model-Agile Methods – Extreme Programming-Rapid Application development – Software Prototyping- Software Reuse – The Reuse Landscape – Design Patterns – Generator-Based Reuse –Application Frameworks – Application System Reuse - Software Evolution Program Evolution Dynamics – Software Maintenance – Evolution Processes – Legacy system evolution Planning -Verification and Validation – Software Inspections – Automated Static analysis – Verification and Formal methods - Software Testing – System Testing – Component Testing – Test case Design –Test Automation.

UNIT V SOFTWARE PROJECT MANAGEMENT 9

Software Cost Estimation – productivity – Estimation Techniques – Algorithmic Cost Modeling –Project Duration and Staffing - Process and Product Quality – Quality Assurance and Standards –Planning – Control- Software Measurement and Metrics - Process Improvement – Process Classification – Measurement –Analysis and Modeling –Change – The CMMI process improvement Framework - Configuration Management. –Planning Change Management – Version and Release Management – System Building – CASE tools for configuration management.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- explore the strength and weakness of life cycle models such as water fall, incremental and spiral model.
- plan, schedule, identify the risk involved and track the development of project for ensuring the software quality.
- identify the functional and non-functional requirements for the project and use it to develop the project using life cycle model.
- apply design processes and concepts for architectural, data, software, user interface and real time systems design.
- verify, and validate the software applications using different types of testing like black box testing, structural testing, unit testing etc.

TEXT BOOK

1. Roger Pressman.S, ―Software Engineering: A Practitioner's Approach", Seventh Edition, McGraw Hill, 2010.

REFERENCES

1. Ian Sommerville,"Software Engineering ", 9th Edition, Pearson Education Asia, 2011.
2. S.A. Kelkar, "Software Engineering, A Concise Study", Prentice Hall of India, 2007
3. Richard E. Fairley, "Principles of Software Engineering", IEEE computer society press, 2010.
4. Shari Pfleeger, Joanne Atlee, "Software Engineering: Theory and Practice", Fourth Edition, Pearson Education, 2010.
5. Pankaj Jalote, "Software Engineering, A Precise Approach", Wiley India, 2010.

WEB LINKS

1. http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=10793
2. http://www.softwaretestingtimes.com/2010/04/software-testing-tutorials-for.html

IT15403 OPERATING SYSTEMS 3 0 0 3

COURSE OBJECTIVES

- To study the basic concepts and functions of operating systems.
- To understand the structure and functions of OS.
- To learn about Processes, Threads and Scheduling algorithms.
- To understand the principles of concurrency and Deadlocks.
- To learn various memory management schemes.

UNIT I INTRODUCTION 9

Introduction: Computer system organization - Introduction to operating systems – operating system structures – services - system calls – system programs. Processes: Process concept – Process scheduling – Operations on processes –Cooperating processes – Inter process communication – Communication in client-server systems. Threads: Multi-threading models – Threading issues. Case Study: Pthreads library.

UNIT II PROCESS MANAGEMENT AND DEADLOCK 10

CPU Scheduling: Scheduling criteria – Scheduling algorithms – Multiple-processor scheduling – Real time scheduling – Algorithm Evaluation. Process Synchronization: The critical-section problem – Synchronization hardware – Semaphores – Classic problems of synchronization – Monitors. Deadlock: System model – Deadlock characterization –Methods for handling deadlocks – Deadlock prevention – Deadlock avoidance –Deadlock detection – Recovery from deadlock. Case Study: Process scheduling in Linux.

UNIT III MEMORY MANAGEMENT 9

Main Memory: Background – Swapping – Contiguous memory allocation –Paging – Segmentation – Segmentation with paging. Virtual Memory: Background –Demand paging – Page replacement – Allocation of frames –Thrashing. Case Study: Memory management in windows and Solaris.

UNIT IV FILE SYSTEMS 9

File-System Interface: File concept – Access methods – Directory structure – File system mounting – File sharing - Protection. File-System Implementation: Directory implementation –Allocation methods – Free-space management – efficiency and performance – recovery– Network file systems. Case studies: File system in Windows XP.

UNIT V I/O SYSTEMS AND MASS STORAGE MANAGEMENT 8

I/O Systems – I/O Hardware – Application I/O interface – kernel I/O subsystem –streams – performance. Mass-Storage Structure: Disk attachment - Disk scheduling – Disk management –Swap-space management – RAID — stable storage. Case study: I/O in Linux.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- design various scheduling algorithms.
- apply the principles of concurrency.
- design deadlock, prevention and avoidance algorithms.
- compare and contrast various memory management schemes.
- schedule and manage the disk effectively .

TEXT BOOK

1. Silberschatz, Galvin, and Gagne, “Operating System Concepts”, Ninth Edition, Wiley India Pvt Ltd, 2013.

REFERENCES

1. Andrew S. Tanenbaum, “Modern Operating Systems”, Fourth Edition, Pearson Education, 2014.
2. 2William Stallings, “Operating Systems – internals and design principles”, Prentice Hall, 7thEdition, 2011.
3. Harvey M. Deital, “Operating Systems”, Third Edition, Pearson Education, 2007.
4. Andrew S. Tannenbaum & Albert S. Woodhull, “Operating System Design and Implementation”, Prentice Hall, 3rd Edition, 2006.
5. Gary J.Nutt, “Operating Systems”, Pearson/Addison Wesley, 3rd Edition, 2004.

WEB LINKS

1. http://courses.cs.vt.edu/csonline/OS/Lessons/
2. http://www.linux-tutorial.info/modules.php?name=MContent&pageid=4

EC15408 MICROPROCESSOR AND MICROCONTROLLER 3 0 0 3

COURSE OBJECTIVES

- To study the Architecture of 8086 microprocessor.
- To study about programming of 8086 microprocessor.
- To learn the design aspects of I/O and Memory Interfacing circuits.
- To study the Architecture of 8051 microcontroller.
- To learn the concepts of system design using microcontroller.

UNIT I THE 8086 MICROPROCESSOR 9

Introduction to microprocessor, Bus-Address bus, Data bus and Control bus, Connecting Microprocessor to I/O devices, Introduction to 8086 – Microprocessor architecture, 8086 signals – Basic configurations and Interrupts.

UNIT II 16 BIT MICROPROCESSOR INSTRUCTION SET AND ASSEMBLY LANGUAGE PROGRAMMING 9

Addressing modes - Operand types- Instruction set and assembler directives – Assembly language programming.

UNIT III I/O INTERFACING 9

Memory Interfacing and I/O interfacing - Parallel communication interface – Serial communication interface – D/A and A/D Interface - Timer – Keyboard /display controller – Interrupt controller – DMA controller.

UNIT IV MICROCONTROLLER 9

Architecture of 8051 – Signals - Special Function Registers (SFRs) - I/O Ports – Memory-Interrupts - Instruction set - Addressing modes - Assembly language programming.

UNIT V SYSTEM DESIGN USING MICROCONTROLLER 9

Case studies – Traffic light control, washing machine control, DC & Stepper Motor & Keyboard Interfacing - ADC, DAC - External Memory Interface.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- design and implement programs on 8086 microprocessor.
- analyze and design multiprocessor system.
- design I/O circuits.
- design memory interfacing circuits.
- design and implement 8051 microcontroller based systems.

TEXT BOOKS

1. Krishna Kant, “Microprocessors and Microcontrollers Architecture, programming and system design using 8085, 8086, 8051 and 8096”. PHI 2007.
2. Kenneth J.Ayala, “The 8051 Microcontroller Architecture, Programming and applications”, Second edition, Penram International.
3. Mohamed Ali Mazidi, Janice Gillispie Mazidi, Rolin McKinlay, “The 8051 Microcontroller and Embedded Systems: Using Assembly and C”, Second Edition, Pearson Education, 2011.

REFERENCES

1. Doughlas V.Hall, “Microprocessors and Interfacing, Programming and Hardware:, TMH.
2. A.K.Ray & K.M Bhurchandi, “Advanced Microprocessor and Peripherals – Architecture, Programming and Interfacing”, Tata Mc Graw Hill , 2006.

WEBLINKS

1. http://nptel.ac.in/courses/106103068/47
2. http://nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/microcontrollers/chap2.pdf

MA15404 NUMERICAL METHODS 3 0 0 3

COURSE OBJECTIVES

- To solve any type of mathematical equations, integrations and differentiations of any functions using Numerical methods.
- To develop the skills of engineering students in the basis of complete procedure for solving different kinds of problems occur in engineering numerically.
- To develop efficient algorithms for solving problems in science, engineering and technology.
- The methods introduced in the solution of ordinary differential equations and partial differential equations will be useful in attempting any engineering problem.

UNIT I SOLUTION OF EQUATIONS AND EIGEN VALUE PROBLEMS 9

Solution of equation –Iteration method : Newton Raphson method – Solution of linear system by Gaussian elimination and Gauss - Jordon method – Iterative method – Gauss-Seidel method – Inverse of a matrix by Gauss Jordon method – Eigenvalue of a matrix by power method.

UNIT II INTERPOLATION AND APPROXIMATION 9

Lagrangian Polynomials – Divided differences – Newton's Divided Difference, Hermite Interpolation Polynomial and Interpolating with a cubic spline – Newton‟s forward and backward difference formulas.

UNIT III NUMERICAL DIFFERENTIATION AND INTEGRATION 9

Differentiation using interpolation formulae –Numerical integration by trapezoidal and Simpson's 1/3– Romberg's method – Two and Three point Gaussian quadrature formulas – Double integrals using trapezoidal and Simpsons' rule.

UNIT IV INITIAL VALUE PROBLEMS FOR ORDINARY DIFFERENTIAL EQUATIONS 9

Single step methods: Taylor series method – Modified Euler method for first order equation – Fourth order Runge – Kutta method for solving first and second order equations – Multistep methods: Milne's and Adam's predictor and corrector methods.

UNIT V BOUNDARY VALUE PROBLEMS IN ORDINARY AND PARTIAL DIFFERENTIAL EQUATIONS 9

Finite difference solution of second order ordinary differential equation – Finite difference solution of one dimensional heat equation by explicit and implicit methods – One dimensional wave equation and two dimensional Laplace and Poisson equations.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- an ability to develop confidence for self-learning and life- long learning.
- manage resources like men, money, machinery and material with modern management tools.
- applications of numerical methods in computer field.
- the methods discussed on interpolation will be useful in constructing approximate polynomial to represent the data and to find the intermediate values.

TEXT BOOKS

1. Erwin Kreyszig., " Advanced Engineering Mathematics " 10th edition, Wiley Publications, 2010.
2. T. Veerarajan. and T .Ramachandran, "Numerical Methods with programming in C", 2nd ed., Tata McGraw-Hill, 2006.
3. Sankar Rao K " Numerical Methods For Scientists And Engineers –3rd Edition Prentice Hall of India Private, New Delhi, 2007.

REFERENCES

1. P. Kandasamy, K. Thilagavathy and K. Gunavathy, "Numerical Methods", S.Chand Co. Ltd., New Delhi, 2003.
2. Gerald C.F. and Wheatley, P.O., "Applied Numerical Analysis" 6th Edition, Pearson Education Asia, New Delhi, 2002.
3. M.K.Jain , S.R.K. Iyangar , R.K.Jain , "Numerical Methods For Scientific & Engineering Computation" , New Age International (P) Ltd , New Delhi , 2005.

WEB LINKS

1. http://nptel.ac.in/courses/122102009/
2. https://ece.uwaterloo.ca/~ece204/tutorials/t1.html

IT15404 JAVA PROGRAMMING AND APPLICATIONS 3 0 0 3

COURSE OBJECTIVES

- To understand the concepts of Object Oriented Programming.
- To understand the concepts of inheritance.
- To develop an application in event driven programming.
- To develop an application in generic programming.
- To develop an application in concurrent programming.

UNIT I OBJECT-ORIENTED PROGRAMMING – FUNDAMENTALS 9

Review of OOP - Objects and classes in Java – defining classes – methods –access specifies – static members – constructors – finalize method – Arrays – Strings -Packages – Java Doc comments.

UNIT II OBJECT-ORIENTED PROGRAMMING – INHERITANCE 10

Inheritance – class hierarchy – polymorphism – dynamic binding – final keyword –abstract classes – the Object class – Reflection – interfaces – object cloning – inner classes – proxies.

UNIT III EVENT-DRIVEN PROGRAMMING 10

Graphics programming – Frame – Components – working with 2D shapes – Using color, fonts, and images - Basics of event handling – event handlers – adapter classes –actions – mouse events – AWT event hierarchy – introduction to Swing – Model-View-Controller design pattern – buttons – layout management – Swing Components.

UNIT IV GENERIC PROGRAMMING 8

Motivation for generic programming – generic classes – generic methods – generic code and virtual machine – inheritance and generics – reflection and generics – exceptions –exception hierarchy – throwing and catching exceptions – Stack Trace Elements -assertions – logging.

UNIT V CONCURRENT PROGRAMMING 8

Multi-threaded programming – interrupting threads – thread states – thread properties –thread synchronization – thread-safe Collections – Executors – synchronizers – threads and event-driven programming.

TOTAL: 45 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- understand the needs of object oriented programming.
- differentiate the functionalities of object oriented approach and procedural languages.
- demonstrate the concepts of event-driven programming.
- exhibit the concepts of generic programming using Java .
- perform the concepts of concurrent programming.

TEXT BOOKS

1. Cay S. Horstmann and Gary Cornell, “Core Java: Volume I – Fundamentals”, Eighth Edition, Sun Microsystems Press, 2008.
2. Herbert Schildt, Java2-CompleteReference, Tata McGraw Hill, 2011.
3. Deitel & Deitel, Java How to Program, Prentice Hall of India, 2010.

REFERENCES

1. K. Arnold and J. Gosling, “The JAVA programming language”, Third edition, Pearson Education, 2000.
2. Timothy Budd, “Understanding Object-oriented programming with Java”, Updated Edition, Pearson Education, 2000.
3. C. Thomas Wu, “An introduction to Object-oriented programming with Java”, Fourth Edition, Tata McGraw-Hill Publishing Company Ltd., 2006.
4. Gary Cornell and Cay S. Horstmann, Core Java Vol.1andVol.2,Sun Microsystems Press,2008
5. Herbert Schildt, Java, A Beginner's Guide, Tata McGraw Hill,2007

WEB LINKS

1. www.javatpoint.com/java-oops-concepts
2. www.w3resource.com/java.../java-object-oriented-programming.php

IT15405 OPERATING SYSTEMS LABORATORY 0 0 4 2

COURSE OBJECTIVES

- To implement scheduling algorithms.
- To learn to use the file allocation and organization strategies.
- To be familiar with implementation of deadlock avoidance & detection algorithms.
- To implement page replacement algorithms.
- To be exposed to process creation and inter process communication.

LIST OF EXPERIMENTS

1. Simulate the following CPU scheduling algorithms: a) Round Robin b) SJF c) FCFS d) Priority.
2. Simulate all file allocation strategies: a) Sequential b) Indexed c) Linked.
3. Implement the producer – consumer problem using semaphores.
4. Simulate all File Organization Techniques:

   a) Single level directory b) Two level c) Hierarchical d) DAG.
5. Simulate Bankers Algorithm for Dead Lock Avoidance.
6. Simulate an Algorithm for Dead Lock Detection.
7. Simulate all page replacement algorithms a) FIFO b) LRU c) Optimal.
8. Simulate Shared memory and IPC.
9. Simulate Paging Technique of memory management.
10. Implement Threading & Synchronization Applications.

TOTAL: 60 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- compare the performance of various CPU scheduling algorithm.
- implement file allocation & organization strategies.
- implement deadlock avoidance, and detection algorithms.
- critically analyze the performance of the various page replacement algorithms.
- create processes and implement inter process communication.

EC15409 MICROPROCESSOR AND MICROCONTROLLER LABORATORY 0 0 4 2

COURSE OBJECTIVES

- To implement the assembly language programming of 8086 and 8051.
- To experiment the interface concepts of various peripheral device with the processor.
- To impart the knowledge about the instruction set.
- To understand the basic idea about the data transfer schemes and its applications.
- To develop skill in simple program writing for 8051 & 8086 and applications.

LIST OF EXPERIMENTS

Assembly Language programming using 8086 and MASM

1. Basic arithmetic and Logical operations.
2. Move a data block without overlap.
3. Floating point operations, string manipulations, sorting and searching.
4. Counters and Time Delay.

Interfacing with 8086 microprocessor

5. Traffic light control.
6. Stepper motor control.
7. Digital clock.
8. Key board and Display.
9. Serial interface and Parallel interface.

Programming using 8051 microcontroller

10. Basic arithmetic and Logical operations.
11. Unpacked BCD to ASCII.

TOTAL: 60 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to

- write assembly language programmes for various applications.
- interface different peripherals with microprocessor.
- execute programs in 8051.
- explain the difference between simulator and emulator.
- develop strong competencies in Physics and its applications in a technology-rich, interactive environment.

IT15406 JAVA PROGRAMMING LABORATORY 0 0 4 2

COURSE OBJECTIVES

- To be familiar with core programming in java.
- To be familiar with Interface and Thread concepts.
- To design front end and back end connectivity.
- To know about GUI & Event driven programming.
- To understand the concepts of synchronization, multi-threading.

LIST OF EXPERIMENTS

1. Develop a Java package with simple Stack and Queue classes. Use Java Documents for documentation.
2. Design a class for Complex numbers in Java. In addition to methods for basic operations on complex numbers, provide a method to return the number of active objects created.
3. Design a Date class similar to the one provided in the java.util package.
4. Develop with suitable hierarchy, classes for Point, Shape, Rectangle, Square, Circle, Ellipse, Triangle, Polygon, etc. Design a simple test application to demonstrate dynamic polymorphism.
5. Design a Java interface for ADT Stack. Develop two different classes that implement this interface, one using array and the other using linked-list. Provide necessary exception handling in both the implementations.
6. Write a Java program to read a file that contains DNA sequences of arbitrary length one per line (note that each DNA sequence is just a String). Your program should sort the sequences in descending order with respect to the number of 'TATA' sub sequences present. Finally write the sequences in sorted order into another file.
7. Develop a simple paint-like program that can draw basic graphical primitives indifferent dimensions and colors. Use appropriate menu and buttons.
8. Develop a scientific calculator using even-driven programming paradigm of Java.
9. Develop a template for linked-list class along with its methods in Java.
10. Design a thread-safe implementation of Queue class. Write a multi-threaded producer-consumer application that uses this Queue class.
11. Develop a multi-threaded GUI application of your choice.

TOTAL: 60 PERIODS

COURSE OUTCOMES

At the end of the course, the student should be able to:

- develop core java programs and solving problems.
- implement multiple inheritance and multi thread programs.
- develop the data structures such as list, linked list.
- design AWT based applications using exceptional handling.
- design frontend using AWT and backend with a database and mini projects by using core JAVA.