FACULTY OF ENGINEERING

Scheme of Instruction & Examination

(AICTE Model Curriculum for the Academic Year 2021-2022)

and

Syllabi

B.E. III and IV Semester

of

Four Year Degree Programme

in

Computer Science and Engineering (Data Science)

(With effect from the academic year 2021–2022)

(As approved in the faculty meeting held on XX-XX-XX)

Issued by

Dean, Faculty of Engineering

Osmania University, Hyderabad – 500 007

2021-22

SCHEME OF INSTRUCTION & EXAMINATION
B.E. (Computer Science and Engineering – Data Science) III – SEMESTER

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	**MC112CE**	**Environmental Science**	**2**	**-**	**-**	**2**	**30**	**70**	**3**	**-**
2	**MC113PY**	**Essence of Indian TraditionalKnowledge**	**2**	**-**	**-**	**2**	**30**	**70**	**3**	**-**
3	HS103ME	Operations Research	3	1	-	4	30	70	3	3
4	ES306EC	Basic Electronics	3	-	-	3	30	70	3	3
5	PC301CD	Data Structures and Algorithms	3	1	-	4	30	70	3	3
6	PC302CD	Discrete Mathematics	3	-	-	3	30	70	3	3
7	PC303CD	Programming Languages	3	-	-	3	30	70	3	3
8	PC304CSD	Python Programming	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
9	PC351CD	Data Structures and Algorithms using C Lab	-	-	2	2	25	50	3	1
10	PC352CD	Python Programming Lab	-	-	2	2	25	50	3	1
11	ES351EC	Basic Electronics Lab	-	-	2	2	25	50	3	1
			22	**02**	**06**	**30**	**315**	**710**		**21**

HS: Humanities and Social Sciences BS: Basic Science ES: Engineering Science
MC: Mandatory Course PC: Professional Core
L: Lecture T: Tutorial P: Practical D: Drawing
CIE: Continuous Internal Evaluation SEE: Semester End Evaluation (Univ. Exam)
PY: Philosophy, BZ: Biology/ Life Sciences, CE: Civil Engineering, CS: Computer Science and Engineering
EC: Electronics and Communication Engineering, ME: Mechanical Engineering.

Note:

1. Each contact hour is a clock hour
2. The duration of the practical class is two hours, however it can be extended wherever necessary, to enable the student to complete the experiment.
3. All the mentioned **Mandatory Courses** should be offered either in I–Semester or II–Semester only **from the academic year 2021-2022**.
4. For those of the students admitted during the academic year 2020-2021, since the Mandatory Courses were not offered during the I–Semester or II–Semester, they should be offered either in III–Semester or IV–Semester of the **academic year 2021-2022.**

Course Code	Course Title						Core/Elective
MC802CE	**Environmental Science**						**Mandatory**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**2**	**-**	**-**	**-**	**30**	**70**	**-**

Course Objectives

- To create awareness and impart basic knowledge about the environment and its allied problems.
- To know the functions of ecosystems.
- To understand importance of biological diversity.
- To study different pollutions and their impact on environment.
- To know social and environment related issues and their preventive measures.

Course Outcomes

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

1. Adopt environmental ethics to attain sustainable development.
2. Develop an attitude of concern for the environment.
3. Conservation of natural resources and biological diversity.
4. Creating awareness of Green technologies for nation's security.
5. Imparts awareness for environmental laws and regulations.

UNIT-I

The Multidisciplinary Nature of Environmental Studies: Definition, scope and importance, need for public awareness.

Natural Resources: Water Resources – Use and over utilization of surface and ground water, flood, drought, conflicts over water, Dams: Benefits and Problems. Food Resources –World Food Problems, effects of modern agriculture, fertilizer-pesticides problems, water logging, salinity, Forest Resources –Use and over exploitation, deforestation & its effect on tribal people. Land Resources –Land Degradation, environmental effect of mining, man induced landslides, soil erosion and desertification**.** Energy Resources –Growing energy needs, Renewable and Non-renewable energy resources.

UNIT-II

Ecosystems: Concept of an ecosystem, Structure and function of an ecosystem, Producers, consumers and decomposers, Energy flow in ecosystem, food chains, ecological pyramids, ecological succession, types of ecosystems (marine, pond, river, forest, grassland, desert)

UNIT-III

Biodiversity: Levels of Biodiversity, Bio-geographical classification of India, Value of biodiversity, Threats to biodiversity, endangered and endemic species of India, Conservation of biodiversity, global and national efforts.

UNIT-IV

Environmental Pollution: Definition, Causes, effects and control measures of air pollution, water pollution, soil pollution, noise pollution, thermal pollution, solid waste management.

Environment Protection Act: Air, water, forest and wildlife Acts, issues in the enforcement of environmental legislation.

UNIT-V

Social Issues and the Environment: Watershed management and environmental ethics. Climate change, global warming, acid rain, ozone layer depletion.

Environmental Disaster Management: Types of disasters, impact of disasters on environment, infrastructure, and development. Basic principles of disaster mitigation, disaster management, and methodology. Disaster management cycle and disaster management in India.

Field Work:

- Visit to a local area to document environmental issues- agricultural area/ pond/lake/terrestrial ecosystem
- Visit to a local polluted area- market/slum area/Industrial area/traffic area

Suggested Readings:

1. A.K. De, *Environmental Chemistry,* Wiley Eastern Ltd.
2. E.P. Odum, *Fundamentals of Ecology,* W.B. Sunders Co., USA.
3. M.N. Rao and A.K. Datta, *Waste Water Treatment,* Oxford and IBK Publications.
4. Benny Joseph, *Environmental Studies,* Tata McGraw Hill, 2005.
5. V.K. Sharma, *Disaster Management,* National Centre for Disaster Management, IIPE,1999.

Course Code	Course Title						Core/Elective
MC803PY	**Essence of Indian Traditional Knowledge**						**Mandatory**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**2**	**-**	**-**	**-**	**30**	**70**	**-**

Course Objectives

The course will introduce the students to

- To get a knowledge in Indian Philosophical Foundations.
- To Know Indian Languages and Literature and the fine arts in India & Their Philosophy.
- To explore the Science and Scientists of Medieval and Modern India

Course Outcomes

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

1. Understand philosophy of Indian culture.
2. Distinguish the Indian languages and literature among difference traditions.
3. Learn the philosophy of ancient, medieval and modern India.
4. Acquire the information about the fine arts in India.
5. Know the contribution of scientists of different eras.
6. The essence of Yogic Science for Inclusiveness of society.

UNIT – I

Introduction to Indian Philosophy: Basics of Indian Philosophy, culture, civilization, culture and heritage, general characteristics of culture, importance of culture in human literature, Indian culture, Ancient Indian, Medieval India, Modern India.

UNIT – II

Indian Philosophy & Literature: Vedas Upanishads, schools of Vedanta, and other religion Philosophical Literature. Philosophical Ideas the role of Sanskrit, significance of scriptures to current society, Indian Philosophies, literature of south India.

Indian languages and Literature-II: Northern Indian languages & Philosophical & cultural & literature.

UNIT – III

Religion and Philosophy: Religion and Philosophy in ancient India, Religion and Philosophy in Medieval India, Religious Reform Movements in Modern India (selected movements only)

UNIT – IV

Indian Fine Arts & Its Philosophy (Art, Technology & Engineering): Indian Painting, Indian handicrafts, Music, divisions of Indian classic music, modern Indian music, Dance and Drama, Indian Architecture (ancient, medieval and modern), Science and Technology in Indian, development of science in ancient, medieval and modern Indian.

UNIT – V

Education System in India: Education in ancient, medieval and modern India, aims of education, subjects, languages, Science and Scientists of Ancient India, Scientists of Medieval India, Scientists of Modern India. The role Gurukulas in Education System, Value based Education.

Suggested Readings:

1. Kapil Kapoor, "Text and Interpretation: The India Tradition'', ISBN: 81246033375, 2005
2. "Science in Samskrit", Samskrita Bharti Publisher, ISBN-13:978-8187276333,2007
3. NCERT, "Position paper on Arts, Music, Dance and Theatre'', ISBN 81-7450-494-X, 2006
4. S. Narain, "Examination in Ancient India", Arya Book Depot, 1993
5. Satya Prakash, "Founders of Sciences in Ancient India", Vijay Kumar Publisher, 1989
6. M.Hiriyanna, "Essentials of Indian Philosophy", Motilal Banarsidass Publishers, ISBN-13: 978-8120810990,2014
7. Chatterjee. S & Dutta "An Introduction to Indian Philosophy"

<table>
<tr><td>Course Code</td><td colspan="6">Course Title</td><td>Core/Elective</td></tr>
<tr><td><b>HS103ME</b></td><td colspan="6"><b>Operations Research</b></td><td><b>Core</b></td></tr>
<tr><td rowspan="2">Prerequisite</td><td colspan="4">Contact Hours per Week</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td><td rowspan="2">Credits</td></tr>
<tr><td>L</td><td>T</td><td>D</td><td>P</td></tr>
<tr><td><b>-</b></td><td><b>3</b></td><td><b>1</b></td><td><b>-</b></td><td><b>-</b></td><td><b>30</b></td><td><b>70</b></td><td><b>3</b></td></tr>
</table>

Course Objectives

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

Course Outcomes

After completing this course, the 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
5. Prepare the students to have the knowledge of Sequencing model at the end student would able to develop optimum model for job scheduling.
6. Prepare students to understand Queuing theory concepts and various optimization techniques at the end students would able to develop models for waiting line cases.

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 x n and m x 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. 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.
6. Data Reconciliation by Prof. Shanker Narasimha

Course Code	Course Title						Core/Elective
ES306EC	**Basic Electronics**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

The objectives of this course is to impart knowledge

- To understand the characteristics of diodes and transistor configurations
- To understand the design concepts of biasing of BJT and FET
- To understand the design concepts of feedback amplifiers and oscillators
- To study the design concepts of OP Amp and data converters

Course Outcomes

After completing this course, 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, Thermocouple, 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 D and David Cooper, *Modern Electronic Instrumentation and Measurements Techniques,* 1st edition, Prentice Hall of India, 2006.
3. Salivahanan, Suresh Kumar and Vallavaraj, *Electronic Devices and Circuits,* 2nd edition, Tata McGraw-Hill, 2010.

Course Code	Course Title						Core/Elective
PC301CD	**Data Structures and Algorithms**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**1**	**-**	**-**	**30**	**70**	**3**

Course Objectives

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

Course 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 ofan algorithm- Time and Space Complexity, Asymptotic Notations.

Arrays: Arrays-ADT, Polynomials, Sparse matrices, Strings-ADT, Pattern Matching.

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 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 Readings:

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), PHl.
3. Gilberg R. F and Forouzan B. A, Data structures: A Pseudo code Approach with C, Second Edition (2007), CengageLeaming
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, CliffordStein, Introduction to Algorithms, Third Edition (2009), MITPress
6. Yedidyah Langsam, Moshe J.Augenstein ,Aaron M. Tenenbaum, Data Structures Using C and C++ , Second Edition(2009),PHI

Course Code	Course Title						Core/Elective
PC302CD	**Discrete Mathematics**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To Learn mathematical concepts as applied in computer science for solving logical problems.
- To model relationships, analyse data, apply probability concepts and use functions to solve problems.
- To develop the mathematical skills needed for advanced quantitative courses.

Course Outcomes

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

1. Apply Propositional and Predicate logic for a variety of problems in various domains.
2. Understand Set Theory, Venn Diagrams, relations, functions and apply them to Real-world scenarios.
3. Model and solve the real world problems using Generating Functions and Recurrence Relations.
4. To identify the basic properties of graphs and trees and use these concepts to model simple applications.
5. Understand General properties of Algebraic systems and study lattices as partially ordered sets and their applications.
6. Apply the knowledge and skills obtained to investigate and solve a variety of discrete mathematics problems.

UNIT – I

Logic – Sets and Functions – Logic, Propositional equivalences – Predicates and quantifiers – Nested Quantifiers-Sets-Set Operations, Functions.

Algorithms- Integers – Matrices: Algorithms, Complexity of Algorithms. The Integers and Division, Integers and Algorithms, Applications of Number Theory, Matrices.

UNIT – II

Mathematical Reasoning, Induction, and Recursion: Proof Strategy, Sequence and Summation, Mathematical Induction, Recursive Definitions and Structural Induction, Recursive Algorithms.

Counting – Basics, Pigeonhole principle, Permutations and combinations – Binomial Coefficients, Generalized Permutations and combinations, Generating permutations and combinations.

UNIT – III

Discrete Probability: An Introduction to Discrete Probability theory, Expected Value and Variance.

Advanced Counting Techniques: Recurrence relations – Solving Recurrence Relations, - Divide and conquer relations – and Recurrence Relations, Generating function – Inclusion – Exclusion – Applications of Inclusion – Exclusion.

UNIT – IV

Relations: Relations & their Properties, n-ray relations and applications, Representing relations – Closures, equivalence relations, partial orderings.

Graphs: Introduction, Graph terminology, representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamiltonian paths, Shortest path problems, Planar graphs, Graph colouring.

UNIT –V

Trees: Introduction to Trees, Application of Trees, Spanning Trees, Minimum Spanning Trees.
Boolean Algebra: Boolean function, Representing Boolean functions, Logic Gates

Suggested Readings:

1. Kenneth H. Rosen – Discrete Mathematics and its Application – 5th Edition, McGraw Hill, 2003.
2. J. K. Sharma, Discrete Mathematics, Second Edition, Macmillan, 2005.
3. J.P. Tremblay, R. Manohar, Discrete Mathematical Structure with Application to Computer Science, McGraw Hill – 1997.
4. Joel. Mott. Abraham Kandel, T.P. Baker, Discrete Mathematics for Computer Scientist & Mathematicians, Prentice Hail N.J., 2nd Edition, 1986.

Course Code	Course Title						Core/Elective
PC303CD	**Programming Languages**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To briefly describe various programming paradigms.
- To provide conceptual understanding of High level language design and implementation.
- To introduce the power of scripting languages.
- To provide an introduction to formalisms for specifying syntax and semantics of programming languages.
- To provide an exposure to core concepts and principles in contemporary programming languages.
- To analyse and optimize the complexity of the programming languages.

Course Outcomes

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

1. Ability to express syntax and semantics in formal notation.
2. Ability to apply suitable programming paradigm for the application.
3. Gain Knowledge and comparison of the features programming languages
4. program in different language paradigms and evaluate their relative benefits.
5. Identify and describe semantic issues associated with variable binding, scoping rules, parameter passing, and exception handling.
6. Understand the design issues of object-oriented and functional languages.

UNIT- I

Preliminary Concepts: Reasons for studying, concepts of programming languages, Programming domains, Language Evaluation Criteria, influences on Language design, Language categories, Programming Paradigms – Imperative, Object Oriented, functional Programming, Logic Programming. Programming Language Implementation – Compilation and Virtual Machines, programming environments. Syntax and Semantics: general Problem of describing Syntax and Semantics, formal methods of describing syntax - BNF, EBNF for common programming languages features, parse trees, ambiguous grammars, attribute grammars, denotational semantics and axiomatic semantics for common programming language features.

UNIT- II

Data types: Introduction, primitive, character, user defined, array, associative, record, union, pointer and reference types, design and implementation uses related to these types. Names, Variable, concept of binding, type checking, strong typing, type compatibility, named constants, variable initialization. Expressions and Statements: Arithmetic relational and Boolean expressions, Short circuit evaluation mixed mode assignment, Assignment Statements, Control Structures – Statement Level, Compound Statements, Selection, Iteration, Unconditional Statements, guarded commands.

UNIT-III

Subprograms Blocks and Fundamentals of sub-programs: Scope and lifetime of variable, static and dynamic scope, Design issues of subprograms and operations, local referencing environments, parameter passing methods, overloaded sub-programs, generic sub-programs, parameters that are subprogram names, design issues for functions user defined overloaded operators, co routines.

UNIT- IV

Abstract types: Data Abstractions and encapsulation, introductions to data abstraction, design issues, language examples, C++ parameterized ADT, object oriented programming in small talk, C++, Java, C#, Ada 95 Concurrency: Subprogram level concurrency, semaphores, monitors, massage passing, Java threads, C# threads. Exception handling: Exceptions, exception Propagation, Exception handler in Ada, C++ and Java. Logic Programming Language: Introduction and overview of logic programming, basic elements of prolog, application of logic programming.

UNIT- V

Functional Programming Languages: Introduction, fundamentals of FPL, LISP, ML, Haskell, application of Functional Programming Languages and comparison of functional and imperative Languages. Scripting Language: Pragmatics, Key Concepts, Case Study: Python – Values and Types, Variables, Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation, Module Library.

Suggested Readings:

1. Concepts of Programming Languages Robert W. Sebesta 8/e, Pearson Education,2008.
2. Programming Language Design Concepts, D. A. Watt, Wiley dreamtech, rp-2007
3. Programming Languages, 2nd Edition, A.B. Tucker, R.E. Noonan, TMH.
4. Programming Languages, K. C.Louden, 2nd Edition, Thomson, 2003.
5. LISP, Patric Henry Winston and Paul Horn, Pearson Education.
6. Programming in Prolog, W.F. Clocksin,& C.S.Mellish, 5th Edition, Springer.
7. Programming Python, M.Lutz, 3rd Edition, O'reilly, SPD, rp-2007.
8. Core Python Programming, Chun, II Edition, Pearson Education, 2007.
9. Guide to Programming with Python, Michael Dawson, Thomson, 2008

Course Code	Course Title						Core/Elective
PC304CD	**Python Programming**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

The Objectives of Python Programming are

- To learn about Python programming language syntax, semantics, and the runtime environment
- To be familiarized with universal computer programming concepts like data types, containers
- To be familiarized with general computer programming concepts like conditional execution, loops & functions
- To be familiarized with general coding techniques and object-oriented programming

Course Outcomes

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

- Develop essential programming skills in computer programming concepts like data types, containers
- Apply the basics of programming in the Python language
- Solve coding tasks related conditional execution, loops
- Solve coding tasks related to the fundamental notions and techniques used in object oriented programming

UNIT- I

Introduction: Introduction to Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, Operators. Type conversions, Expressions, More about Data Output.

Data Types, and Expression: Strings Assignment, and Comment, Numeric Data Types and Character Sets, Using functions and Modules.

Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables. Repetition Structures: Introduction, while loop, for loop, calculating a Running Total, Input Validation Loops, Nested Loops.

UNIT- II

Control Statement: Definite iteration for Loop Formatting Text for output, Selection if and if else Statement Conditional Iteration The While Loop

Strings and Text Files: Accessing Character and Substring in Strings, Data Encryption, Strings and Number Systems, String Methods Text Files.

UNIT-III

List and Dictionaries: Lists, Defining Simple Functions, Dictionaries Design with Function: Functions as Abstraction Mechanisms, Problem Solving with Top Down Design, Design with Recursive Functions, Case Study Gathering Information from a File System, Managing a Program’s Namespace, Higher Order Function.
Modules: Modules,StandardModules,Packages.

UNIT- IV

File Operations: Reading config files in python, Writing log files in python, Understanding read functions, read(), readline() and readlines(), Understanding write functions, write() and writelines(), Manipulating file pointer using seek, Programming using file operations

Object Oriented Programming: Concept of class, object and instances, Constructor, class attributes and destructors, Real time use of class in live projects, Inheritance , overlapping and overloading operators, Adding and retrieving dynamic attributes of classes, Programming using Oops support Design with Classes: Objects and Classes, Data modelling Examples, Case Study An ATM, Structuring Classes with Inheritance and Polymorphism

UNIT- V

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Redefined Clean-up Actions.

Graphical User Interfaces: The Behaviour of Terminal Based Programs and GUI -Based, Programs, Coding Simple GUI-Based Programs, Other Useful GUI Resources. Programming: Introduction to Programming Concepts with Scratch.

Suggested Readings:

1. Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage.
2. Python Programming: A Modern Approach, VamsiKurama, Pearson.
3. Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.
4. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

Course Code	Course Title						Core/Elective
PC351CD	**Data Structures and Algorithm using C Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

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

Course 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.

Programming Exercise 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.

Course Code	Course Title						Core/Elective
PC352CD	**Python Programming Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- Introducing a new object oriented programming
- Enabling students to learn Big Data, Machine Learning etc.
- Preparing students to cope up with new Market tendencies
- To learn programs in MATLAB environment
- To handle Functions, Polynomials by using MATLAB commands
- Ability to solve any Mathematical functions
- To learn Mathematical Modeling in a new approach
- To plot Graphics (2-D) easily and effectively

Course 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 2 10. 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. Allen B Downey," Think Python", O'Reilly, 2012.

6. Stormy Attaway, "MATLAB: A Practical Introduction to Programming and Problem Solving".3rd Edition.

Course Code	Course Title						Core/Elective
PC353EC	**Basic Electronics Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- To understand the characteristics of diodes and transistor configurations
- To understand the design concepts of biasing of BJT and FET
- To understand the design concepts of feedback amplifiers and oscillators
- To study the design concepts of OP Amp and data converters

Course Outcomes

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

1. Ability to design diode circuits & understand the application of Zener diode.
2. Ability to analyse characteristics of BJTs & FETs.
3. Ability to understand the different oscillator circuits.
4. Ability to understand operation of HWR & FWR circuits with & without filters.
5. Ability tom 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 Readings:

1. Maheshwari and Anand, *Laboratory Experiments and PSPICE Simulations in Analog Electronics,* 1st edition, Prentice Hall of India, 2006.

David Bell A., *Laboratory Manual for Electronic Devices and Circuits,* Prentice Hall of India, 2001.

SCHEME OF INSTRUCTION & EXAMINATION
B.E. (Computer Science and Engineering – Data Science) IV – SEMESTER

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	**MC801PO**	**Indian Constitution**	**2**	**-**	**-**	**2**	**30**	**70**	**3**	**-**
2	HS104EG	Effective Technical Communication in English	3	-	-	3	30	70	3	3
3	HS105CM	Finance and Accounting	3	-	-	3	30	70	3	3
4	BS205MT	M-III (Probability & Statistics)	3	1	-	3	30	70	3	3
5	ES305EC	Signals and Systems	3	-	-	3	30	70	3	3
6	PC401CD	OOP using JAVA	3	-	-	3	30	70	3	3
7	PC402CD	Operating systems	3	1	-	3	30	70	3	3
8	PC403CD	Database Management Systems	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
9	PC451CD	Operating Systems Lab	-	-	2	2	25	50	3	1
10	PC452CD	OOP using JAVA Lab	-	-	2	2	25	50	3	1
11	PC453CD	Database Management Systems Lab	-	-	2	2	25	50	3	1
			23	**02**	**06**	**31**	**315**	**710**		**24**

HS: Humanities and Social Sciences BS: Basic Science ES: Engineering Science
MC: Mandatory Course PC: Professional Core
L: Lecture T: Tutorial P: Practical D: Drawing
CIE: Continuous Internal Evaluation SEE: Semester End Evaluation (Univ. Exam)
PO: Political Science, EG: English, CM: Commerce, MT: Mathematics,
CS: Computer Science and Engineering, EC: Electronics and Communication Engineering,

Note:

1. Each contact hour is a clock hour
2. The duration of the practical class is two hours, however it can be extended wherever necessary, to enable the student to complete the experiment.
3. All the mentioned **Mandatory Courses** should be offered either in I–Semester or II–Semester only **from the academic year 2021-2022**.
4. For those of the students admitted during the academic year 2020-2021, since the Mandatory Courses were not offered during the I–Semester or II–Semester, they should be offered either in III–Semester or IV–Semester of the **academic year 2021-2022.**
5. The students have to undergo a Summer Internship of two-week duration after IV – Semester and credits will be awarded in V – Semester after evaluation.

<table>
<tr><td>Course Code</td><td colspan="6">Course Title</td><td>Core/Elective</td></tr>
<tr><td><b>MC801PO</b></td><td colspan="6"><b>Indian Constitution</b></td><td><b>Mandatory</b></td></tr>
<tr><td rowspan="2">Prerequisite</td><td colspan="4">Contact Hours per Week</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td><td rowspan="2">Credits</td></tr>
<tr><td>L</td><td>T</td><td>D</td><td>P</td></tr>
<tr><td><b>-</b></td><td><b>2</b></td><td><b>-</b></td><td><b>-</b></td><td><b>-</b></td><td><b>30</b></td><td><b>70</b></td><td><b>-</b></td></tr>
</table>

Course Objectives

- To create awareness among students about the Indian Constitution.
- To acquaint the working conditions of union, state, local levels, their powers and functions.
- To create consciousness in the students on democratic values and principles articulated in the constitution.
- To expose the students on the relations between federal and provincial units.
- To divulge the students about the statutory institutions.

Course Outcomes

After completing this course, the student will

1. Know the background of the present constitution of India.
2. Understand the working of the union, state and local levels.
3. Gain consciousness on the fundamental rights and duties.
4. Be able to understand the functioning and distribution of financial resources between the centre and states.
5. Be exposed to the reality of hierarchical Indian social structure and the ways the grievances of the deprived sections can be addressed to raise human dignity in a democratic way.

UNIT-I

Evolution of the Indian Constitution: 1909 Act, 1919 Act and 1935 Act. Constituent Assembly: Composition and Functions; Fundamental features of the Indian Constitution.

UNIT-II

Union Government: Executive-President, Prime Minister, Council of Minister
State Government: Executive: Governor, Chief Minister, Council of Minister
Local Government: Panchayat Raj Institutions, Urban Government

UNIT-III

Rights and Duties: Fundamental Rights, Directive principles, Fundamental Duties

UNIT-IV

Relation between Federal and Provincial units: Union-State relations, Administrative, legislative and Financial, Inter State council, NITI Ayog, Finance Commission of India

UNIT-V

Statutory Institutions: Elections-Election Commission of India, National Human Rights Commission, National Commission for Women

Suggested Readings:

1. Abhay Prasad Singh & Krishna Murari, Constitutional Government and Democracy in India, Pearson Education, New Delhi, 2019
2. D.D. Basu, Introduction to the constitution of India, Lexis Nexis, New Delhi
3. Subhash Kashyap, Our Parliament, National Book Trust, New Delhi
4. Peu Ghosh, Indian Government &Politics, Prentice Hall of India, New Delhi
5. B.Z. Fadia & Kuldeep Fadia, Indian Government &Politics, Lexis Nexis, New Delhi

Course Code	Course Title					Core/Elective	
HS104EG	**Effective Technical Communication in English**					**Core**	
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

To expose the students to:

- Features of technical communication
- Types of professional correspondence
- Techniques of report writing
- Basics of manual writing
- Aspects of data transfer and presentations.

Course 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
5. Enhance their skills of information transfer and presentations

UNIT-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)

UNIT-II

Technical Writing-I (Official correspondence): Emails, IOM, Business letters, Business proposals.

UNIT-III

Technical writing-II (Reports): Project report, Feasibility report, Progress report, Evaluation report.

UNIT-IV

Technical writing- III (Manuals): Types of manuals, User manual, Product manual, Operations manual.

UNIT-V

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 ed.). New Delhi, OUP.
2. Rizvi, Ashraf, M. (2017). *Effective Technical Communication* (2nd ed.). New Delhi, Tata McGraw Hill Education.
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.
   Jungk, Dale. (2004). *Applied Writing for Technicians*. New York, McGraw-Hill Higher Education.

<table>
<tr><td>Course Code</td><td colspan="6">Course Title</td><td>Core/Elective</td></tr>
<tr><td><b>HS105CM</b></td><td colspan="6"><b>Finance and Accounting</b></td><td><b>Core</b></td></tr>
<tr><td rowspan="2">Prerequisite</td><td colspan="4">Contact Hours per Week</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td><td rowspan="2">Credits</td></tr>
<tr><td>L</td><td>T</td><td>D</td><td>P</td></tr>
<tr><td><b>-</b></td><td><b>3</b></td><td><b>-</b></td><td><b>-</b></td><td><b>-</b></td><td><b>30</b></td><td><b>70</b></td><td><b>3</b></td></tr>
</table>

Course Objectives

The course will introduce the students

- To provide basic understanding of Financial and Accounting aspects of a business unit
- To provide understanding of the accounting aspects of business
- To provide understanding of financial statements
- To provide the understanding of financial system
- To provide inputs necessary to evaluate the viability of projects
- To provide the skills necessary to analyse the financial statements

Course 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 of finances.
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

Course Code	Course Title						Core/Elective
BS205MT	**M-III (Probability and Statistics)**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**1**	**-**	**-**	**30**	**70**	**3**

Course Objectives

1. Interpret the measures of central tendency and dispersion.
2. Distinguish between explanatory and response variables and analyze data using correlation and regression.
3. Apply various probability distributions.
4. Apply tests of hypothesis.
5. 5. Employ basic analysis of time series data.

Course Outcomes

The expected outcomes of the Course are:

1. Compute and interpret descriptive statistics.
2. Evaluate random processes which occur in engineering applications governed by the Binomial, Poisson, Normal and Exponential distributions.
3. Fit the models using Regression Analysis.
4. Apply Inferential Statistics to make predictions or judgments about the population from which the sample data is drawn.
5. Interpret Time series data.

Unit I : Random Variables, Basic Statistics, Correlation and Regression

Notion of Randomness, Random Experiment, Random variables – Discrete and Continuous, Probability mass function and density function, constants of r.v.s (Mean, Variance, Monents about mean), Concept of Bivariate distributions and Covariance. Measures of central tendency and moments. Correlation : Karl-Pearson's correlation coefficient and Spearman's Rank correlation, Statements of their properties and problems, Simple and Multiple Linear Regression (three variables case only), Statements of properties of Regression coefficients and problems.

Unit II : Probability Distributions Discrete Distributions:

Binomial and Poisson distributions - definition, real life examples, Statements of their Mean and Variance, related problems, evaluation of statistical parameters. Continuous Distributions: Normal, Exponential and Gamma distributions - definition, real life examples, Statements of their Mean and Variance and related problems, evaluation of statistical parameters for Normal distribution.

Unit III : Testing of Hypothesis-1 (Large sample)

Concept of Sampling distribution and Standard error, tests for single proportion, difference of proportions, single mean, difference of means and Chi-square test for independence of attributes. Estimation of confidence interval for population mean and population proportions.

Unit IV : Testing of Hypothesis-2 (Small Sample)

Tests for single mean, difference of means, Population variance, ratio of variances, ANOVA 1-way and 2- way. Estimation of confidence interval for Population mean.

Unit V : Time Series analysis

Components of Time series, Additive and Multiplicative Decomposition of Time series components, Measuring trend by method of Moving averages, Straight line and Second degree parabola, Measuring seasonal variation by

Ratio to Trend method and Ratio to Moving averages method.

Suggested Readings:

1. S. C.Gupta&V.K.Kapoor, “Fundamentals of Mathematical Statistics”, S.Chand.
2. Richard A.Johnson,” Probability and Statistics for Engineers”, Pearson Education.
3. Jay Devore, “Probability and Statistics for Engineering and the Sciences”,Cengage learning.
4. Murat Kulahci,“Time series analysis and forecasting by example”,John Wiley & Sons
5. S. C.Gupta&V.K.Kapoor, “Fundamentals of Applied Statistics”, S.Chand.

<table>
<tr><td>Course Code</td><td colspan="6">Course Title</td><td>Core/Elective</td></tr>
<tr><td><b>ES305EC</b></td><td colspan="6"><b>Signals and Systems</b></td><td><b>Core</b></td></tr>
<tr><td rowspan="2">Prerequisite</td><td colspan="4">Contact Hours per Week</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td><td rowspan="2">Credits</td></tr>
<tr><td>L</td><td>T</td><td>D</td><td>P</td></tr>
<tr><td><b>-</b></td><td><b>3</b></td><td><b>-</b></td><td><b>-</b></td><td><b>-</b></td><td><b>30</b></td><td><b>70</b></td><td><b>3</b></td></tr>
</table>

Course Objectives

- 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.
- To understand Sampling theorem, with time and frequency domain analysis of discrete time signals with DTFS, DTFT and Z-Transform.
- 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.

Course 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, 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.

Course Code	Course Title						Core/Elective
PC401CD	**OOP using JAVA**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

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

Course Outcomes

1. Identify classes, objects, members of a class and the relationships needed to solve a problem.
2. Use interfaces and creating user-defined packages.
3. Utilize exception handling and Multithreading concepts to develop Java programs.
4. Compose programs using the Java Collection API.
5. Design a GUI using GUI components with the integration of event handling.
6. Create files and read from computer files.

UNIT-I

Introduction: OOP concepts, history of Java, Java buzzwords, data types, variables, scope and life time of variables, operators, expressions, control statements, type conversion and casting, simple java programs.

Classes and Objects: Concept of classes, objects, constructors, methods, this keyword, super keyword, garbage collection, overloading methods and constructors, parameter passing, Arrays

String handling: String, String Buffer, String Builder

UNIT -II

Inheritance: Base class object, subclass, member access rules, super uses, using final with inheritance, method overriding, abstract classes.

Interfaces: Defining and implementing an interface, differences between classes and interfaces and extending interfaces Polymorphism.

Packages: Defining, creating and accessing a package, importing packages, exploring packages

UNIT -III

Exception handling: Concepts and benefits of exception handling, exception hierarchy, checked and unchecked exceptions, usage of-try, catch, throw, throws and finally, built in exceptions, creating User defined exceptions.

Multithreading: Difference between multi-threading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups.

UNIT -IV

Basic I/O Streams: Java I/O classes and interfaces, Files, Stream and Byte classes, Character streams, Serialization

Exploring java.lang: Object class, Wrapper classes

Exploring java.util: Scanner, StringTokenizer, BitSet, Date, Calendar, Timer

Regular Expressions: Pattern class, Matcher class, Split method. Enum and Internationalization

UNIT -V

AWT & Event Handling: The AWT class hierarchy, user interface components - labels, buttons, canvas, scrollbars, text components, checkbox, checkbox groups, choices, lists.

Events, event sources, event classes, event listeners, delegation event model, handling mouse and key board events, adapter classes.

Layout manager: Border, Grid, Flow, Card and Grid Bag layouts**.**

Swings: Introduction, limitations of AWT, components, containers,

Exploring Swing Components - JApplet, JFrame and JComponent, Icons and Labels, Text fields, JButton class, Checkboxes, Radio buttons, ScrollPanes.

Suggested Readings:

1. Java The complete reference, 8th edition, Herbert Schildt, TMH.
2. Understanding OOP with Java, up dated edition, T. Budd, Pearson education.
3. Head First Java, 2nd Edition by Bert Bates, Kathy Sierra Publisher: O'Reilly Media, Inc.
4. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
5. An Introduction to OOP, second edition, T. Budd, Pearson Education.
6. Introduction to Java programming 6th edition, Y. Daniel Liang, Pearson Education.
7. An introduction to Java programming and object oriented application development, R. A. Johnson-Thomas.

Course Code	Course Title						Core/Elective
PC402CD	**Operating Systems**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**1**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To learn the fundamentals of Operating Systems.
- To learn the mechanisms of OS to handle processes and threads and their communication.
- To learn the mechanisms involved in memory management in contemporary OS.
- To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection.

Course Outcomes

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, Dinning 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.

Course Code	Course Title						Core/Elective
PC403CD	**Database Management Systems**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To Learn mathematical concepts as applied in computer
- To introduce three scheme architecture and DBMS functional components.
- To learn formal and commercial query languages of RDBMS
- To Study different file organization and indexing techniques
- To familiarize theory of serializablity and implementation of concurrency control, and recovery

Course Outcomes

1. Understand the mathematical foundations on which RDBMS are built
2. Model a set of requirements using the Extended Entity Relationship Model (EER), transform an EER model into a relational model and refine the relational model using theory of normalization
3. Develop Database application using SQL and Embedded SQL
4. Use the knowledge of file organization and indexing to improve database application performance
5. Understand the working of concurrency control and recovery mechanisms in RDBMS

UNIT-I

Introduction: Database System Application, Purpose of Database Systems, View of Values, Nested Sub-queries, Complex Queries views, Modification of the Databaae, Joined Relations

Data, Database Language, Relational Databases, Database Design, Object-Based and Semi-Structured Databases, Data Storages and Querying , Transaction Management ,Data Mining and Analysis, Database Architecture, Database Users and Administrators.

Database Design and the E-R Model: Overview of the Design Process, The Entity Relationship Model Constraints, Entity-Relationship Design issues, Weak Entity Sets Extended E-R Features Database Design for banking Enterprise, Reduction to Relational Schemas, Other Aspects of Database Design

UNIT-II

Relational Model: Structure of Relational Databases, Fundamental Relational-Algebra Operations, Additional Relational-Algebra Operations, Extended Relational-Algebra Operations, Null Values, Modification of the Databases

Structured Query Language: Data Definition, Basic Structure of SQL Queries, Set Operations, Aggregate Functions, Null

UNIT-III

Advanced SQL: SQL Data Types and Schemes, Integrity constraints, Authorization, Embedded SQL, Dynamic SQL, Functions and Procedural Constructs, Recursive Queries, Advanced SQL Features.

Relational Database Design: Features of Good Relational Design, Atomic Domains and First Normal Form, Functional Dependency Theory, Decomposition using Functional Dependencies.

UNIT-IV

Indexing and Hashing: Basic Concepts, Ordered Indices, B*-tree index files, B-tree index files, multiple key access, static hashing, dynamic hashing, comparison of ordered indexing and hashing bitmap indices.

Index definition in SQL transactions: Transaction concepts, transaction state, implementation of atomicity and durability, concurrent executions, serializability, recoverability, implementation of isolation, testing for serializability.

UNIT-V

Concurrency Control: Lock based protocols, timestamp based protocols, validation based protocols, multiple granularity, multi version schemes, deadlock handling, insert and delete operations, weak levels of consistency, concurrency of index structures.

Recovery system: Failure classification, storage structure, recovery and atomicity, log-based recovery, recovery with concurrent transactions, buffer management, failure with loss of non-volatile storage, advanced recovery techniques, remote backup systems.

Suggested Readings:

1. Abraham Silberschatz, Henry F Korth, S Sudarshan, Database System Concepts, McGraw-Hill, 6th Edition, 2010
2. Ramakrishnan, Gehrke, Database Management Systems, McGraw-Hill, 3rd Edition, 2003
3. Elmasri, Navathe, Somayajulu, Fundamentals of Database Systems, Pearson Education, 4th Edition, 2004.

1.

Course Code	Course Title						Core/Elective
PC451CD	**Operating Systems Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

The objectives of the course are to impart knowledge of the:

- Learn different types of CPU scheduling algorithms.
- Demonstrate the usage of semaphores for solving synchronization problem.
- Understand memory management techniques and different types of fragmentation.
- Understand Banker‘s algorithm used for deadlock avoidance.

Course Outcomes

After the completion of the course, the student will be able to:

2. Evaluate the performance of different types of CPU scheduling algorithms.
3. Implement producer-consumer problem, reader-writers problem, Dining philosopher’s problem.
4. Simulate Banker’s algorithm for deadlock avoidance.
5. Implement paging replacement and disk scheduling techniques.
6. Use different system calls for writing application programs.

List of Experiments:

1. Write a 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 SynchronizationProblems: 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.

Suggested Readings:

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

Course Code	Course Title						Core/Elective
PC452CD	**OOP using JAVA Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- To build software development skills using java programming for real world applications.
- To implement frontend and backend of an application
- To implement classical problems using java programming.

Course Outcomes

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

1. Design interfaces and packages.
2. Compose program for implementation of multithreading concepts.
3. Develop program using Collection Framework.
4. Develop small GUIs using GUI components with the integration of event handling.
5. Handle I/O Streams from various sources.
6. Write programs using the Java Concepts.

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.
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, Linked List, Tree map and Hash map.
10. Write a Java program to illustrate Legacy classes like Vector, Hashtable, 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

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.

Course Code	Course Title	Core/Elective
PC453CD	**Database Management Systems Lab**	**Core**

Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- To practice various DDL commands in SQL
- To write simple and complex queries in SQL
- To familiarize PL/SQL

Course Outcomes

After the completion of the course, the student will be able to:

1. Design and implement a database schema for a given problem
2. Populate and query a database using SQL and PL/SQL
3. Develop multi-user database application using locks

Creation of database (exercising the commands for creation)

1. Simple to complex condition query creation using SQL Plus.
2. Usage of triggers and stored procedures
3. Creation of forms for student information, library information, pay roll etc.
4. Writing PL/SQL procedures for data validation.
5. Report generation using SQL reports.
6. Creating password and security features for applications.
7. Using of file locking, table locking facilities in applications.
8. Creation of small full-fledged database application spreading over 3 sessions.

Note: The creation of sample database for the purpose of the experiments is expected to be pre-decided by the instructor.