

BE(CSE) SEMESTER-III, IV with Effect from AY(21-22)

CSE: SEMESTER – III

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/D	Contact Hrs/Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	HS 103 ME	Operations Research	3	-	-	3	30	70	3	3
2	ES 306 EC	Basic Electronics	3	-	-	4	30	70	3	3
3	ES 303 EC	Digital Electronics	3	1	-	4	30	70	3	3
4	PC 301 CS	Data Structures and Algorithms	3	1	-	4	30	70	3	3
5	PC 302 CS	Discrete Mathematics	3	1	-	4	30	70	3	3
6	PC 303 CS	OOP using JAVA	3	1	-	4	30	70	3	3
Practical/ Laboratory Courses										
7	PC 351 CS	Data Structures and Algorithms Lab	-	-	2	2	25	50	3	1
8	PC 352 CS	Advanced Computer Skills Lab	-	-	2	2	25	50	3	1
9	ES 351 EC	Basic Electronics Lab	-	-	2	2	25	50	3	1
10	PC 353 CS	OOP using JAVA Lab	-	-	2	2	25	50	3	1
Total			18	4	08	30	280	620		22

CSE: SEMESTER – IV

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/D	Contact Hrs/Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	HS 104 EG	Effective Technical Communication in English	3	-	-	3	30	70	3	3
2	HS 105 CM	Finance and Accounting	3	1	-	4	30	70	3	3
3	BS 207 MT	Mathematics – III (Probability & Statistics)	3	-	-	3	30	70	3	3
4	ES 305 EC	Signals and Systems	3	-	-	3	30	70	3	3
5	PC 401 CS	Operating Systems	3	-	-	3	30	70	3	3
6	PC 402 CS	Computer Organization	3	1	-	4	30	70	3	3
7	PC 403 CS	Database Management Systems	3	1	-	4	30	70	3	3
Practical/ Laboratory Courses										
8	PC 451 CS	Computer Organization Lab	-	-	2	2	25	50	3	1
9	PC 452 CS	Operating Systems Lab	-	-	2	2	25	50	3	1
10	PC 453 CS	Database Management Systems Lab	-	-	2	2	25	50	3	1
Total			21	3	06	30	285	640		24

HS 103 ME

Instruction: 3 periods per week

CIE: 30 marks

Credits: 3

OPERATIONS RESEARCH

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

1. Use variables for formulating complex mathematical models in management science, industrial engineering and transportation models.
2. Use the basic methodology for the solution of linear programming problems
3. Understand the mathematical tools that are needed to solve optimization problems like Transportation models and Assignment models
4. Understand the replacement models with change in money value considering with time and without time.
5. Model a system as a queuing model and compute important performance measures

Outcomes: Student will be able to:

1. Prepare the students to have the knowledge of Linear Programming Problem in Operations
2. Research at the end students would be able to understand the concept and develop the models for different applications.
3. Make students understand the concept Replacement models at the end students would able to explain various features and applications of replacement models in real time scenario.
4. Prepare the students to understand theory of Game in operations research at the end students would able to explain application of Game theory in decision making for a conflict

UNIT – I

Introduction: Definition and Scope of Operations Research.

Linear Programming: Introduction, Formulation of linear programming problems, graphical method of solving LP problem, simplex method, maximization and minimization, Degeneracy in LPP, Unbounded and, Infeasible solutions.

UNIT – II

Duality: Definition, Relationship between primal and dual solutions, Economic Interpretation, Post optimal of sensitivity analysis, Dual Simplex Method..

UNIT – III

Transportation Models: Finding an initial feasible solution - North West corner method, least cost method, Vogel's Approximation method, Finding the optimal solution, optimal solution by stepping stone and MODI methods, Special cases in Transportation problems - Unbalanced Transportation problem.

Assignment Problems: Hungarian method of Assignment problem, Maximization in Assignment problem, unbalanced problem, problems with restrictions, travelling salesman problems.

UNIT – IV

Replacement Models: Introduction, replacement of items that deteriorate ignoring change in money value, replacement of items that deteriorate considering change in money value with time, replacement of items that fail suddenly - Individual replacement policy, Group replacement policy.

Game Theory: Introduction, 2 person zero sum games, Maximin - Minimax principle, Principle of Dominance, Solution for mixed strategy problems, Graphical method for $2 \times n$ and $m \times 2$ games.

UNIT – V

Sequencing Models: Introduction, General assumptions, processing n jobs through 2 machines, processing ' n ' jobs through m machines, Processing 2 jobs through m machines

Queuing Theory: Introduction, single channel - Poisson arrivals - exponential service times with infinite population & finite population, Multi channel - poisson arrivals - Exponential service times with infinite population.

Introduction to Optimization Techniques: Single objective & Multi objective optimization Techniques like G.A, NSGA, P.Q.O & MPSO Techniques.

Suggested Readings:

1	Hamdy, A. Taha, "Operations Research-An Introduction", Sixth Edition, Prentice Hall of India Pvt. Ltd.,1997
2	S.D. Sharma, Operations Research, Kedarnath, Ramnath & Co., Meerut,2009
3	J.B. Gupta, "Utilization of Electric Power and Electric Traction" S.K. Kataria & Sons Publications, 2010 Hrvey M. Wagner, Principles of Operations Research, Second Edition, Prentice Hall of India Ltd., 1980.
4	V.K. Kapoor, Operations Research, S. Chand Publishers, New Delhi,2004
5	R. Paneer Selvam, Operations Research, Second Edition, PHI Learning Pvt. Ltd., New Delhi,2008.

ES 306 EC

BASIC ELECTRONICS

Instruction: 3+1 periods per week

Duration of SEE: 3 hours

CIE: 30 marks

SEE: 70 marks

Credits: 3

Course Objectives

1. To understand the characteristics of diodes and transistor configurations
2. To understand the design concepts of biasing of BJT and FET
3. To understand the design concepts of feedback amplifiers and oscillators
4. To study the design concepts of OP Amp and data converters

Course Outcomes

The student will be able to:

1. Study and analyse the rectifiers and regulator circuits.
2. Study and analyse the performance of BJTs, FETs on the basis of their operation and working.
3. Ability to analyse & design oscillator circuits.
4. Ability to analyse different logic gates & multi-vibrator circuits.
5. Ability to analyse different data acquisition systems

UNIT – I
PN Junction Diode: Characteristics, Half wave rectifier, Full wave rectifier, filters, ripple, regulation, TIF and efficiency, Zener diode and Zener diode regulators. CRT construction and CRO applications.
UNIT – II
Transistors: BJT construction and working, modes of operation, configurations of BJT (CB, CE, CC), small signal h-parameter model of CE, CE amplifier analysis. Construction and working of JFET, V-I characteristics of JFET.
UNIT – III
Feedback concepts: Types of negative feedback – modification of gain, bandwidth, input and output impedances, applications. Oscillators: RC Phase shift, Wein bridge, LC and crystal Oscillators (Qualitative treatment only).
UNIT – IV
Operational Amplifier: OP-AMP Block diagram, Ideal OP-AMP, DC and AC Characteristics, Inverting and Non-Inverting Amplifiers, Adder/Subtractor, Integrator, Differentiator. Logic gate circuits - Introduction to Digital systems- AND, NAND, NOR, XOR gates, Binary half adder, full adder.
UNIT – V
Data Acquisition Systems: Construction and Operation of transducers- Strain guage LVDT, Thermo couple, Instrumentation systems. Data Converters: R-2R Ladder DAC, Successive approximation and Flash ADC.

Suggested Readings:

1	Robert Boylestad L. and Louis Nashelsky, Electronic Devices and Circuit Theory, PHI,2007.
2	Helfrick Dand DavidCooper,ModernElectronicInstrumentationandMeasurementsTechniques, 1 st Edition, Prentice Hall of India, 2006.
3	Salivahanan, Suresh Kumar and Vallavaraj, Electronic Devices and Circuits, 2 nd Edition, Tata McGraw-Hill,2010.

ES 303 EC

DIGITAL ELECTRONICS

Instruction: 3+ 1 periods per week

Duration of SEE: 3 hours

CIE: 30 marks

SEE: 70 marks

Credits: 3

Course Objectives

1. To learn the principles of digital hardware and support given by it to the software.
2. To explain the operation and design of combinational and arithmetic logic circuits.
3. To design hardware for real world problems.

Course Outcomes

At the end of this course the students will be able to

1. Understand the design process of digital hardware, use Boolean algebra to minimize the logical expressions and optimize the implementation of logical functions.
2. Understand the number representation and design combinational circuits like adders, MUX etc.
3. Design Combinational circuits using PLDs and write VHDL code for basic gates and combinational circuits.
4. Analyse sequential circuits using flip-flops and design registers, counters.
6. Represent a sequential circuit using Finite State machine and apply state minimization techniques to design a FSM

UNIT – I
Design Concepts: Digital Hardware, Design process, Design of digital hardware. Introduction to logic circuits – Variables and functions, Logic gates and networks. Boolean algebra, Synthesis using gates, Design examples. Optimized implementation of logic functions using K-Map and Quine-McCluskey Tabular method
UNIT – II
Number Representation: Addition and Subtraction of signed and unsigned numbers. Combinational circuit building blocks: Half adder, Full adder, Multiplexers. Decoders. Encoders. Code converters, BCD to 7-segment converter, Arithmetic comparator circuits.
UNIT – III
Design of combinational circuits using Programmable Logic Devices (PLDs): General structure of a Programmable Array Logic (PAL), Programmable Logic Arrays (PLAs), Structure of CPLDs and FPGAs, 2- input and 3-input lookup tables (LUTs). Introduction to Verilog HDL: Verilog code for basic logic gates, adders, decoders.
UNIT – IV
Sequential Circuits: Basic Latch, Gated SR Latch, gated D Latch, Master-Slave edge triggered flip-flops, T Flip-flop, JK Flip-flop, Excitation tables. Registers, Counters, Verilog code for flip-flops
UNIT – V
Synchronous Sequential Circuits: Basic Design Steps, Finite State machine (FSM) representation using Moore and Mealy state models, State minimization, Design of FSM for Sequence Generation and Detection, Algorithmic State Machine charts.

Proposed for the academic years 2020-2024

Suggested Readings:

1	Moris Mano and Michael D Ciletti, Digital Design, Pearson, fourth Edition,2008.
2	ZviKohavi, Switching and Finite Automata Theory, 3rd ed., Cambridge University Press-New Delhi, 2011.
3	Samir Palnitkar, “Verilog HDL A Guide to Digital Design and Synthesis,” 2nd Edition, Pearson Education,2006.

Course Code	Course Title				Core/Elective		
PC301CS	DATA STRUCTURES AND ALGORITHMS				Core		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
-	L	T	D	P	30	70	3
	3	1	-	-			

Objectives:

1. To develop proficiency in the specification, representation, and implementation of abstract data types and data structures.
2. To discuss the linear and non-linear data structures and their applications
3. To introduce the creation, insertion and deletion operations on binary search trees and balanced binary search trees.
4. To introduce various internal sorting, searching techniques and their time complexities

Outcomes:

After completing this course, the student will be able to:

1. Understand the importance of abstract data type and implementing the concepts of data structure using abstract data type.
2. Evaluate an algorithm by using algorithmic performance and measures.
3. Distinguish between linear and non-linear data structures and their representations in the memory using array and linked list.
4. Apply the suitable data structure for a real world problem and think critically for improvement in solutions.
5. Determine the suitability of the standard algorithms: Searching, Sorting and Traversals

UNIT – I
Algorithms: Introduction, Algorithm Specifications, Recursive Algorithms, Performance Analysis of an algorithm- Time and Space Complexity, Asymptotic Notations.
Arrays: Arrays - ADT, Polynomials, Sparse matrices, Strings-ADT, PatternMatching.
UNIT – II
Stacks and Queues: Stacks, Stacks using Arrays, Stacks using dynamic arrays, Evaluation of Expressions – Evaluating Postfix Expression, Infix to Postfix.
Queues: Queues ADT, operations, Circular Queues, Applications
UNIT – III
Linked Lists: Singly Linked Lists and Chains, Linked Stacks and Queues, Polynomials, Operations for Circularly linked lists, Equivalence Classes, Sparse matrices, Doubly Linked Lists.
Hashing: Static Hashing, Hash Tables, Hash Functions, Overflow Handling, Theoretical Evaluation of Overflow Techniques
UNIT – IV
Trees: Introduction, Binary Trees, Binary Tree Traversals, Heaps, Binary Search trees (BST) : Definition, Searching an element, Insertion into a BST, Deletion from a BST.
Efficient Binary Search Trees: AVL Trees: Definition, Searching an element, Insertion into a AVL

UNIT – V

Graphs: Graph Abstract Data Type, Elementary Graph operations (DFS and BFS), Minimum Cost Spanning Trees (Prim's and Kruskal's Algorithms).

Sorting and Searching: Insertion sort, Quick sort, Best computing time for Sorting, Merge sort, Heap sort, shell sort, Sorting on Several Keys, List and Table Sorts, Summary of Internal Sorting, Linear and Binary Search algorithms.

Suggested Books:

1. Horowitz E, Sahni S and Susan Anderson-Freed, Fundamentals of Data structures in C, 2nd Edition (2008), Universities Press

Reference Books:

1. Mark A Weiss, Data Structures and Algorithm Analysis In C, Second Edition (2002), Pearson
2. Kushwaha D. S and Misra A.K, Data structures A Programming Approach with C, Second Edition (2014), PHI.
3. Gilberg R. F and Forouzan B. A, Data structures: A Pseudocode Approach with C, Second Edition (2007), Cengage Learning
4. Tanenbaum A. M , Langsam Y. Augenstein M. J, Data Structures using C, Second Edition (2008), Pearson.
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, Introduction to Algorithms, Third Edition (2009), MIT Press
6. Yedidyah Langsam , Moshe J. Augenstein ,Aaron M. Tenenbaum, Data Structures Using C and C++ , Second Edition (2009), PHI

Proposed for the academic years 2020-2024
DISCRETE MATHEMATICS

PC 302 CS

Instruction: 3 +1 periods per week

CIE: 30 marks

Credits: 3

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

3. To explain with examples, the basic terminology of functions, relations, and sets.
4. To perform the operations associated with sets, functions, and relations.
5. To relate practical examples to the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
4. To describe the importance and limitations of predicate logic.
5. To relate the ideas of mathematical induction to recursion and recursively defined structures.
6. To use Graph Theory for solving problems.

Outcomes:

After completing this course, the student will be able to:
5. Illustrate by examples the basic terminology of functions, relations, and sets and demonstrate knowledge of their associated operations.
6. Understand basics of counting, apply permutations and combinations to handle different types of objects.
7. Describe and use recursively-defined relationships to solve problems using generating functions.
8. Analyse semi group, monoid group and abelian group with suitable examples and appreciate group theory applications in computer arithmetic.
9. Demonstrate in practical applications the use of basic counting principles of permutations, combinations, inclusion/exclusion principle and the pigeon hole methodology.

UNIT – I

Mathematical Logic: Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, equivalence implication, Normal forms, Quantifiers, universal quantifiers. Predicates: Predicative logic, Free & Bound variables, Rules of inference, Consistency, proof of contradiction, Automatic Theorem Proving

UNIT – II

Relations: Properties of Binary Relations, equivalence, transitive closure, compatibility and partial ordering relations, Lattices, Hasse diagram. Functions: Inverse Function Composition of functions, recursive Functions, Lattice and its Properties, Algebraic structures: Algebraic systems Examples and general properties, Semi groups and monads, groups sub groups' homomorphism, Isomorphism.

UNIT – III

Elementary Combinatorics: Basis of counting, Combinations & Permutations, with repetitions, Constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, the principles of Inclusion – Exclusion. Pigeon hole principles and its application.

UNIT – IV

Recurrence Relation: Generating Functions, Function of Sequences Calculating Coefficient of generating function, Recurrence relations, Solving recurrence relation by substitution and Generating

Proposed for the academic years 2020-2024

funds. Characteristics solution of in homogeneous Recurrence Relation.

UNIT – V

Graph Theory: Representation of Graph, DFS, BFS, Spanning Trees, planar Graphs. Graph Theory and Applications, Basic Concepts Isomorphism and Sub graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Chromatic Numbers.

Suggested Readings:

- | |
|--|
| 1. Elements of Discrete Mathematics- A Computer Oriented Approach- C L Liu, D P Mohapatra. Third Edition, Tata McGrawHill. |
| 7. Discrete Mathematics for Computer Scientists & Mathematicians, J.L. Mott, A. Kandel, T.P. Baker, PHI. |
| 8. Discrete Mathematics and its Applications, Kenneth H. Rosen, Fifth Edition.TMH. |
| 9. Discrete Mathematical Structures Theory and Application-Malik & Sen,Cengage. |
| 10. Discrete Mathematics with Applications, Thomas Koshy,Elsevier |
| 11. Logic and Discrete Mathematics, Grass Man & Trembley, Pearson Education |

OOP using JAVA PC303CS

Instruction: 3 +1 periods per week

Duration of SEE: 3 hours

CIE: 30 marks

SEE: 70 marks

Credits: 3

Objectives:

1. To understand fundamentals of object-oriented programming in Java which includes defining classes, invoking methods, difference between applet and application programs, using class libraries
2. To create Java application programs using sound OOP practices such as interfaces, exception handling, multi threading.
3. To understand fundamentals of object-oriented programming in Java which includes defining classes, invoking methods, difference between applet and application programs, using class libraries
4. Use Collection framework, AWT and event handling to solve real world problems.
5. Exploring Swing, and implementing Servlets.

Outcomes:

1. Achieve proficiency in object-oriented concepts and also learns to incorporate the same into the Java programming language.
2. Create Java application programs using sound OOP practices e.g. Inheritance, interfaces and proper program structuring by using packages, access control specifiers.
3. Understand and Implement the concepts of Exception Handling in JAVA.
4. Develop the ability to solve real-world problems through software development in high-level programming language using Large APIs of Java as well as the Java standard class library.
5. Understand File, Streams, Input and Output Handling in java.
6. Create graphical user interface and Applets in java as well as apply the knowledge of Event Handling.

UNIT – I

Object Oriented Programming: Principles, Benefits of Object Oriented Programming.

Introduction to Java: Java buzzwords, bytecode. Java Programming Fundamentals: Applet and Application program using simple java program, data types, variables, arrays, operators, expressions, control statements, type conversion and casting, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, introducing

Proposed for the academic years 2020-2024

access control, static, final, nested and inner classes, exploring string class, using command-linear arguments. Inheritance: Inheritance concept, types of inheritance, Member access rules, use of super and final. Polymorphism - dynamic binding, method overriding, abstract classes and methods.
<i>UNIT – II</i>
Interfaces: Defining an interface, implementing interfaces, extending interface. Packages: Defining, Creating and Accessing a Package, importing packages Exception handling: Benefits of exception handling, classification, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, built in exceptions, creating own exception sub classes Multithreading: Java Thread Model, The Main Thread, creating a Thread, creating multiple threads, using is Alive() and join(), thread priorities, synchronization, inter thread communication, deadlock
<i>UNIT – III</i>
Collections: Overview of Java Collection frame work, commonly used Collection classes – Array List, Linked List, Hash Set, Tree Set, Collection Interfaces – Collection, List, Set. Accessing Collection via iterator, working with Map. Legacy classes and interfaces – Vector, Hashtable, Stack, Dictionary, Enumeration interface. Other Utility classes: String Tokenizer, Date, Calendar, Gregorian Calendar, ScannerJava Input/Output: exploring java.io, Java I/O classes and interfaces, File, Stream classes, byte stream, character stream, serialization.
<i>UNIT – IV</i>
GUI Programming with java: The AWT class hierarchy, MVC architecture. Applet Revisited: Basics, architecture and skeleton, simple applet program. Event Handling: Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces. Handling mouse and keyboard events, Adapter classes. Database Programming using JDBC: Introduction to JDBC, JDBC Drivers & Architecture, CRUD operation Using JDBC, Connecting to non-conventional Databases.
<i>UNIT – V</i>
Exploring Swing: JLabel, ImageIcon, JTextField, the Swing buttons, JTabbedPane, JScrollPane, JList, JComboBox. Servlet: Life cycle, using tomcat, simple servlet, servlet API, javax.servlet package, reading servlet parameters, javax.servlet.http package, handling HTTP requests and responses

Suggested Readings:

1. Herbert Scheldt, "The Complete Reference Java, 7th Edition, Tata McGraw Hill, 2006.
2. James M Slack, Programming and Problem Solving with JAVA, Thomson Learning, 2002.
3. C Thomas Wu, An Introduction to Object Oriented Programming with Java 5th Edition, McGraw Hill Publishing, 2010.
4. H. M. Dietel and P. J. Dietel, Java How to Program, Sixth Edition, Pearson Education /PHI.

Course Code	Course Title				Core/Elective		
PC351CS	DATA STRUCTURES AND ALGORITHMS LAB				Core		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
-	L	T	D	P	25	50	1
	-	-	-	2			

Objectives:

1. To develop skills to design and analyse simple linear and nonlinear data structures, such as stacks, queues and lists and their applications.
2. To gain programming skills to implement sorting and searching algorithms
3. To Strengthen the ability to identify and apply the suitable data structures for the given real world problem
4. To Gain knowledge in practical applications of data structures

Outcomes:

After completing this course, the student will be able to:

1. Implement various data structures using arrays, linked lists
2. Develop ADT necessary for solving problems based on Stacks and Queues
3. Implement binary trees, general tree structures, advanced search trees, heaps, graphs.
4. Implement hash functions and handle collisions
5. Implement various kinds of sorting techniques and apply appropriate techniques for solving a given problem

List of Experiments (Using C programming Language):

1. Implementation of Stacks and Queues using Arrays.
2. Implementation of Circular Queue.
3. Implementation of Infix to Postfix Conversion, Postfix Expression Evaluation.
4. Implementation of Singly Linked List
5. Implementation of Doubly Linked List.
6. Implementation of Circular Linked List.
7. Implementation of Stacks, Queues using Linked Lists.
8. Implementation of Binary Search and Hashing
9. Implementation of Operations on Binary Tree (Delete Entire Tree, Copy Entire Tree, Mirror Image, Level Order, Search for a Node etc.)
10. Implementation of Tree Traversals on Binary Trees.
11. Implementation of Binary Search Tree. (Insertion, Deletion and Search operations)
12. Implementation of operations on AVL Trees.
13. Implementation of Traversal on Graphs.
14. Implementation of Prims and Kruskals Algorithm.
15. Implementation of Selection, Merge, Quick, Heap, and Insertion Sort.

ADVANCED COMPUTER SKILLS LAB

PC352CS

Instruction: 2 periods per week

CIE: 25 marks

Duration of SEE: 3 hours

SEE: 50 marks

Objectives:

1. To learn programming of python with a focus of basic structure
2. To gain programming skills of python using function and OOP concept
3. To gain practical knowledge of MATLAB toolkit along with operations in matrices and plotting 2D graph

Outcomes:

After completing this course, the student will be able to:
1. Implement basic syntax in Python
2. Analyze and implement different kinds of OOP concept in real world problems.
3. Implement MATLAB operations and graphic functions.

List of Programming Exercises:

1. Python Variables, Executing Python from the Command Line, Editing Python Files, Python Reserved Words.
2. Comments, Strings and Numeric Data Types, Simple Input and Output
3. Control Flow and Syntax, Indenting, if Statement, Relational Operators, Logical Operators, Bit Wise Operators, while Loop, break and continue, for Loop, Lists, Tuples, Sets, Dictionaries
4. Functions: Passing parameters to a Function, Variable Number of Arguments, Scope, Passing Functions to a Function, Mapping Functions in a Dictionary, Lambda, Modules, Standard Modules
5. OOP concepts: Classes, File Organization, Special Methods, Inheritance, Polymorphism, Special Characters, Character Classes, Quantifiers, Dot Character, Greedy Matches, Matching at Beginning or End, Match Objects, Compiling Regular Expressions
6. MATLAB Menus, Toolbars, Computing with MATLAB, Script Files and the Editor/Debugger, MATLAB help System
7. MATLAB controls: Relational Logical Variables. Conditional Statements: if – else – elseif, switch, Loops: for – while –break, continue. User-Defined Functions.
8. Arrays, Matrices and Matrix Operations Debugging MATLAB Programs. Working with Data Files, and Graphing Functions: XY Plots –Sub-plots

Suggested Readings:

1. Mark Summerfield," Programming in Python A Complete introduction to the Python Language", Addison-Wesley Professional,2009.
2. Martin C. Brown," PYTHON: The Complete Reference", McGraw-Hill,2001.
3. W.J. Palm III, Introduction to MATLAB 7 for Engineers, McGraw-Hill International Edition,2005.
4. Wesley J Chun," Core Python Applications Programming", Prentice Hall,2012
5. Hordeski, Michael F, HVAC Control in the New Millennium, Fairmont press, 2001
6. Bela G. Liptak, Process Control-Instrument Engineers Handbook, Chilton book co.

BASIC ELECTRONICS LAB

ES 351 EC

Instruction: 2 periods per week

CIE: 25 marks

Credits: 1

Duration of SEE: 3 hours

SEE: 50 marks

Objectives:

1.To understand the characteristics of diodes and transistor configurations
2.To understand the design concepts of biasing of BJT and FET
3.To understand the design concepts of feedback amplifiers and oscillators
4.To study the design concepts of OP Amp and data converters

Outcomes:

1.After completing this course, the student will be able to:
2.Ability to design diode circuits & understand the application of Zener diode.
3.Ability to analyse characteristics of BJTs & FETs.
4.Ability to understand the different oscillator circuits.
5.Ability to understand operation of HWR & FWR circuits with & without filters.
6.Ability to design Analog-to-Digital converters & Digital-to-Analog converters.

List of Experiments:

1	CRO-Applications, Measurements of R, L and C using LCR meter, Colour code method and soldering practice.
2	Characteristics of Semiconductors diode (Ge, Si and Zener)
3	Static Characteristics of BJT-Common Emitter
4	Static Characteristics of BJT-Common Base
5	Static Characteristics of FET
6	RC-Phase Shift Oscillator
7	Hartley and Colpitts Oscillators
8	Common Emitter Amplifier
9	Astable Multivibrator
10	Full-wave rectifier with and without filters using BJT
11	Operational Amplifier Applications
12	Strain Gauge Measurement
13	Analog-to-Digital and Digital to Analog Converters

Suggested Reading:

1	Maheshwari and Anand, <i>Laboratory Experiments and PSPICE Simulations in Analog Electronics</i> , 1st edition, Prentice Hall of India, 2006.
2	David Bell A., <i>Laboratory Manual for Electronic Devices and Circuits</i> , Prentice Hall of India, 2001.

OOP using JAVA Lab

PC 353 CS

Instruction: 2 periods per week

CIE: 25 marks

Credits: 1

Duration of SEE: 3 hours

SEE: 50 marks

Objectives:

The objectives of the course are to impart knowledge of:
1. To build software development skills using java programming for real world applications.
2. To implement frontend and backend of an application
3. To implement classical problems using java programming.

Outcomes:

After the completion of the course, the student will be able to:
1. Develop Java applications using the concepts of Inheritance, interfaces, packages, access control specifiers.
2. Implement the concepts of Exception Handling in java Applications.
3. Read and write data using different Java I/O streams.
4. Create graphical user interfaces and Applets by applying the knowledge of Event Handling.
5. Create robust applications using Java standard class libraries and retrieve data from a database with JDBC.
6. Ability to solve real-world problems by designing user friendly GUI with befitting backend through the APIs of Java.

List of Experiments:

1. Write a Java program to illustrate the concept of class with method overloading
2. Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use String Tokenizer class of java.util)
3. Write a Java program to illustrate the concept of Single level and Multi level Inheritance.
4. Write a Java program to demonstrate the Interfaces & Abstract Classes.
5. Write a Java program to implement the concept of exception handling.

Proposed for the academic years 2020-2024

6. Write a Java program to illustrate the concept of threading using Thread Class and runnable Interface.
7. Write a Java program to illustrate the concept of Thread synchronization.
8. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.
9. Write a Java program to illustrate collection classes like Array List, LinkedList, Tree map and Hash map.
10. Write a Java program to illustrate Legacy classes like Vector, Hash table, Dictionary & Enumeration interface.
11. Write a Java program to implement iteration over Collection using Iterator interface and List Iterator interface
12. Write a Java program that reads a file name from the user, and then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
13. Write a Java program to illustrate the concept of I/O Streams
14. Write a Java program to implement serialization concept
15. Write a Java applet program to implement Colour and Graphics class
16. Write a Java applet program for handling mouse & key events
17. Write a Java applet program to implement Adapter classes
18. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field to display the result.
19. Write an example for JDBC prepared statement with ResultSet
20. Write a Java Program to get primary key value (auto-generated keys) from inserted queries using JDBC
21. Write a Java Program to create a simple JList
22. Write a Java Program to create a simple checkbox using JCheckBox
23. Write a Java Program to create a checkbox and Item Listener toit.
24. 1. Write Servlet application to print current date &time 2. Html & Servlet Communication 3. Auto refresh a page 4. Demonstrate session tracking 5. Select record from database 6. Application for login page 7. Insert record into database 8. Count the visits on webpage 9. Insert teacher record in Database

BE(CSE) SEMESTER- IV with Effect from AY(21-22)

CSE: SEMESTER – IV

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/D	Contact Hrs/Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	HS 104 EG	Effective Technical Communication in English	3	-	-	3	30	70	3	3
2	HS 105 CM	Finance and Accounting	3	1	-	4	30	70	3	3
3	BS 207 MT	Mathematics – III (Probability & Statistics)	3	-	-	3	30	70	3	3
4	ES 305 EC	Signals and Systems	3	-	-	3	30	70	3	3
5	PC 401 CS	Operating Systems	3	-	-	3	30	70	3	3
6	PC 402 CS	Computer Organization	3	1	-	4	30	70	3	3
7	PC 403 CS	Database Management Systems	3	1	-	4	30	70	3	3
Practical/ Laboratory Courses										
8	PC 451 CS	Computer Organization Lab	-	-	2	2	25	50	3	1
9	PC 452 CS	Operating Systems Lab	-	-	2	2	25	50	3	1
10	PC 453 CS	Database Management Systems Lab	-	-	2	2	25	50	3	1
Total			21	3	06	30	285	640		24

ELECTIVE TECHNICAL COMMUNICATION IN ENGLISH

HS 104 EG

Instruction: 3 periods per week

CIE: 30 marks

Credits: 3

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

1. Features of technical communication
2. Types of professional correspondence
3. Techniques of report writing
4. Basics of manual writing
5. Aspects of data transfer and presentations

Outcomes:

On successful completion of the course, the students would be able to
1. Handle technical communication effectively
2. Use different types of professional correspondence
3. Use various techniques of report writing
4. Acquire adequate skills of manual writing

Enhance their skills of information transfer and presentations

<i>UNIT – I</i>
Definition and Features of Technical communication: Definition and features of technical communication (precision, relevance, format, style, use of visual aids), Differences between general writing and technical writing, Types of technical communication (oral and written)
<i>UNIT – II</i>
Technical Writing-I (Official correspondence): Emails, IOM, Business letters, Business proposals
<i>UNIT – III</i>
Technical writing-II (Reports): Project report, Feasibility report, Progress report, Evaluation report.
<i>UNIT – IV</i>
Technical writing- III (Manuals): Types of manuals, User manual, Product manual, Operations manual.
<i>UNIT – V</i>
Information Transfer and Presentations: Non-verbal (bar diagram, flow chart, pie chart, tree diagram) to verbal (writing), Verbal (written) to non-verbal, Important aspects of oral and visual presentations.

Suggested Readings:

1. Raman, Meenakshi & Sharma, Sangeeta. (2015). Technical communication: Principles and Practice, 3rd Edition, New Delhi
2. Rizvi, Ashraf, M. (2017). Effective Technical Communication (2nd ed.). New Delhi, Tata McGraw Hill Education <i>Security</i> .

- | |
|--|
| 3. Sharma, R. C., & Mohan, Krishna. (2017). Business Correspondence and Report Writing: A Practical Approach to Business & Technical Communication (4th ed.). New Delhi, Tata McGraw Hill Education. |
| 4. Tyagi, Kavita & Misra, Padma. (2011). Advanced Technical Communication. New Delhi, PHI Learning. |
| 5. Jungk, Dale. (2004). Applied Writing for Technicians. New York, McGraw-Hill Higher Education |

FINANCE AND ACCOUNTING

HS 105 CM

Instruction: 3 +1 periods per week

CIE: 30 marks

Credits: 3

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

1.To provide basic understanding of Financial and Accounting aspects of a business unit
2. To provide understanding of the accounting aspects of business
3. To provide understanding of financial statements
4. To provide the understanding of financial system
5. To provide inputs necessary to evaluate the viability of projects
6. To provide the skills necessary to analyse the financial statements

Outcomes:

After successful completion of the course the students will be able to
1. Evaluate the financial performance of the business unit.
2. Take decisions on selection of projects.
3. Take decisions on procurement offinances.
4. Analyse the liquidity, solvency and profitability of the business unit.
5. Evaluate the overall financial functioning of an enterprise.

UNIT – I

Basics of Accounting: Financial Accounting–Definition- Accounting Cycle – Journal - Ledger and Trial Balance-Cash Book-Bank Reconciliation Statement (including Problems)

UNIT – II

Final Accounts: Trading Account-Concept of Gross Profit- Profit and Loss Account-Concept of Net Profit- Balance Sheet (including problems with minor adjustments)

UNIT – III

Financial System and Markets: Financial System-Components-Role-Considerations of the investors and issuers- Role of Financial Intermediaries. Financial Markets-Players- Regulators and instruments - Money Markets Credit Market- Capital Market (Basics only)

UNIT – IV

Basics of Capital Budgeting techniques: Time Value of money- Compounding- Discounting- Future Value of single and multiple flows- Present Value of single and multiple Flows- Present Value of annuities- Financial Appraisal of Projects– Payback Period, ARR- NPV, Benefit Cost Ratio, IRR (simple ratios).

UNIT – V

Financial statement Analysis: Financial Statement Analysis- Importance-Users-Ratio Analysis-liquidity, solvency, turnover and profitability ratios.

Suggested Readings:

1. Satyanarayana. S.V. and Satish. D., Finance and Accounting for Engineering, Pearson Education
2. Rajasekharan, Financial Accounting, Pearson Education
3. Sharma. S.K. and Rachan Sareen, Financial Management, Sultan Chand
4. Jonathan Berk, Fundamentals of Corporate Finance, Pearson Education
5. Sharan, Fundamentals of Financial Management, Pearson Education

**B.E Mathematics syllabus for affiliated engineering colleges of
Osmania University
(w.e.f: academic year 2019-2020)
For Branches: CSE, IT, EEE & EIE**

BS 207MT	Mathematics-III: Probability and Statistics	2L:IT:OP	3 credits
-----------------	--	-----------------	------------------

Course objectives :

- To provide the knowledge of probability distributions , tests of significance, correlation and regression.

Course Outcomes :

At the end of the course students will be able to

- apply various probability distributions to solve practical problems, to estimate unknown parameters of populations and apply the tests of hypotheses
- perform a regression analysis and to compute and interpret the coefficient of correlation

Unit-I : Introduction of Probability, Conditional probability, Theorem of Total probability, Baye’s Theorem and its applications, Random variables, Types of random variables, Probability mass function and Probability density function, Mathematical expectations.

Unit-II: Discrete probability distributions: Binomial and Poisson distributions, Mean, variance, moment generating function and evaluation of statistical parameters for these distributions, Moments, skewness and Kurtosis.

Unit-III: Continuous probability distributions, Uniform, Exponentian and Normal distributions, Mean, variance, moment generating function and evaluation of statistical parameters for these distributions

Unit-IV: Curve fitting by the method of least squares : fitting of straight lines, second degree parabolas and more general curves, Correlation, regression and Rank correlation. Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations.

Unit-V : Test for single mean, difference of means and correlation coefficients, test for ratio of variances , Chi-square test for goodness of fit and independence of attributes.

Text / References:

1. Advanced Engineering Mathematics, R.K.Jain & Iyengar, Narosa Publications.
2. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 2000.
3. Engineering Mathematics, P.Sivaramakrishna Das & C.Vijaya Kumar, Pearson India Education Services Pvt.Ltd.
4. N.P. Bali and M. Goyal, "A text book of Engineering Mathematics", Laxmi Publications, 2010.
5. Fundamentals of Mathematical Statistics, S.C.Gupta & V.K.Kapoor, S.Chand Pub.
6. P. G. Hoel, S. C. Port and C. J. Stone, "Introduction to Probability Theory", Universal Book Stall, 2003.
7. W. Feller, "An Introduction to Probability Theory and its Applications", Vol. 1, Wiley, 1968.

SIGNALS AND SYSTEMS

ES305EC

Instruction: 3 periods per week

CIE: 30 marks

Credits: 3

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

- | |
|---|
| 1. To explain signals and systems representations/classifications and also describe the time and frequency domain analysis of continuous time signals with Fourier series, Fourier transforms and Laplace transforms. |
| 2. To understand Sampling theorem, with time and frequency domain analysis of discrete time signals with DTFS, DTFT and Z-Transform. |
| 3. To present the concepts of convolution and correlation integrals and also understand the properties in the context of signals/systems and lay down the foundation for advanced courses. |

Outcomes:

- | |
|---|
| 1. Define and differentiate types of signals and systems in continuous and discrete time |
| 2. Apply the properties of Fourier transform for continuous time signals |
| 3. Relate Laplace transforms to solve differential equations and to determine the response of the Continuous Time Linear Time Invariant Systems to known inputs |
| 4. Apply Z-transforms for discrete time signals to solve Difference equations |
| 5. Obtain Linear Convolution and Correlation of discrete time signals with graphical representation |

UNIT – I

Some useful operations on signals: Time shifting, Time scaling, Time inversion. Signal models: Impulse function, Unit step function, Exponential function, Even and odd signals. Systems: Linear and Non-linear systems, Constant parameter and time varying parameter systems, Static and dynamic systems, Causal and Non-causal systems, Lumped Parameter and distributed parameter systems, Continuous-time and discrete- time systems, Analog and digital systems.

UNIT – II

Fourier series: Signals and Vectors, Signal Comparison: correlation, Signal representation by orthogonal signal set, Trigonometric Fourier Series, Exponential Fourier Series, LTI system response to periodic inputs.

UNIT – III

Continuous-Time Signal Analysis: Fourier Transform: Aperiodic signal representation by Fourier integral, Fourier Transform of some useful functions, Properties of Fourier Transform, Signal transmission through LTI Systems, ideal and practical filters, Signal energy. Laplace transform: Definition, some properties of Laplace transform, solution of differential equations using Laplace transform.

UNIT – IV

Discrete-time signals and systems: Introduction, some useful discrete-time signal models, Sampling continuous-time sinusoids and aliasing, Useful signal operations, examples of discrete-time systems. Fourier analysis of discrete-time signals, periodic signal representation of discrete-time Fourier series, aperiodic signal representation by Fourier integral.

UNIT – V

Discrete-time signal analysis: Z-Transform, some properties of Z-Transform, Solution to Linear difference equations using Z-Transform, System realization. Relation between Laplace transform and Z-Transform. DTFT: Definition, Properties of DTFT, comparison of continuous-time signal analysis with discrete-time signal analysis.

Suggested Readings:

1. B. P. Lathi, Linear Systems and Signals, Oxford University Press, 2nd Edition, 2009
2. Alan V O P Penheim, A. S. Wlisky, Signals and Systems, 2nd Edition, PrenticeHall
3. Rodger E. Ziemer, William H Trenter, D. Ronald Fannin, Signals and Systems, 4th Edition, Pearson 1998.
4. Douglas K. Linder, Introduction to Signals and Systems, McGraw Hill, 1999
5. P. Ramakrishna Rao, Signals and Systems, TMH

OPERATING SYSTEMS

PC 401 CS

Instruction: 3 periods per week

CIE: 30 marks

Credits: 3

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

1. To learn the fundamentals of Operating Systems.
2. To learn the mechanisms of OS to handle processes and threads and their communication.
3. To learn the mechanisms involved in memory management in contemporary OS.
4. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection.
5. To know the components and management aspects of concurrency management.

Outcomes:

Student will be able to
1. Identify System calls and evaluate process scheduling criteria of OS.
2. Develop procedures for process synchronization of an OS.
3. Demonstrate the concepts of memory management and of disk management.
4. Solve issues related to file system interface and implementation, I/O systems.
5. Describe System model for deadlock, Methods for handling deadlocks.

UNIT – I

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.

UNIT – II

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling Criteria, Scheduling algorithms, multiprocessor scheduling.

UNIT – III

Process Synchronization: Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Peterson's Solution, classical problems of synchronization: The Bounded buffer problem, Producer\Consumer Problem, reader's & writer problem, Dining philosopher's problem. Semaphores, Event Counters, Monitors, Message Passing,

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Methods for Handling: Deadlocks: Deadlock prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

UNIT – IV

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation, fragmentation, and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, structure of page table, Protection and sharing, Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms, Trashing.

UNIT – V

I/O Hardware: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software,

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods, Free-space management, directory implementation, efficiency, and performance.

Secondary-Storage Structure: Disk structure, Disk scheduling algorithms, Disk Management, RAID structure.

Suggested Readings:

1. Avi Silberschatz, Peter Galvin, Greg Gagne, *Operating System Concepts Essentials*, 9th Edition, Wiley Asia Student Edition, 2017.
2. William Stallings, *Operating Systems: Internals and Design Principles*, 5th Edition, Prentice Hall of India, 2016.
3. Maurice Bach, *Design of the Unix Operating Systems*, 8th Edition, Prentice-Hall of India, 2009.
4. Daniel P. Bovet, Marco Cesati, *Understanding the Linux Kernel*, 3rd Edition, , O'Reilly and Associates.

COMPUTER ORGANIZATION

PC 402 CS

Instruction: 3+1 periods per week

CIE: 30 marks

Credits: 3

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

1. To provide in depth knowledge to the students about the design and organization of a digital computer, operation of various functional units, instruction set design and factors that influence the performance of a computer.
2. To enable the students with the understanding of basic computer architecture with instruction set and programming of 8085 in particular.
3. To learn the functionality and interfacing of various peripheral devices.

Outcomes:

After the completion of the course, the student will be able to:
1. To understand the architecture of modern computer, Bus structures.
2. Analyze the Different memories and evaluate the mapping techniques.
3. Discuss the architecture, the instruction set and addressing modes of 8085 processor.
4. Analyze Stacks, Subroutine, Interrupts of 8085, different PPI techniques, the uses of interfaces 8259, RS 232C, USART (8251), and DMA controller. Design the applications of interfacing circuit's 8254/8253timer, A/D and D/A converter, Keyboard/Display controller.

UNIT – I

Basic Structure of Computers: Computer Types, Functional Units, Basic Operational Concepts, Bus Structures, Performance, Multiprocessors and Multicomputers, Historical perspective.

Input/output Organization: Accessing I/O devices, Interrupts, Processor examples, Direct memory access, parallel interface and serial interface.

UNIT – II

The Memory System: Basic concepts, Semiconductor RAM memories, Read-Only memories, Speed, Size and Cost, Cache memories, Performance considerations, Virtual Memories, Memory management requirements, Secondary Storage.

UNIT – III

8085 Architecture: Introduction to microprocessors and microcontrollers, 8085 Processor Architecture, Internal operations, Instructions and timings. Programming the 8085 - Introduction to 8085 instructions, Addressing modes and Programming techniques with Additional instructions.

UNIT – IV

Stacks and subroutines, interfacing peripherals - Basic interfacing concepts, interfacing output displays, Interfacing input keyboards. Interrupts - 8085 Interrupts, Programmable Interrupt Controller (8259A). Direct Memory Access (DMA) - DMA Controller (Intel 8257), Interfacing 8085 with Digital to Analog and Analog to Digital converters.

UNIT – V

Programmable peripheral interface (Intel 8255A), Programmable communication interface (Intel 8251), Programmable Interval timer (Intel 8253 and 8254), Programmable Keyboard /Display controller (Intel 8279). Serial and parallel bus standards RS 232 C, IEEE 488.

Suggested Readings:

1. Carl Hamacher, Zvonko Vranesic, SafwatZaky, Computer Organization, 5th Edition, McGraw Hill, 2002.
2. Ramesh S Gaonkar, Microprocessor Architecture, Programming, and Applications with the 8085, 5/E Prentice Hall,2002.
3. Pal Chouduri, Computer Organization and Design, Prentice Hall of India,1994.
4. M. M. Mano, Computer System Architecture, 3rd Edition, PrenticeHall.

Database Management Systems

PC403CS

Instruction: 3+1 periods per week

CIE: 30 marks

Credits: 3

Duration of SEE: 3 hours

SEE: 70 marks

Objectives:

1. To get familiar with fundamental concepts of database management which includes database design, database languages, and database-system implementation.
2. To get familiar with data storage techniques and indexing.
3. To impart knowledge in transaction Management, concurrency control techniques and recovery techniques.
4. To master the basics of SQL and construct queries using SQL.
5. To become familiar with database storage structures and access techniques

Outcomes:

1. Develop the knowledge of fundamental concepts of database management and Designing a database using ER modeling approach.
2. Implement storage of data, indexing, and hashing.
3. Apply the knowledge about transaction management, concurrency control and recovery of database systems.
4. Ability to design entity relationship model and convert entity relationship diagrams into RDBMS and formulate SQL queries on the data

Apply normalization for the development of application software

<i>UNIT – I</i>
Introduction to Database: File System Organization: Sequential - Pointer - Indexed – Direct. Purpose of Database System - Database Characteristics - Users of Database System - Advantages of DBMS Approach - Schemas and Instances - Three Schema Architecture and Data Independence - The Database System Environment - Relational Algebra
<i>UNIT – II</i>
Logical Database Design: Relational DBMS - Codd's Rule - Entity-Relationship model - Extended ER Normalization - Functional Dependencies - Anomaly - 1NF to 5NF - Domain Key Normal Form – Denormalization.
<i>UNIT – III</i>
Indexing: Types of Single Level Ordered Indexes - Multilevel Indexes - Dynamic Multilevel Indexes. Transaction Processing and Concurrency Control: Transaction Concepts - ACID Properties - Transaction States - Concurrency Control Problems - Serializability - Recoverability - Pessimistic and Optimistic Concurrency Control Schemes.

UNIT – IV

Introduction to the Relational Model – Integrity Constraint Over relations – Enforcing Integrity constraints– Querying relational data – Logical data base Design – Introduction to views – Destroying /altering Tables and Views.

Relational Algebra – Selection and projection set operations – renaming – Joins – Division – Examples of Algebra overviews – Relational calculus – Tuple relational Calculus – Domain relational calculus – Expressive Power of Algebra and calculus

UNIT – V

Form of Basic SQL Query – Examples of Basic SQL Queries – Introduction to Nested Queries – Correlated Nested Queries Set – Comparison Operators – Aggregative Operators – NULL values – Comparison using Null values – Logical connectivity's – AND, OR and NOT – Impact on SQL Constructs – Outer Joins – Disallowing NULL values – Complex Integrity Constraints in SQL Triggers and Active Data bases.

Advanced Topics: Overview: Parallel Database - Multimedia Database - Mobile Database - Web Database - Multidimensional Database. Data Warehouse - OLTP Vs OLAP - NoSQL Database.

Suggested Readings:

1. Abraham Silberchatz, Henry F Korth and Sudarshan S, "Database System Concepts", Tata McGraw- Hill, New Delhi, 2010.
2. Ramez Elmasri and Shamkant B Navathe, "Fundamentals of Database Systems", Addison Wesley, USA, 2010.
3. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", Tata McGraw-Hill, New Delhi, 2008.
4. Gupta G K, "Database Management System", Tata McGraw-Hill, New Delhi, 2011.
5. Atul Kahate, "Introduction to Database Management Systems", Pearson Education, New Delhi, 2009

Practical / Laboratory Courses

Computer Organization Lab

PC 451 CS

Instruction: 2 periods per week

CIE: 25 marks

Credits: 1

Duration of SEE: 3 hours

SEE: 50 marks

Objectives:

The objectives of the course are to impart knowledge of:
1. To become familiar with the architecture and Instruction set of Intel 8085 microprocessor.
2. To provide practical hands on experience with Assembly Language Programming.
3. To familiarize the students with interfacing of various peripheral devices with 8085 microprocessors.

Outcomes:

After the completion of the course, the student will be able to:
1. Interpret the principles of Assembly Language Programming, instruction set in developing microprocessor based applications.
2. Develop Applications such as: 8-bit Addition, Multiplication, Division, array operations, swapping, negative and positive numbers.
3. Analyse the interfaces like serial ports, digital-to-analog Converters and analog-to-digital converters etc.
4. Build interfaces of Input-output and other units like stepper motor with 8085. Analyse the function of traffic light controller.

List of Programs:

1. Tutorials on 8085 Programming.
2. Interfacing and programming of 8255. (E.g. traffic light controller).
3. Interfacing and programming of 8254.
4. Interfacing and programming of 8279.
5. A/D and D/A converter interface.
6. Stepper motor interface.
7. Display interface

Note: Adequate number of programs covering all the instructions of 8085 instruction set should be done on the 8085 microprocessor trainer kit.

OPERATING SYSTEMS LAB

PC 452 CS

Instruction: 3 periods per week

CIE: 25 marks

Credits: 1

Duration of SEE: 3 hours

SEE: 50 marks

Objectives:

1.	Learn different types of CPU scheduling algorithms.
2.	Demonstrate the usage of semaphores for solving synchronization problem.
3.	Understand memory management techniques and different types of fragmentation.
4.	That occur in them and various page replacement policies.
5.	Understand Banker's algorithm used for deadlock avoidance.
6.	Learn various disk scheduling algorithms.

Outcomes:

Student will be able to
<ul style="list-style-type: none">• Evaluate the performance of different types of CPU scheduling algorithms.• Implement producer-consumer problem, reader-writers problem, Dining philosopher's problem.• Simulate Banker's algorithm for deadlock avoidance.• Implement paging replacement and disk scheduling techniques.• Use different system calls for writing application programs.

I. CASE STUDY

Perform a case study by installing and exploring various types of operating systems on a physical or logical (virtual) machine.

II. List of Experiments (preferred programming language is C)

1. Write C programs to implement UNIX system calls and file management
2. Write C programs to demonstrate various process related concepts.
3. Write C programs to demonstrate various thread related concepts.
4. Write C programs to simulate CPU scheduling algorithms: FCFS, SJF, Round Robin
5. Write C programs to simulate Intra & Inter-Process Communication (IPC) techniques: Pipes, Messages Queues, Shared Memory.
6. Write C programs to simulate solutions to Classical Process Synchronization Problems: Dining Philosophers, Producer-Consumer, Readers-Writers
7. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.
8. Write C programs to simulate Page Replacement Algorithms: FIFO, LRU
9. Write C programs to simulate implementation of Disk Scheduling Algorithms: FCFS, SSTF.

Software Required:

StarUML/Umbrello, NetBeans/Eclipse IDE, XAMPP/MEAN stack, JUnit, JMeter, Selenium, Bugzilla

Database Management Systems Lab

PC 453 CS

Instruction: 2 periods per week

Duration of SEE: 3 hours

CIE: 25 marks

SEE: 50 marks

Credits: 1

Objectives:

The objectives of the course are to impart knowledge of:
1. To practice various DDL commands in SQL
2. To write simple and Complex queries in SQL
3. To familiarize PL/SQL

Outcomes:

After the completion of the course, the student will be able to:
10. Design and implement a database schema for a given problem
11. Develop the query statements with the help of structured query language.
12. Populate and query a database using SQL and PL/SQL
13. Develop multi-user database application

Design GUI using forms and implement database connectivity:

List of Programs
1. Creation of database (exercising the commands for creation)
2. Simple condition query creation using SQL Plus
3. Complex condition query creation using SQL Plus
4. Usage of Triggers and Stored Procedures.
5. Creation of Forms for student Information, library information, Pay roll etc.
6. Writing PL/SQL procedures for data validation
7. Generation using SQL reports
8. Creating Password and Security features for applications.
9. Usage of File locking table locking, facilities in applications.
10. Creation of small full pledged database application spreading over to 3sessions.

Suggested Readings:

1. Nilesh Shah, Database System Using Oracle, PHI,2007.
2. Rick F Vander Lans, Introduction to SQL, Fourth edition, PearsonEducation,2007.
3. Benjamin Rosenzweig, Elena Silvestrova, Oracle PL/SQL by Example, Third edition, Pearson Education, 2004.
4. Albert Lulushi, Oracle Forms Developer's Handbook, Pearson Education,2006.

CSE: SEMESTER -V

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	D/P	Contact Hrs/Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	PC 501 CS	Software Engineering	3	1	-	4	30	70	3	3
2	PC 502 CS	Programming Languages	3	1	-	4	30	70	3	3
3	PC 503 CS	Automata Languages & Computation	3	1	-	4	30	70	3	3
4	PE-I	Professional Elective-I	3	-	-	3	30	70	3	3
5	PE-II	Professional Elective-II	3	-	-	3	30	70	3	3
6	PE-III	Professional Elective-III	3	-	-	3	30	70	3	3
Practical/Laboratory Courses										
7	PC 551 CS	Software Engineering Lab	-	-	2	2	25	50	3	1
8	PC 552 CS	Programming Languages Lab	-	-	2	2	25	50	3	1
9	PW 533 CS	Mini Project	-	-	4	4	25	50	3	2
Total			18	03	08	29	255	570		22

Profession Elective – I	
Course Code	Course Title
PE 511 CS	Artificial Intelligence
PE 512 CS	Advanced Computer Architecture
PE 513 CS	Network Security
PE 514 CS	Foundations of Cryptography

Profession Elective – II	
Course Code	Course Title
PE 531 CS	Web Technologies
PE 532 CS	Embedded Systems
PE 533 CS	Graph Theory
PE 534 CS	Data Analytics

Profession Elective – III	
Course Code	Course Title
PE 521 CS	Block Chain Technologies
PE 522 CS	Information Retrieval Systems
PE 523 CS	Soft Computing
PE 524 CS	Computer Graphics