Scheme of Instruction, Evaluation

And

Syllabi of

With effect from Academic Year 2023-24

B.E. COMPUTER SCIENCE AND ENGINEERING

III & IV Semesters

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

UNIVERSITY COLLEGE OF ENGINEERING

(Autonomous)

Esd.1917 **Hyderabad – 500 007, TS, INDIA** **Estd. 1929**

SCHEME OF INSTRUCTION AND EXAMINATION
B. E (CSE)
SEMESTER- III

SNo	Code	Course Title	Scheme of Instruction			Contact Hrs/Wk	Scheme of Examination			Credits
			L	T	P		Hrs	CIE	SEE	
Theory										
1	**MT 301 BS**	Engineering Mathematics – III (Numerical Methods)	3	0	-		3	40	60	3
2	**CS 301PC**	Computer Organization and Microprocessors	3	0	-		3	40	60	3
3	**CS 302 PC**	Data Structures	3	0	-		3	40	60	3
4	**CS 303 PC**	Object Oriented Programming using Java	3	0	-		3	40	60	3
5	**CS 304 PC**	Principles of Programming Languages	3	0	-		3	40	60	3
6	**EC 301 ES**	Basic Electronics	3	0	-		3	40	60	3
Practicals										
7	**CS 351PC**	Computer Organization and Microprocessors Lab	-	-	2		3	25	50	1
8	**CS 352PC**	Data Structures Lab	-	-	2x2		3	25	50	2
9	**CS 353PC**	Object Oriented Programming using Java Lab	-	-	2		3	25	50	1
		Total	**18**	**0**	**8**		**27**	**315**	**510**	**22**

MT301BS	**ENGINEERING MATHEMATICS-III (NUMERICAL METHODS)**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**-**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives:	
1	Apply general methodology to solve linear first order and second order partial differential equations
2	To study the classification of second order partial differential equations and solve them by using separation of variables methods
3	To introduce a few numerical methods to solve nonlinear algebraic and transcendental equations and system of linear equations
4	To provide the necessary basic concepts of numerical differentiation, numerical integration
5	To solve the initial volume problems.

Course Outcomes:	
On completion of this course, the student will be able to:	
CO-1	Find the solutions of first and second order PDE
CO-2	Find solutions of the heat equation, wave equation, and the Laplace equation subject to boundary conditions
CO-3	Solve nonlinear equations, system of linear equations
CO-4	Find Numerical Integration
CO-5	Perform numerical differentiation

UNIT – I
Definition of Partial Differential Equations, First order partial differential equations, Solutions of first order linear PDEs , Solution to homogenous and non-homogenous linear partial differential equations of second order by complimentary function and particular integral method.

UNIT – II
Second-order linear equations and their classification, Initial and boundary conditions, D'Alembert's solution of the wave equation, Heat diffusion and vibration problems, Separation of variables method to Solve simple problems in Cartesian coordinates. The one dimensional diffusion equation and its solution by separation of variables.

UNIT – III
Bisection method, Newton-Raphson method, Solution of linear system of equations- Gauss elimination method, LU decomposition method, Gauss-Jacobi and Gauss-Seidel iteration methods.

UNIT – IV
Interpolation, Lagrange's interpolation, Newton's divided difference interpolation, Newton's Forward and Backward difference interpolations. Numerical differentiation, Interpolation approach, Numerical integration-Trapezoidal rule,Simpson's 1/3 rule.

UNIT – V
Taylor's series method, Euler's method, Picard's method of successive approximations, Runge-Kutta method of 4th order

Suggested Reading:

1	R. K. Jain & S.R.K Iyengar, Advanced Engineering Mathematics, Narosa Publications, 4th Edition 2014 (Text Book).
2	Erwin Kreyszi, Advanced Engineering Mathematics, John Wiley, 9th Edition, 2012.
3	B.S. Grewal, Higher Engineering Mathematics, Khanna Publications, 43rd Edition, 2014.
4	M.K.Jain, S.R.K.Iyengar and R.K.Jain, N*umerical methods for scientific and engineering computation*, 6th Edition, New AgeInternationalLimited.,2012
5	B .V . Ramana, Higher Engineering Mathematics, 23rd reprint, 2015.
6	S.S.Sastry, Introductory Methods of Numerical Analysis,5th Edition, PHI Private Limited, 2012.
7	H.K. Dass, Er. Rajnish Varma, Higher Engineering Mathematics, S.Chand Technical 3rd Edition.

CS301PC	COMPUTER ORGANIZATION AND MICROPROCESSORS					
Prerequisites	Digital Logic Design		**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To understand the Instruction Set Architecture: Instruction format, types, various addressing modes.
2	To understand the basic components and design of the CPU, the ALU and control unit.
3	To understand the parallelism both in terms of a single processor and multiple processors.
4	To understand the 8085 and 8051 architecture To understand the 8085 and 8051 architecture.

Course Outcomes On completion of this course, the student will be able to	
CO1	Able to understand the Instruction Set Architecture: Instruction format, types, various addressing modes
CO2	Able to understand the basic components and design of the CPU: the ALU and control unit write multi threaded programs with synchronization.
CO3	Able to understand the parallelism both in terms of a single processor and multiple processors
CO4	Able to understand the 8085 anad 8051 architectures
CO5	Able to apply interfacing with I/O Organization, Interrupt-driven I/O, DMA

UNIT – I

Data Representation: Fixed and Floating Point representations.
Overview of Computer Function and Interconnections: Computer components, Interconnection structures, Bus interconnection, Bus structure, and Data transfer.
Register Transfer Microoperations: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic, Logic and Shift micro-operations, Arithmetic Logic Shift Unit.

UNIT – II

Basic Computer Organization and Design: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory reference instruction, Input-Output and Interrupt.
Micro programmed Control: Control memory, Address Sequencing, Microprogramexample, Design of Control Unit.

UNIT – III

Central Processing Unit: General Register Organization, Stack Organization, Instruction formats, Addressing modes, Data Transfer and Manipulation, and Program control.
Floating Point Arithmetic Operations. Pipeline Processing: Arithmetic, Instruction and RISC Pipelines.
Memory Organization: Cache memory, Virtual memory, Memory Management hardware

UNIT – IV
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. Input Output Organization: Modes of Transfer, Priority Interrupt, Direct Memory Access (DMA), I/O Processor. Basic Inter facing concepts with 8085, Programmable Interrupt Controller (8259A). Direct Memory Access(DMA) - DMA Controller (Intel 8257)

UNIT – V
Introduction to Microcontrollers, 8051 – Architecture, Instruction set, Addressing modes and Programming techniques. Comparison of various families of 8-bit micro controllers. System Design Techniques - Interfacing of LCD, ADC, Sensors, Stepper motor, Keyboard and DAC using microcontrollers. Communication Standards - Serial RS 232 and USB. Features of Multi-Core Processors architectures and Graphics Processing Units.

Suggested Reading:

1	Morris Mano M "Computer System Architecture", 3rd Edition, Pearson Education India, 2007.
2	William Stallings "Computer Organization and Architecture", PHI, 7th Edition, 2008.
3	Ramesh S. Gaonkar "Microprocessor Architecture, Programming, and Applications with the 8085", 5th Edition, Prentice Hall, 2002.
4	Myke Predko "Programming and Customizing the 8051 Microcontroller", Tata McGraw Hill, 1994.

CS 302 PC	**DATA STRUCTURES**					
Prerequisites	Programming for Problem Solving		**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To introduce the time and space complexities of algorithms.
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 techniques and their time complexities

Course Outcomes On completion of this course, the student will be able to	
CO1	Analyze the time and space complexities of algorithms.
CO2	Implement linear, non-linear data structures and balanced binary trees.
CO3	Analyze and implement various kinds of searching and sorting techniques.
CO4	Find a suitable data structure and algorithm to solve a real world problem.

UNIT – I

Performance and Complexity Analysis: Space Complexity, Time Complexity, Asymptotic Notation (Big-Oh), Complexity Analysis. Linear List-Array Representation: Vector Representation, Multiple Lists Single Array. Linear List-Linked Representation: Singly Linked Lists, Circular Lists, Doubly Linked Lists, Applications (Polynomial Arithmetic). Arrays and Matrices: Row And Column Major Representations, Sparse Matrices.

UNIT – II

Stacks: Array Representation, Linked Representation, Applications (Recursive Calls, Infix to Postfix, Postfix Evaluation). Queues: Array Representation, Linked Representation. Skip Lists and Hashing: Skip Lists Representation, Hash Table Representation, Application- Text Compression.

UNIT – III

Trees: Definitions and Properties, Representation of Binary Trees, Operations, Binary Tree Traversal. Binary Search Trees: Definitions, Operations on Binary Search Trees. Balanced Search Trees: AVL Trees, and B-Trees.

UNIT – IV

Graphs: Definitions and Properties, Representation, Graph Search Methods (Depth First Search and Breadth First Search) Application of Graphs: Shortest Path Algorithm (Dijkstra), Minimum Spanning Tree (Prim's and Kruskal's Algorithms).

UNIT – V
Searching: Linear Search and Binary Search Techniques and their complexity analysis. Sorting and Complexity Analysis: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, and Heap Sort.

Suggested Reading:

1	Sartaj Sahni, *Data Structures--Algorithms and Applications in C++*, 2nd Edition, Universities Press (India) Pvt. Ltd., 2005.
2	Mark Allen Weiss, *Data Structures and Problem Solving using C++*, Pearson Education International, 2003.
3	Michael T. Goodrich, Roberto Tamassia, David M. Mount, *Data Structures and Algorithms in C++*, John Wiley & Sons, 2010.

CS 303 PC	OBJECT ORIENTED PROGRAMMING USING JAVA					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To introduce fundamental object oriented concepts of Java programming Language-such as classes, inheritance packages and interfaces.
2	To introduce concepts of exception handling and multithreading.
3	To use various classes and interfaces in java collection framework and utility classes.
4	To understand the concepts of GUI programming using AWT and Swing controls.
5	To introduce Java I/O streams and serialization

Course Outcomes On completion of this course, the student will be able to	
CO1	Use object-oriented programming concepts to solve real world problems.
CO2	Demonstrate the behavior of programs involving constructs like string, arrays, garbage collection.
CO3	Understand the impact of exception handling to avoid abnormal termination of program and able to solve multi-threaded programs with synchronization.
CO4	implement real world applications using java collection frame work and I/O classes
CO5	Write Event driven GUI programs using AWT/Swing

UNIT – I

Object Oriented System Development: understanding object oriented development, understanding object oriented concepts, benefits of object oriented development.

Java Programming Fundamentals: Introduction, overview of Java, data types, variables and arrays, operators, control statements, classes, methods, inheritance, packages and interfaces.

UNIT – II

Exceptional Handling, Multithreaded Programming, I/O Basics, Reading Console Input and Output, Reading and Writing Files, Print Writer Class, String Handling.

UNIT – III

Exploring Java. Lang, Collections Overview, Collection Interfaces, Collection Classes, Iterators, Random Access Interface, Maps, Comparators, Arrays, Legacy Classes and Interfaces, String Tokenizer, Bit set, Date, Calendar, Observable Timer.

UNIT – IV

GUI Programming & Event Handling: Event Handling Mechanisms, The Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes, Introduction, AWT classes working with Graphics, Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout.

Java Swing: Basics of Swing, Difference between AWT & Swing, MVC Architecture, Components and Container, Exploring Swing Controls-J Label and Image Icon, J Text Field, The Swing Buttons-J Button, J Toggle Button, J Check Box, J Radio Button, J Tabbed Pane, J Scroll Pane, J List, J Combo Box, Swing Menus, Dialogs.

UNIT – V
Java I/O Classes and Interfaces, Files, Stream and Byte Classes, Character Streams, Serialization.

Suggested Reading:

1	Herbert Schildt, "The Complete Reference JAVA", Tata McGraw Hill ,7th Edition,2005.
2	James M Slack, "Programming and Problem Solving with JAVA", First Edition, Thomson Learning, 2002.
3	C.Thomas Wu, "An Introduction to Object-Oriented Programming with Java", Tata McGrawHill, 5th Edition, 2005

CS 304 PC	PRINCIPLES OF PROGRAMMING LANGUAGES					
Prerequisites			**L**	**T**	**P**	**C**
			3	**1**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To introduce the major programming paradigms, and the principles and techniques involved in design and implementation of modern programming languages
2	To introduce notations to describe syntax and semantics of programming languages
3	To analyze and explain behavior of simple programs in imperative languages using concepts such as binding, scope, control structures, subprograms and parameter passing mechanisms.
4	To introduce the concepts of ADT and object oriented programming for large scale software development.
5	To introduce the concepts of concurrency control and exception handling.

Course Outcomes On completion of this course, the student will be able to	
CO1	Understand the programming paradigms of modern programming languages
CO2	Describe syntax and semantics of programming languages
CO3	Analyze the behavior of simple programs in imperative languages.
CO4	Understand the concepts of ADT and object oriented programming for large scale software development
CO5	Understand the concepts of functional programming and logic programming

UNIT – I

Preliminary Concepts: Reasons for Studying Concepts of Programming Languages, Programming Domains, Language Evaluation Criteria, Influences on Language Design, Language Categories, Language Design Trade-offs, Implementation Methods, Programming Environments, Evolution of the Major Programming Languages.

Describing Syntax and Semantics: General Problem of Describing Syntax, Formal Methods of Describing Syntax, Attribute Grammars, Describing the Meaning of Programs.

UNIT – II

Names, Binding, Type Checking, and Scopes: Names, Variables, The Concept of Binding, Type Checking, Strong Typing, Type Compatibility, Scope, Scope and Lifetime, Referencing Environments, Named Constants, Data Types: Primitive Data Types, Character String Types, User- Defined Ordinal Types, Array Types , Associative Arrays, Record Types, Union Types, Pointer and Reference Types, optional types,

Expressions and Assignment Statements: Arithmetic Expressions, Overloaded Operators, Type Conversions, Relational and Boolean Expressions, Short-Circuit Evaluation, Assignment Statements, Mixed- Mode Assignment.

UNIT – III
Statement-Level Control Structures: Selection Statements, Iterative Statements, Unconditional Branching, Guarded Commands. **Subprograms:** Fundamentals and Design Issues for Subprograms, Local Referencing Environments, Parameter –Passing Methods, Parameters That are Subprograms Names, Overloaded Subprograms, Generic Subprograms, Design Issues for Functions, User-Defined Overloaded Operators. **Implementing Subprograms:** The General Semantics of Calls and Returns, Implementing "Simple" Subprograms, Implementing Subprograms with Stack-Dynamic Local Variables, Nested Subprograms, Blocks, Implementing Dynamic Scoping. **Abstract Data Types:** The Concept of Abstraction, Introduction to Data Abstraction, Design Issues for Abstract Data Types, Language Examples, Parameterized ADT, Encapsulation Constructs, Naming Encapsulation.

UNIT – IV
Object Oriented Programming: Design Issues, Object Oriented Programming in Smalltalk, C++, Java, C#, Ada 95, Ruby, The Object Model of JavaScript, Implementation of Object Oriented Constructs. **Concurrency:** Subprogram level Concurrency, Semaphores, Monitors, Massage Passing, Ada Support for Concurrency, Java Threads, C# Threads, Statement-Level Concurrency. Exception Handling and **Event Handling:** Introduction to Exception Handling, Exception Handling in Ada, C++ and Java, Introduction to Event Handling, Event Handling with Java.

UNIT – V
Functional Programming Languages: Introduction, Mathematical Functions, Fundamentals of FPL, LISP, Introduction to Scheme, COMMON LISP,ML, Haskell, Application of Functional Programming Languages and A Comparison of Functional and Imperative Languages, Functional interfaces (Java 8.0) **Logic Programming Languages:** Introduction to Predicate Calculus, Predicate Calculus and Proving Theorems, An Overview of Logic Programming. The Origins, Basic Elements and Deficiencies of Prolog, Applications of Logic Programming. **Scripting Languages:** Common Characteristics, Data Types , Object Orientation Names and Scopes, String and Pattern Manipulation Problem Domains, Scripting the World Wide Web

Suggested Reading:

1	"*Concepts of Programming Languages*" Robert.W.Sebesta 12th Edition, Pearson Education, 2019
2	"*Programming Language Pragmatics* " Michal Scott 4th Edition Morgan Kaufmann Publishers, 2015
3	*Java Precisely* Peter Sestoft 3rd Edition, MIT press 2016
4	*Programming Languages: Principles & Practices* Kenneth A. Lambert and Kenneth C. Louden, 3rd Edition, Cengage Learning 2012.
5	*Programming languages* , Watt, Wiley Dreamtech, First Edition , 2004.

EC 301 ES	**BASIC ELECTRONICS**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**1**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To study the characteristics of P-N Junction diodes and their parameters such as forward and reverse bias behavior, breakdown voltage, and rectification properties.
2	To understand the construction, working, and parameters of BJTs, Junction Field-Effect Transistors(JFETs).
3	To classify feedback amplifiers based on their configurations and characteristics.
4	To study the applications of OP Amps in inverting, non-inverting amplifiers, summer, integrator, and differentiator circuits.
5	To understand the principles of Photoelectric Devices (Photo diode and Photo Transistor), Light Emitting Diodes (LEDs), and Liquid Crystal Displays (LCDs).

Course Outcomes On completion of this course, the student will be able to	
CO1	Analyze the characteristics of P-N Junction diodes andunderstand their parameters
CO2	Apply the operation and characteristics of different BJT ,FETconfigurations and their use as amplifiers
CO3	Design and analyze inverting, non-inverting amplifiers,integrators, differentiators, and other OP Amp-based circuits
CO4	Design and analyze inverting, non-inverting amplifiers,integrators, differentiators, and other OP Amp-based circuits
CO5	Understand the working principles of different transducers and their applications in data acquisition system

UNIT – I
Semi-Conductor Theory: Energy Levels, Intrinsic and Extrinsic Semiconductors,Mobility, Diffusion and Drift current. Hall Effect, Characteristics of P-N Junctiondiode, Parameters and Applications. Rectifiers: Half wave and Full wave Rectifiers (Bridge, center tapped) with andwithout filters, ripple regulation and efficiency. Zener diode as voltage regulator.

UNIT – II
Bipolar Junction Transistor: BJT, Current components, CE, CB, CC configurations,characteristics, Transistor as amplifier. Analysis of CE, CB, CC Amplifiers (qualitative treatment only). JFET: Construction and working, parameters, CS, CG, CD Characteristics, CS amplifier.

UNIT – III
Feedback Concepts – Properties of Negative Feedback Amplifiers, Classification, Parameters. Oscillators – Barkhausen Criterion, LC Type and RC Type Oscillators andCrystal Oscillators. (Qualitative treatment only).

UNIT – IV
Operational Amplifiers – Introduction to OP Amp, characteristics and applications– Inverting and Non-inverting Amplifiers, Summer, Integrator, Differentiator, Instrumentation Amplifier. Digital Systems: Basic Logic Gates, half, Full Adder and Subtractors.

UNIT – V
Data Acquisition Systems: Study of transducer (LVDT, Strain gauge, Temperature,and Force). Photo Electric Devices and Industrial Devices: Photo diode, PhotoTransistor, LED, LCD, SCR, UJT Construction and Characteristics only. **Display Systems:** Constructional details of C.R.O and Applications.

Suggested Reading:

1	Jocob Millman, Christos C. Halkias and Satyabrata Jit, Electronics Devices and Circuits, 3rd Edition, McGraw Hill Education (India) Private Limited, 2010.
2	Rama Kanth A. Gaykward, Op-AMPS and Linear Integrated Circuit, 4th Edition, Prentice Hall of India, 2000
3	M. Morris Mano, Digital Design, 3rd Edition, Prentice Hall of India, 2002.
4	William D Cooper, and A.D. Helfrick, Electronic Measurements and Instrumentations Techniques, 2nd Edition, Prentice Hall of India, 2008.
5	S. Shalivahan, N. Suresh Kumar, A. Vallava Raj, Electronic Devices and Circuits, 2nd Edition., McGraw Hill Education (India) Private Limited, 2007.

CS 351 PC	COMPUTER ORGANIZATION AND MICROPROCESSORS LAB					
Prerequisites	Digital Logic Design		L	T	P	C
			-	-	2	1
Evaluation	CIE	25 Marks	SEE		50 Marks	

Course Objectives	
1	To Excel in logic design, spanning gates, flip-flops, multiplexers, decoders, encoders, counters, and shift-registers, using Verilog.
2	To Create BCD adders, adder/subtracter units, carry-look ahead adders, and counters, showcasing Verilog skills
3	To Create ALU and 4-bit processors using Verilog, gaining practical insights into processor architecture.
4	To Demonstrate 8085 Microprocessor programming and 8051 Microcontroller skills, including interfacing, A/D-D/A conversion, stepper motor control, and display interaction.
5	To Acquire practical experience in interfacing and programming devices like 8255, 8254, and 8279, integrating them with microprocessors and microcontrollers.

Course Outcomes On completion of this course, the student will be able to	
CO1	Showcase profound comprehension and practical implementation of logic design, effectively using Verilog for gates, flip-flops, and various IC chips.
CO2	Display adeptness in designing and implementing essential processor components .
CO3	Demonstrate the ability to write, execute, and debug programs for 8085 Microprocessor and 8051 Microcontroller
CO4	Illustrate hands-on mastery in interfacing and programming devices such as 8255, 8254, and 8279

List of Programs

PART A: Programs using VERILOG

1. Review of the different logic design ckts., a) Gates b) Flip/Flop(RS, JK, D, T),
2. Familiarity with state of art IC-chips, e.g. a) Multiplexer , b) Decoder, c) Encoder,
d) Counter, e)Shift-Register, f)adder Truth Table verification and clarification from Data-book.
3. Design a BCD adder.
4. Design an Adder/Subtracter composite unit
5. Design a carry-look ahead Adder
6. Design a ripple counter and carry-look ahead counter.
7. Design ALU and 4-bit processor

PART B: 8085 Programming using Microprocessor Trainer Kit

8. Simple programming examples using 8085 instruction set. To understand the use of various instructions and addressing modes.
9. Interfacing and programming of 8255
10. Interfacing and programming of 8254.

11. Interfacing and programming of 8279.

PART C: 8051 Programming

12. Simple programming examples using 8051 Microcontroller
13. A/D and D/A converter interface
14. Stepper motor interface
15. Display Interface

CS 352 PC	DATA STRUCTURES LAB					
Prerequisites	**Programming for Problem Solving**		**L**	**T**	**P**	**C**
			-	**-**	**2x2**	**2**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	To develop skills to design and analyze simple linear and non linear data structures.
2	To identify and apply the suitable data structure for the given real world problem.
3	To gain knowledge in practical applications of data structures

Course Outcomes On completion of this course, the student will be able to	
CO1	Design and analyze the time and space efficiency of the data structure
CO2	Identity the appropriate data structure for given problem
CO3	Have practical knowledge on the applications of data structures

List of Experiments

1) Implement the following operations on singly linked list:
i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Implement the following operations on doubly linked list:
i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Implement the following operations on circular linked list:
i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Implementation of Stacks, Queues (using both arrays and linked lists).
5. Implementation of circular queue using arrays.
6. Implementation of double ended queue (de queue) using arrays.
7. Implement a program to evaluate a given postfix expression using stacks.
8. Implement a program to convert a given infix expression to postfix form using stacks.
9. Implementation of Polynomial arithmetic using linked list.
10. Implementation of recursive and nonrecursive functions to perform the following searching operations for a key value in a given list of integers:
i) Linear search ii) Binary search
11. Implementation of hashing with (a) Separate Chaining and (b) Open addressing methods.
12. Implementation of recursive and iterative traversals on binary tree.
13. Implementation of operations on binary tree (delete entire tree, copy entire tree, mirror image, level order, search for a node etc.)
14. Implementation of the following operations on binary search tree (BST):
(a) Minimum key (b) Maximum key (c) Search for a given key (d) Delete a node with given key
15. Implement the following sorting algorithms:
a) Bubble sort b) Selection sort c) Insertion sort (d) Merge sort (e) Quick sort (f) Heap sort

CS 353 PC	OBJECT ORIENTED PROGRAMMING USING JAVA LAB					
Prerequisites			**L**	**T**	**P**	**C**
			-	**-**	**2**	**1**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	Ability to learn the concept of classes, inheritance and abstract classes
2	Learn to demonstrate multithreaded programs with synchronization
3	Demonstrate real world applications using java collection frame work and I/O classes.
4	Model Event driven GUI programs using AWT/Swing

Course Outcomes On completion of this course, the student will be able to	
CO1	Understand the OOPS features
CO2	Understand the usage of abstract classes and interfaces.
CO3	Write multi-threaded programs with synchronization.
CO4	Implement real world applications using java collection frame work and I/O classes
CO5	Write Event driven GUI programs using AWT/Swing

List of Programs

1. A program to illustrate the concept of class with constructors, methods and overloading.
2. A program to illustrate the concept of Inheritance and Dynamic polymorphism.
3. A program to show the concept of packages.
4. A program to illustrate the usage of interfaces and Abstract class.
5. A program to illustrate exception handling keywords.
6. A program to illustrate user define exception using stack.
7. A program to illustrate user define exception for evaluating a post fix expression.
8. A program to illustrate to handle string in java using String and String Buffer.
9. A program to illustrate manipulating array in java
10. A program to illustrate Multithreading.
11. A program to illustrate Thread synchronization.
12. A program to illustrate inter thread communication
13. A program using String tokenizer.
14. A program using Linked list class.
15. A program using Tree set class.
16. A program using Hash set and Iterator classes.
17. A program using Map classes.
18. A program using Enumeration and Comparator interfaces.
19. A program to illustrate Buffered I/O streams and Buffered reader.
20. Write a Java program to read text from file from a specify index or skipping byte using file Input stream.
21. Write a Java program to determine number of byte return to file using data output stream.
22. A program to illustrate Byte Array I/O Streams.
23. A program to illustrate the usage of Serialization.
24. An application involving GUI with different controls, menus and event handling.
25. A program to implement asimple calculator using grid layout manager.

26. A program to implement Recursive Fibonacci method using swing .
27. A program to display digital clock using swing .
28. A program to read from a file and write to a file using Applet.
29. A program to display a calendar using JCombo box.
30. A program to illustrate event listener interfaces.

SCHEME OF INSTRUCTION AND EXAMINATION
B. E (CSE)
SEMESTER – IV

SNo	Code	Course Title	Scheme of Instruction			Contact Hrs/Wk	Scheme of Examination			Credits
			L	T	P		Hrs	CIE	SEE	
Theory										
1	**MT401BS**	Engineering Mathematics – IV (Probability & Statistics)	3	0	-		3	40	60	3
2	**CS 401PC**	Automata Languages and Computation	3	0	-		3	40	60	3
3	**CS402PC**	Design and Analysis of Algorithms	3	0	-		3	40	60	3
4	**CS403 PC**	Operating Systems	3	0	-		3	40	60	3
5	**MC401HS**	Managerial Economics & Accountancy	3	0	-		3	40	60	3
6	**EC402ES**	Signals and Systems	3	0	-		3	40	60	3
7	**Professional Elective – I**									
	CS411PE	Data Analytics Using R	3	0	-		3	40	60	3
	CS412PE	Graph Theory								
	CS413 PE	Embedded systems and Micro-Controllers								
	CS 414 PE	Mobile Application Development								
	CS 415 PE	Simulation & Modeling								
Practicals										
8	**CS451PC**	Design and Analysis of Algorithms lab	-	-	2		3	25	50	1
9	**CS452PC**	Operating Systems Lab	-	-	2		3	25	50	1
		Total	**21**	**0**	**4**		**27**	**330**	**520**	**23**

MT401BS	**ENGINEERING MATHEMATICS-IV (PROBABILITY & STATISTICS)**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives:	
1	To Understand the statistical Techniques
2	To study continuous Random variables and their Properties
3	To study the statistical Parameters of three distributions
4	To study empirical laws and curve fitting
5	To introduce tests of significance

Course Outcomes:	
At the end of the course students will be able to	
CO-1	Solve the Problems of Skewness and Kurtosis , Poisson Approximation and other statistical Techniques
CO-2	Solve Continuous Random Variables, exponential and gamma densities
CO-3	Solve the Problems related to Probability distributions, Binomial, Poisson and Normal distributions
CO-4	Obtain empirical formulas of curve fitting , solve the problems of correlation regression and rank correlation
CO-5	Solve the problems of test of significance such as large sample test for single proportion etc.

UNIT – I
Measures of Central tendency, Moments, Skewness and Kurtosis, Discrete random variables, Independent random variables, The multinomial distribution, Poisson approximation to the binomial distribution, Infinite sequences of Bernoulli trials, Sums of independent random variables, Expectation of Discrete Random Variables, Variance of a sum.

UNIT – II
Continuous random variables and their properties, Distribution functions and densities,Normal, Exponential and gamma densities.

UNIT – III
Probability distributions, Binomial, Poisson and Normal-evaluation of statistical parameters for these three 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

UNIT – V
Test of significance, Large sample test for single proportion, Difference of proportions, Single mean, difference of means, and difference of standard deviations. Small Sample 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.

Suggested Reading:

1	R. K. Jain & S.R.K Iyengar, Advanced Engineering Mathematics, Narosa Publications, 4th Edition 2014.
2	Erwin Kreyszi, Advanced Engineering Mathematics, John Wiley, 9th Edition, 2012.
3	B.S. Grewal, Higher Engineering Mathematics, Khanna Publications, 43rd Edition, 2014.
4	S.CGupta&Kapoor: Fundamentals of Mathematical statistics, Sultan chand & sons, New Delhi.
5	B.V .Ramana, Higher Engineering Mathematics, 23rd reprint, 2015.
6	N.P. Bali and M.Goyal, A text book of Engineering Mathematics, Laxmi Publications, 2010.
7	S.Ross,"A First Course in Probability", Pearson Education India,2002

CS401PC	**AUTOMATA LANGUAGES AND COMPUTATION**					
Prerequisites	Data Structures		**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	Introduce the concept of formal specification of languages and different classes of formal languages
2	Discuss automata models corresponding to different levels of Chomsky hierarchy.
3	Understand the concept of computability and decidability

Course Outcomes On completion of this course, the student will be able to	
CO1	Design Finite State Machine, Pushdown Automata, and Turing Machine.
CO2	Determine a languages place in the Chomsky hierarchy (regular, context-free, recursively enumerable).
CO3	Convert among equivalently powerful notations for a language, including among DFAs, NFAs, and regular expressions, and between PDAs and CFGs
CO4	Explain why the halting problem has no algorithmic solution.

UNIT – I: Introduction, Finite state automata, Non-deterministic finite state automata, FA with ε-transitions, Regular expressions, FA with outputs, Applications of FA. Properties of regular sets-Pumping Lemma, Closure properties, Myhill-Nerode Theorem, Minimization of FA, Decision Algorithms.

UNIT –II: Context Free Grammars and Languages–Derivations, Parse-trees, Ambiguity in Grammars and Languages. Pushdown Automata–Definitions, The languages of PDA, Equivalence of PDAs and CFGs, Deterministic Pushdown Automata (DPDA).

UNIT – III: Properties of CFLs–Normal forms for CFGs, Pumping Lemma, Closure properties, Decision algorithms, Deterministic Context Free Languages, Predicting machines, Decision properties, LR(0) grammars, LR(0) and DPDA, LR(k) grammars.

UNIT – IV: Turing Machines–Introduction, Computational Languages and Functions, Techniques for construction of Turing machines. Modifications of TM, TM as enumerator, Restricted TM.

UNIT –V: Undecidability: Recursive and Recursively enumerable languages, UTM and Undecidable problem, Rice Theorem, Post‟s correspondence problem. Chomsky‟s Hierarchy – Regular grammars, Unrestricted grammar, CSL, Relationship between classes of languages.

Suggested Reading:

1	John E. Hopcroft, Jeffrey D. Ullman, *Introduction to Automata Theory, Languages and Computation*, Narosa, 1979.
2	Zvi Kohavi, *Switching and Finite Automata Theory*, TMH, 1976.

CS402PC	**DESIGN AND ANALYSIS OF ALGORITHMS**					
Prerequisites	Data Structures		**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	Analyze the asymptotic performance of algorithms.
2	Write rigorous correctness proofs for algorithms
3	Demonstrate a familiarity with major algorithms and data structures.
4	Apply important algorithmic design paradigms and methods of analysis.
5	Synthesize efficient algorithms in common engineering design situations.

Course Outcomes On completion of this course, the student will be able to	
CO1	Analyze the complexity of the algorithm in asymptotic notations.
CO2	Apply the various algorithm approaches based on the complexities and analyze the graph traversal techniques
CO3	Develop the dynamic programming algorithms, and analyze it to determine its computational complexity

UNIT – I
Introduction: Characteristics of algorithm, Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem

UNIT – II
Fundamental Algorithmic Strategies: Brute-Force, Greedy, Dynamic Programming, Branch and-Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack and Travelling Salesman problem.

UNIT – III
Graph and Tree Algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive Closure, Minimum Spanning Tree, Topological Sorting, Network Flow Algorithm.

UNIT – IV
Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-Complete problems and Reduction techniques.

UNIT – V
Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE.

Suggested Reading:

1	*Introduction to Algorithms*, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill, 4th Edition, 2002
2	*Fundamentals of Algorithms* – E. Horowitz, Satraj Sahani, Computer Science Press, 1997
3	*Algorithm Design*, First Edition, Jon Kleinberg and ÉvaTardos, Pearson, 2006
4	*Algorithm Design: Foundations, Analysis, and Internet Examples*, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley Publishers, 2006
5	*Algorithms - A Creative Approach*, 3rd Edition, UdiManber, Addison-Wesley, 1995.

CS 403 PC	**OPERATING SYSTEMS**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To introduce the concepts of OS structure and process synchronization
2	To study different memory management strategies
3	To familiarize the implementation of file system
4	To understand the principles of system security and protection
5	To discuss the design principles and structure of Windows 7 and Linux

Course Outcomes On completion of this course, the student will be able to	
CO1	Evaluate different process scheduling algorithms and mechanisms available in an OS to control access to resource.
CO2	Describe the steps in address translation and different page replacement strategies
CO3	Compare different file allocation methods and decide appropriate allocation strategy for given type of file

UNIT – I
Introduction to Operating Systems: OS structure and strategies, Process concepts, Multithreaded Programming, Process scheduling, Process synchronization, Deadlocks.

UNIT – II
Memory management strategies with example architectures: Swapping, Contiguous allocation,Paging, Segmentation, Segmentation with paging , Virtual memory management : Demand paging, Page replacement, Thrashing.

UNIT – III
File system interface: File concepts, Access methods and protection. File system implementation: File system structure, Allocation methods, Directory implementation of file systems, Mass storage structures, I/O systems

UNIT – IV
System Protection: Principles and Domain, Access Matrix and implementation, Access control and access rights, Capability based systems, Language based Protection. System Security: Problem, Program threats, cryptography, user authentication, implementing security defenses, Firewalling, Computer security Classification.

UNIT – V
Case Studies: The Linux System–Design principles, Kernel modules, Process management, Scheduling, Memory management, File systems, Input and Output, Inter process communication. Windows 7 –Design principles, System components, Terminal services and fast user switching File systems, Networking, Programmer interface

Suggested Reading:

1	Abraham Silberschatz, Peter B Galvin, *Operating System Concepts*, 9th Edition, Wiley, 2016
2	William Stallings, *Operating Systems-Internals and Design Principles*, 8th Edition, Pearson, 2014
3	Andrew S Tanenbaum, *Modern Operating Systems*, 4th Edition, Pearson, 2016.

MC401HS	**MANAGERIAL ECONOMICS & ACCOUNTANCY**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To learn important concepts of Managerial Economics and apply them to evaluate business decisions
2	To understand various parameters that determine the consumers' behavior.
3	To evaluate the factors that affect production.
4	To understand the concepts of capital budgeting and payback period.
5	To study the concepts of various book-keeping methods.

Course Outcomes On completion of this course, the student will be able to	
CO1	Demonstrate the ability to apply fundamental concepts of Managerial Economics to analyze and assess business decisions, considering economic principles and their implications
CO2	Gain insights into the diverse factors influencing consumers' behavior and decision-making processes, allowing the evaluation of market demand and consumer preferences.
CO3	Assess the multiple determinants impacting production processes, including resource allocation, technology, and costs, enabling effective production management decisions.
CO4	Understand and apply concepts related to capital budgeting, including the computation and analysis of payback periods, aiding in effective investment decision-making.
CO5	Acquire knowledge of various book-keeping methods, comprehending their significance and application in financial record-keeping and analysis for informed business decisions

UNIT – I

Meaning and Nature of Managerial Economics: Managerial Economics and its usefulness to Engineers, Fundamental Concepts of Managerial Economics-Scarcity, Marginalism, Equimarginalism, Opportunity costs, Discounting, Time Perspective, Risk and Uncertainty, Profits, Case study method.

UNIT – II

Consumer Behavior: Law of Demand, Determinants, Types of Demand; Elasticity of Demand (Price, Income and Cross-Elasticity); Demand Forecasting, Law of Supply and Concept of Equilibrium. (Theory questions and small numerical problem can be asked)

UNIT – III

Theory of Production and Markets: Production Function, Law of Variable Proportion, ISO quants, Economics of Scale, Cost of Production (Types and their measurement), Concept of Opportunity Cost, Concept of Revenue, Cost-Output relationship, Break-Even Analysis, Price - Output determination under Perfect Competition and Monopoly (theory and problems can be asked)

UNIT – IV

Capital Management: Significance, determination and estimation of fixed and working capital requirements, sources of capital, Introduction to capital budgeting, methods of payback and discounted cash flow methods with problems. (Theory questions and numerical problems on estimating working capital requirements and evaluation of capital budgeting opportunities can be asked)

UNIT – V
Book-keeping: Principles and significance of double entry book keeping, Journal, Subsidiary books, Ledger accounts, Trial Balance, concept and preparation of Final Accounts with simple adjustments, Analysis and interpretation of Financial Statements through Ratios. (Theory questions and numerical problems on preparation of final accounts, cash book, petty cash book, bank reconciliation statement, calculation of some ratios)

Suggested Reading:

1	Mehta P.L., Managerial Economics —Analysis, Problems and Cases , Sulthan Chand & Sons Educational Publishers, 2011
2	Maheswari S.N., Introduction to Accountancy , Vikas Publishing House, 2005
3	Pandey I.M., Financial Management , Vikas Publishing House, 2009

EC 402 ES	**SIGNALS AND SYSTEMS**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course 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.
4	To Understand Fourier Analysis of discrete-time signals, including the periodic signal representation using the Discrete-Time Fourier Series.
5	To Understand system realization in the Z-Transform domain.

Course Outcomes On completion of this course, the student will be able to	
CO1	Apply the knowledge of linear algebra topics like vector space, basis, dimension, inner product, norm and orthogonal basis to signals.
CO2	Analyze the spectral characteristics of continuous-time periodic and a periodic signals using Fourier analysis.
CO3	Understand the process of sampling and the effects of under sampling. Classify systems based on their properties and determine the response of LSI system using convolution.
CO4	Analyze system properties based on impulse response and Fourier analysis.
CO5	Apply the Laplace transform and Z- transform for analyze of continuous-time and discrete-time signals and systems

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, a periodic signal representation by Fourier integral.

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

Suggested Reading:

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, Prentice Hall.
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.

PROFESSIONAL ELECTIVE – I

CS 411 PE	DATA ANALYTICS USING R					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	Gain knowledge on statistical data manipulation and analysis.
2	To understand Loading and Handling Data in R, Exploring Data in R.
3	To understand Machine Learning Algorithms with R Programming.

Course Outcomes On completion of this course, the student will be able to	
CO1	Develop simple applications and perform data visualization in R.
CO2	Solve the problems on regression and time series using R.
CO3	Utilize R programming to perform text mining and parallel computing.
CO4	Apply machine learning algorithms on real-time data analytics problems in R.

UNIT – I

Introduction to R: History, Advantages of R over Other Programming Languages, IDES and Text Editors, Handling Packages in R, Working with Directory, Data Types in R, Commands for Data Exploration.

Loading and Handling Data in R: Introduction, Challenges of Analytical Data Processing, Expression, Variables and Functions, Missing Values, Treatment in R, "as" operator, Vectors, Matrices, Factors, List, Analytical Tasks, Methods for Reading Data.

Exploring Data in R: Introduction, Data Frames, R functions for understanding data in data frames, Load Data Frames, Exploring Data, Finding the missing values, Invalid Values and Outliers, Descriptive Statistics, Spotting Problems in Data with Visualization.

UNIT – II

Linear Regression using R: Introduction, Model Fitting, Linear Regression, Assumptions of Linear Regression, Validating Linear Assumptions, Case Study.

Logistic Regression: Introduction, What is Regression, Generalized Linear Models, Binary Logistic Regression, Diagnosing Logistic Regression, Multinomial Logistic Regression Models.

UNIT – III

Decision Tree: Introduction, What is Decision Tree, Decision Tree Representation in R, appropriate problems for Decision Tree Learning, Basic Decision Tree Learning Algorithm, Measuring Features, Issues in Decision Tree Learning.

Clustering : Basic Concepts in Clustering - Hierarchical Clustering - kmeans Algorithm - CURE Algorithm. Association Rules : Frequent Itemset - Mining Algorithm Interfaces - Auxiliary Functions - Sampling from Transaction - Generating Synthetic Transaction Data - Additional Measures of Interestingness.

UNIT – IV
Text Mining : Few Challenges - Text Mining in R - General Architecture of Text Mining Systems - Pre-processing of Documents in R - Core Text Mining Operations - Text Mining Query Languages - Mining Frequent Patterns, Associations, and Correlations - Frequent Itemsets, Closed Itemsets and Association Rules - Mining Methods - Pattern Evaluation Methods - Sentiment Analysis. **Parallel Computing with R :** Introduction of R Tool Libraries - Opportunities in HPC to Empower R - Support for Parallelism in R - Comparison of Parallel Packages in R.

UNIT – V
Time Series in R: Introduction, What is Time Series Data, Reading Time Series Data, Plotting Time Series Data, Forecasts Using Exponential Smoothing, ARIMA Models. **Case Study :** Log Analysis - Recommendation Engines - Audience/Customer Insights Analysis - In-store Customer Traffic Prediction - Insurance Fraud Detection - Personalised Product Recommendations - Making User Generated Content Valuable - Credit Card Spending by Customer Groups can be Identified by using Business Needs - Sales Forecasting.

Suggested Reading:

1	Seema Acharya, Data Analytics Using R, McGraw Hill Education (India) Private Limited, 2018.
2	Bharti Motwani, "Data Analytics with R", Wiley India Private Limited, 2019.
3	John Mount and Nina Zumel, Practical Data Science with R, 2nd edition, Willey, 2019.

CS 412 PE	GRAPH THEORY					
Prerequisites	**Data Structures and Discrete Mathematics**		**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To familiarize a variety of different problems in Graph Theory
2	To learn various techniques to prove theorems
3	To understand and analyze various graph algorithms

Course Outcomes On completion of this course, the student will be able to	
CO1	Write precise and accurate mathematical definitions of objects in graph theory
CO2	Validate and critically assess a mathematical proof
CO3	Develop algorithms based on diverse applications of Graphs in different domains

UNIT – I
Preliminaries: Graphs, isomorphism, subgraphs, matrix representations, degree, operations on graphs, degree sequences Connected graphs and shortest paths: Walks, trails, paths, connected graphs, distance, cutvertices, cut-edges, blocks, connectivity, weighted graphs, shortest path algorithms Trees: Characterizations, number of trees, minimum spanning trees.

UNIT – II
Special classes of graphs: Bipartite graphs, line graphs, chordal graphs Eulerian graphs: Characterization, Fleury‟s algorithm, chinese-postman-problem.

UNIT – III
Hamilton graphs: Necessary conditions and sufficient conditions Independent sets, coverings, matchings: Basic equations, matchings in bipartite graphs, perfect matchings, greedy and approximation algorithms

UNIT – IV
Vertex colorings: Chromatic number and cliques, greedy coloring algorithm, coloring of chordal graphs, Brook's theorem Edge colorings: Gupta-Vizing theorem, Class-1 graphs and class-2 graphs, equitable edge-coloring.

UNIT – V
Planar graphs: Basic concepts, Eulers formula, polyhedrons and planar graphs, characterizations, planarity testing, 5-color-theorem. **Directed graphs:** Out-degree, in-degree, connectivity, orientation, Eulerian directed graphs, Hamilton directed graphs, tournaments.

Suggested Reading:

1	F.Harry, *Graph Theory*, Narosa Publications, 1988.
2	C.Berge: *Graphs and Hypergraphs*, North Holland/Elsevier, 1973
3	J A Bondy and U.S. R Murthy, *Graph Theory with Applications*, Elsevier Science Ltd, 1976
4	Douglas B West, *Introduction to Graph Theory*, Prentice Hall, 2004.

CS 413 PE	EMBEDDED SYSTEMS AND MICRO CONTROLLERS					
Prerequisites	**Computer Organization and Microprocessors**		**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To learn the architecture and programming of typical microcontroller.
2	To introduce the basic concepts of small and medium scale embedded system design using microcontroller
3	Collect knowledge of architecture of ARM 7processor,
4	Understand the peripherals of LPC2148 microcontroller
5	Learn to design, construct, program, verify, analyze and troubleshoot 8051 andLPC2148b ARM controller C language programs and supporting hardware.

Course Outcomes On completion of this course, the student will be able to	
CO1	Describe the importance and function of each pin of AVR ATmega32 Microcontroller
CO2	Develop embedded C language programs for AVR Microcontroller
CO3	Able to explain the architecture and programming model of ARM 7TDMI
CO4	Ability to undertake problem identification, formulation and selection of an appropriate Microcontroller
CO5	Ability to design and build functional prototype for real world applications

UNIT – I
The Microcontroller AVR: Role of microcontrollers in embedded Systems. Overview of the AVR family, Architecture and instruction set of 8-bit AVR Microcontroller: AVR Microcontroller architecture: Registers, AVR status register, Memory Space, ATmega32 pin-configuration & function of each pin, Addressing mode and instruction set of AVR microcontroller, Data transfer, Arithmetic, Logic and Compare, Rotate and Shift, Branch and Call instructions, Bit manipulation instructions.

UNIT – II
Embedded C Programming for AVR: . AVR data types and assembler directives, AVR assembly language programs, AVR I/O Port Programming, Time delay loop, Simple C programs for general purpose I/O and bit addressability, Timers, Interrupts, serial port, Serial port Interfacing protocols, SPI, I2C, UART. C Language programming for peripherals.

UNIT – III
ARM Architecture: Difference between RISC and CISC, RISC Design philosophy, ARM Design Philosophy, History of ARM microprocessor, ARM processor family. The Acorn RISC Machine, ARM Core data flow model, Architectural inheritance, The ARM7TDMI programmer's model: General purpose registers, CPSR, SPSR, ARM memory map, data format,load and store architecture, ARM states, Interrupts and Exceptions.

UNIT – IV
LPC2148 ARM CPU and embedded C Programming for ARM: Salient features, applications, block diagram, memory mapping. Functional features of Interrupt controller, RTC, USB, UART,I2C, SPI, SSP controllers, watch dog timers and other system control units. C programs for General purpose I/O, general purpose timer, PWM Modulator, UART, I2C Interface, SPI Interface, ADC, DAC

UNIT – V
LPC 2148 – Peripherals : Pin Connect Block- Features, Register description with example. GPIO-Features, Applications, Pin description, Register description with examples. PLL-Features, block diagram, bit structure of PLLCON, PLLCFG, & PLLSTAT, and PLLFEED. PLL frequency Calculation-procedure for determining PLL settings, examples for PLL Configuration Timers-Features, applications, Architecture of timer module, register description, Simple C programs for application using -GPIO, PLL, Timer.

Suggested Reading:

1	Dhananjay Gadre, "Programming and Customizing the AVR Microcontroller", TMH,1st Edition, 2001
2	Andrew N. SLOSS , "ARM System Developer's guide-Designing and optimizing system software , Elsevier Publications, 2016
3	Atul P. Godse, "ARM controller", First edition, Technical Publications, 2020

CS 414 PE	MOBILE APPLICATION DEVELOPMENT					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	To impart knowledge on Android OS design and Features.
2	To know the android application components, user interface
3	To analyze persistent storages.

Course Outcomes On completion of this course, the student will be able to	
CO1	Understand the basics of Android devices and Platform
CO2	Acquire knowledge on basic building blocks of Android programming required for App development
CO3	Understand persistence Data storage mechanism in Android
CO4	Understand advanced application concepts like Animations and Google Maps services etc

UNIT – I

Introduction: Introduction to mobile application development, **Android platform**: Android platform features and architecture, versions, comparison added features in each version, ART (Android Runtime), ADB (Android Debug Bridge).

Development environment/IDE: Android studio and its working environment, gradle build system, emulator setup.

Application anatomy: Application framework basics: resources, layout, values, asset XML representation and generate R. Java file, Android manifest file, creating a simple application.

UNIT – II

ANDROID UI DESIGN

GUI for Android: Introduction to activities, activities life-cycle, **Intent: intent** object, intent filters, adding categories, linking activities, **Views and View Groups**: Basic views, picker views, adapter views, Menu, App Bar etc, basics of screen design; different layouts. App widgets.

Lollipop Material design: new themes, new widgets, Card layouts. Recycler View **Fragments**: Introduction to activities, activities life-cycle.

UNIT – III

DATA PERSISTENCE

Different Data persistence schemes: Shared preferences, File Handling, Managing data using Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference. **Content providers:** User content provider, Android in build content providers.

UNIT – IV
Background Running Process, Networking And Telephony Services **Services: introduction** to services – local service, remote service and binding the service, the communication between service and activity, Intent Service. **Multithreading**: Handlers, AsyncTask **Broadcast receivers:** Local Broadcast Manager, Dynamic broadcast receiver, System Broadcast. Pending Intent, Notifications **Telephony Manager**: Sending SMS and making calls.

UNIT – V
ADVANCED APPLICATIONS **Location based services**: Displaying Maps, Obtaining the Maps API Key , Displaying the Zoom Control, Changing Views, Navigating to a specific Location Getting the Location that was Touched Geocoding and Reverse Geocoding Getting location data and Monitoring a Location. **Mobile Application Development for iOS** Introduction, iOS Navigation and Interface Design, Persistent Data in iOS, Tables in iOS: Navigation and Information Display.

Suggested Reading:

1	Dawn Griffiths, David Griffiths, *"Head First Android Development" ,OReilly 2015.*
2	J.F. DiMarzio's, " Practical Android 4 Games Application Development", Apress, 2011
3	Professional Android 4 Application Development, Reto Meier, Wiley India, 2012.

CS 415 PE	**SIMULATION & MODELING**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**-**	**3**
Evaluation	**CIE**	**40 Marks**	**SEE**		**60 Marks**	

Course Objectives	
1	Define the basics of simulation modeling and replicating the practical situations in organizations
2	Generate random numbers and random variates using different techniques.
3	Develop simulation model using heuristic methods.
4	Analysis of Simulation models using input analyzer, and output analyzer
5	Explain Verification and Validation of simulation model.

Course Outcomes On completion of this course, the student will be able to	
CO1	Describe the role of important elements of discrete event simulation and modeling paradigm.
CO2	Conceptualize real world situations related to systems development decisions, originating from source requirements and goals.
CO3	Develop skills to apply simulation software to construct and execute goal-driven system models.
CO4	Interpret the model and apply the results to resolve critical issues in a real world environment.

UNIT – I

Introduction to Simulation: Simulation, Advantages, Disadvantages, Areas of application, System environment, components of a system, Model of a system, types of models, steps in a simulation study.
Simulation Examples: Simulation of Queuing systems, Simulation of Inventory System, Other simulation examples.

UNIT – II

General Principles: Concepts in discrete - event simulation, event scheduling/ Time advance algorithm, simulation using event scheduling.
Random Numbers: Properties, Generations methods, Tests for Random number- Frequency test, Runs test, Autocorrelation test.

UNIT – III

Random Variate Generation: Inverse Transform Technique- Exponential, Uniform, Weibull, Triangular distributions, Direct transformation for Normal and log normal Distributions, convolution methods- Erlang distribution, Acceptance Rejection Technique Optimisation Via Simulation: Meaning, difficulty, Robust Heuristics, Random Search.

UNIT – IV

Analysis of Simulation Data Input Modelling: Data collection, Identification and distribution with data, parameter estimation, Goodness of fit tests, Selection of input models without data, Multivariate and time series analysis.
Verification and Validation of Model – Model Building, Verification, Calibration and Validation of Models.

UNIT – V
Output Analysis – Types of Simulations with Respect to Output Analysis, Stochastic Nature of output data, Measures of Performance and their estimation, Output analysis of terminating simulation, Output analysis of steady state simulations. Simulation Softwares: Selection of Simulation Software, Simulation packages, Trend in Simulation Software.

Suggested Reading:

1	Jerry Banks, John S Carson, II, Berry L Nelson, David M Nicol, Discrete Event system Simulation, Pearson Education, Asia, 4th Edition, 2007.
2	Geoffrey Gordon, System Simulation, Prentice Hall publication, 2nd Edition, 1978.
3	Averill M Law, W David Kelton, Simulation Modelling & Analysis, McGraw Hill International Editions – Industrial Engineering series, 4th Edition, 2018
4	Narsingh Deo, Systems Simulation with Digital Computer, PHI Publication (EEE), 3rd Edition, 2004,

CS 451 PC	**DESIGN AND ANALYSIS OF ALGORITHMS LAB**					
Prerequisites			**L**	**T**	**P**	**C**
			-	**-**	**2**	**1**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	To learn the importance of designing an algorithm in an effective way by consideringspace and time complexity
2	To learn graph search algorithms.
3	To study network flow and linear programming problems
4	To learn the dynamic programming design techniques.
5	To develop recursive backtracking algorithms.

Course Outcomes On completion of this course, the student will be able to	
CO1	Design an algorithm in an effective manner
CO2	Apply iterative and recursive algorithms.
CO3	Design iterative and recursive algorithms.
CO4	Implement optimization algorithms for specific applications.
CO5	Design optimization algorithms for specific applications.

List of Programs
1. Write a program to find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm. .
2. Write a program to find the shortest path in graph using Dijkstra's algorithm.
3. Write a program that implements N Queen's problem using backtracking algorithm.
4. Write a program to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using backtracking principle.
5. Write a program to implement dynamic programming algorithm to solve all pairs shortest path problem.
6. Write a program to solve 0/1 knapsack problem using Greedy algorithm.
7. Write a program to solve 0/1 knapsack problem using Dynamic programming algorithm.
8.Write a program to solve 0/1 knapsack problem using Backtracking algorithm
9.Write a program to solve 0/1 knapsack problem using Branch and bound algorithm.
10.Write a program that uses dynamic programming algorithm to solve the optimal binary search tree
11.Write a program for solving traveling sales persons problem using Dynamic programming algorithm.
12. Write a program for solving traveling sales persons problem using The back tracking algorithm.
13. Write a program for solving traveling sales persons problem using Branch and Bound.
14. Write a program to obtain the Topological ordering of vertices in a given digraph using Warshall's algorithm.
15.Write a program to compute the transitive closure of a given directed graph using Warshall's algorithm.
16.Write a program to print all the nodes reachable from a given starting node in a digraph using BFS method.332q
17. Write a program to check whether a given graph is connected or not using DFS method.
18. Write a program to find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.

CS 452 PC	**OPERATING SYSTEMS LAB**					
Prerequisites			**L**	**T**	**P**	**C**
			-	**-**	**2**	**1**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	To learn shell programming and the use of filters in the LINUX environment
2	To practice multithreaded programming
3	To implement CPU Scheduling Algorithms and memory management algorithms

Course Outcomes On completion of this course, the student will be able to	
CO1	Write shell scripts for simple system administration tasks
CO2	Write concurrent programs with synchronization constricts
CO3	Compare the performance of various CPU Scheduling Algorithm
CO4	Critically analyze the performance of the various Memory management algorithms

List of Programs
1-3. Memory Management Algorithms
4-5. Examples of Multithreading
6. Producer & Consumer problem using Semaphores and shared memory
7-8.Processor Scheduling algorithms
9. Dining Philosophers problem using Semaphores
10.Readers and Writers problem using Semaphores
11.Shell-programming exercises

****