Proposed for the academic years 2020-2024

FACULTY OF ENGINEERING

OSMANIA UNIVERSITY

FOR AFFLIATED COLLEGES

Scheme of Instruction & Examination

(AICTE Model Curriculum)

and

Syllabi of Four Year Degree Program of

Bachelor of Engineering (B.E.) CSE

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

(2020 – 2024)

Proposed for the academic years 2021-2025

SCHEME OF INSTRUCTION
BE (COMPUTER SCIENCE AND ENGINEERING)
(Artificial Intelligence and Data Science)
AICTE MODEL CURRICULUM
I-SEMESTER
AI and DS as Prescribed by OU

S. No	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P	Contact Hrs/Week	CIE	SEE	Duration in Hours	
Theory Courses										
	Induction Program for 3 weeks									
1	MC801PO	Indian Constitution	2	-	-	2	30	70	3	-
2	HS101EG	English	2	-	-	2	30	70	3	2
3	BS202PH	Physics	3	1	-	4	30	70	3	4
4	BS203MT	Mathematics-I	3	1	-	4	30	70	3	4
5	ES301EE	Basic Electrical Engineering	3	1	-	4	30	70	3	4
Practical/Laboratory Courses										
6	HS151EG	English Lab	-	-	2	2	25	50	3	1
7	BS251PH	Physics Lab	-	-	3	3	25	50	3	1.5
8	ES353CE	Engineering Graphics	-	-	3x2	6	50	50	3	3
9	ES354EE	Basic Electrical Engineering Lab	-	-	2	2	25	50	3	1
Total			**13**	3	**13**	**29**	**275**	**550**	-	**20.5**

BS: Basic Sciences **ES: Engineering Sciences** **MC: Mandatory Course**

L:Lectures **T:Tutorials** **P:Practicals** **D:Drawing**

CIE: Continuous Internal Evaluation **SEE: Semester End Examination**

Proposed for the academic years 2020-2024

SCHEME OF INSTRUCTION & EXAMINATION

B.E. II - SEMESTER

(Artificial Intelligence and Data Science)

CSE: II- SEMESTER

AI and DS as Prescribed by OU

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/D	Contact Hours/ Week	CIE	SEE	Duration in Hours	
Theory Courses										
1	MC802CE	Environmental Science	2	-	-	2	30	70	-	
2	MC803PY	Essence of Indian Traditional knowledge	2	-	-	2	30	70	-	
3	BS201MT	Mathematics-II	3	1	-	4	30	70	3	4
4	BS204CH	Chemistry	3	1	-	4	30	70	3	4
5	ES302CS	Programming for Problem Solving	3	1	-	4	30	70	3	4
Practical/Laboratory Courses										
6	BS252CH	Chemistry Lab	-	-	3	3	25	50	3	1.5
7	ES352ME	Workshop Practice	-	-	2x3	6	50	50	3	3
8	ES351 CS	Programming for Problem Solving Lab	-	-	2	2	25	50	3	1
Total			**13**	**03**	**11**	**26**	**250**	**500**		**17.5**

Proposed for the academic years 2021-2025

SCHEME OF INSTRUCTION & EXAMINATION

B.E. III - SEMESTER

(Artificial Intelligence and Data Science)

S.No	Course Code	Course Title	Scheme of Instruction				Scheme of examination			Credits
			L	T	Pr/Drg	Contact Hrs / Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	PC301AD	Data Structures & Algorithms	3			3	30	70	3	3
2	PC302AD	OOPS Using Java	3		-	3	30	70	3	3
3	PC303AD	Discrete Mathematics	2		-	2	30	70	3	2
4	ES216EC	Digital Electronics	3	1		4	30	70	3	4
5	ES214EC	Basic Electronics	3	1	-	4	30	70	3	4
6	BS205MT	Mathematic –III (Probability and Statistics)	3		-	3	30	70	3	3
Practical / Laboratory Courses										
7	PC351AD	Data Structure & Algorithms using C lab	-	-	2	2	25	50	3	1
8	PC352AD	OOPS Using Java lab			2	2	25	50	3	1
9	ES 351EC	Basic Electronics lab	-	-	2	2	25	50	3	1
Total			**17**	**2**	**6**	**25**	**255**	**570**		**22**

BS: Basic Sciences ES: Engineering Sciences MC: Mandatory Course
PC: Professional Course HS: Humanities and Sciences
L: Lectures T: Tutorials Pr: Practicals Drg: Drawing
CIE: Continuous Internal Evaluation **SEE:** Semester End Examination (Univ. Exam)

Note: 1) Each contact hour is a Clock Hour
2) The practical class can be of two and half hour (clock hours) duration as per the requirement of a particular laboratory.
3) Students admitted into B.E./B.Tech. Courses under lateral entry scheme (through ECET) from the academic year 2017-18 should undergo the following bridge course subjects at III Semester (CBCS).
(1) ES 154 CS Computer Programming Lab
(2) MC 156 EG Engineering English Lab

With effect from the academic year2021-25

SCHEME OF INSTRUCTION & EXAMINATION

B.E. IV - SEMESTER

(Artificial Intelligence and Data Science)

S. No	Course Code	Course Title	Scheme of Instruction				Scheme of examination			Credits
			L	T	Pr/Drg	Contact Hrs / Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	PC401AD	Computer Organization & Microprocessor	3	0	-	3	30	70	3	3
2	PC402AD	Design Analysis & Algorithms	2	0	-	2	30	70	3	2
3	PC403AD	Foundation of Data Science	3	0	-	3	30	70	3	3
4	PC404AD	Operating Systems	3	0	-	3	30	70	3	3
5	PC405AD	Computer Networks	3	0	-	3	30	70	3	3
6	HS105CM	Financial and Accounting	3	0	-	3	30	70	3	3
Practical / Laboratory Courses										
7	PC451AD	Computer Organization & Microprocessor lab	-	-	2	2	25	50	3	1
9	PC452AD	Computer Networks and Operating Systems Lab	-	-	2	2	25	50	3	1
10	PC454AD	Data Science lab	-	-	2	2	25	50	3	1
Total			**17**	**0**	**08**	**25**	**280**	**620**		**20**

BS: Basic Sciences ES: Engineering Sciences MC: Mandatory Course
PC: Professional Course HS: Humanities and Sciences
L: Lectures T: Tutorials Pr: Practicals Drg: Drawing
CIE: Continuous Internal Evaluation **SEE:** Semester End Examination (Univ. Exam)

Note: 1) Each contact hour is a Clock Hour
2) The practical class can be of two and half hour (clock hours) duration as per the requirement of a particular laboratory.

Proposed for the academic years 2021-2025

SCHEME OF INSTRUCTION & EXAMINATION

B.E. V - SEMESTER

(Artificial Intelligence and Data Science)

S.No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	D/P	Contact Hrs/W	CIE	SEE	Duration	
Theory Course										
1.	PC501AD	Machine Learning and Techniques	3	0	-	3	30	70	3	3
2.	PC502AD	Database Management System	3	0	-	**3**	30	70	3	3
3.	PC503AD	Automata languages & Computation	3	0	-	3	30	70	3	3
4.	PC504AD	Artificial Intelligence	3	-	-	3	30	70	3	3
5.	PE-I	Professional Elective-I	3	-	-	3	30	70	3	3
6.	PE-II	Professional Elective-II	3	-	-	3	30	70	3	3
Practical/Laboratory Course										
7.	PC551AD	Machine Learning lab	-	-	2	2	25	50	3	1
8.	PC552AD	DBMS Lab			2	2	25	50	3	1
9.	PW553AD	Mini Project	-	-	4	4	25	50	3	2
Total			**20**	**00**	**08**	**32**	**280**	**640**		**22**

Professional Elective–I	
Course Code	**Course Title**
PE511AD	Mathematical Model for Data Science
PE512AD	Machine Vision
PE513AD	Web Technology
PE514AD	Foundation of Cryptography

Professional Elective–II	
Course Code	**Course Title**
PE521AD	Speech Processing
PE522AD	Artificial Neural Networks
PE523AD	Internet of things
PE524AD	Digital Forensics

With effect from the academic year2021-25

SCHEME OF INSTRUCTION & EXAMINATION

B.E. VI - SEMESTER

(Artificial Intelligence and Data Science)

S.No	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	D/P	Contact Hrs/Wk	CIE	SEE	Duration In Hrs/Wk	
Theory Courses										
1.	PC601AD	Software Engineering	3	0	-	3	30	70	3	3
2.	PC602AD	Advanced Data Science	3	0	-	3	30	70	3	3
3.	PE-III	Professional Elective-III	3	0	-	3	30	70	3	3
4.	PE–IV	Professional Elective-IV	3	-	-	3	30	70	3	3
5	OE-I	Open Elective-I	3	-	-	3	30	70	3	3
6	HS	Effective Technical Communication	3			3	30	70	3	3
Practical/Laboratory Courses										
7	PC654AD	Software Engineering lab	-	-	2	2	25	50	3	1
8	PC655AD	ADS Lab	-	-	2	2	25	50	3	1
9	SI671AD	Summer Internship*	-	-	-	-	25	25	-	2
Total			**15**	**0**	**6**	**21**	**280**	**620**		**22**

Profession Elective–III	
Course Code	**Course Title**
PE631AD	Big Data Analytics
PE632AD	Semantic Web & Social Networks
PE633AD	Block Chain Technology
PE634AD	Design Thinking

Profession Elective–IV	
Course Code	**Course Title**
PE641AD	Compiler Design
PE642AD	Distributed Systems
PE643AD	Web Services
PE644AD	Soft Computing

Open Elective-I		
Sl.No	**Code**	**Name of Subject**
1	OE601 EE	Electrical Energy Conservation and Safety (Not for EEE & EIE Students)
2	OE601 EG	Soft Skills & Interpersonal Skills
3	OE602 MB	Human Resource Development and Organizational Behaviour
4	OE601 LW	Cyber Law and Ethics
5	OE601 CE	Disaster Mitigation (Not for Civil Engg. Students)
6	Code from OU	Foundation of Data Science(Not for AI & DS ,CSE students)
7	Code from OU	Introduction to AI(Not for AI & DS,CSE,IT students)

SCHEME OF INSTRUCTION & EXAMINATION
B.E. VII – SEMESTER
(Artificial Intelligence and Data Science)

S.No	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	D/P	Contact Hrs/Wk	CIE	SEE	Duration In Hrs/Wk	
Theory Course										
1.	PC701AD	Deep Learning	3		-	3	30	70	3	3
2.	PC702AD	Artificial Intelligence and Robotics	3		-	3	30	70	3	3
3.	PE–IV	Professional Elective–V	3	-	-	3	30	70	3	3
4	PE–V	Professional Elective–VI	3	-	-	3	30	70	3	3
5	OE-II	Open Elective II	3	-	-	3	30	70	3	3
Practical/Laboratory Course										
6	PC751AD	AI And Robotics Lab	-	-	2	2	25	50	3	1
7	PC752AD	Deep Learning Lab	-	-	2	2	25	50	3	1
8	PW751AD	Project Seminar (Phase-I)	-	-	6	6	50			3
Total			**15**		**10**	**25**	**250**	**450**		**20**

Profession Elective–V	
Course Code	**Course Title**
PE751AD	Information Retrieval System
PE752AD	Cloud Computing
PE753AD	Mining of Massive Datasets
PE754AD	Business Intelligence

Profession Elective–VI	
Course Code	**Course Title**
PE761AD	Human Computer Interaction
PE762AD	Software Quality Assurance and Testing
PE763AD	Cybersecurity
PE764AD	Natural Language Processing

Open Elective – II		
1	OE603 EE	Non-Conventional Energy Sources (Not for EEE & EIE Students)
2	OE621 ME	Industrial Robotics (Not for Mech Engg& Prod. Engg. students)
3	OE602 CE	Green Building Technologies (Not for Civil Engg. Students)
4	OE 603 IT	Cyber Security (Not for IT Students)
5	AIDS code	Data Handling & Data Visualization(Not for CSE,AI & DS Students)
6	OE775ME	Entrepreneurship

SCHEME OF INSTRUCTION & EXAMINATION

With effect from the academic year2021-25

B.E. VIII – SEMESTER

(Artificial Intelligence and Data Science)

S.No	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	D/P	Contact Hrs/Wk	CIE	SEE	Duration In Hrs/Wk	
Theory Course										
1.	OE-III	Open Elective III	3		-	3	30	70	3	3
2	PE	Professional Elective–VII	3		-	3	30	70	3	3
Practical/Laboratory Course										
3	PW861CS	Technical Seminar			4	4	50			2
4	PW862AD	Project Work-II			16	16	50	100		8
Total			**6**		**20**	**20**	**160**	**240**		**16**

Proposed for the academic years 2021-2025

Profession Elective–VII	
Course Code	**Course Title**
PE871AD	Social Medial Analytics
PE872AD	Edge Computing
PE873AD	Knowledge Management
PE874AD	Fundamental of Augmented Reality and Virtual Reality

Open Elective – III		
1	OE605 EE	Smart Building Systems (Not for EEE & EIE Students)
2	OE631 AE	Automotive Safety and Ergonomics (Not for Auto. Engg students)
3	OE603 CE	Road Safety Engineering (Not for Civil Engg. Students)
4	OE604 IT	Software Engineering (Not for IT Students)
5	AI DS code	Machine Learning & Techniques (Not for IT & CSE ,CSE stream Students)

Program Electives For V, VI , VII and VIII SEM

Elective	**Sem V**		**Sem VI**		**Sem VII**		**Sem VIII**
	PE-I	**PE-II**	**PE-III**	**PE-IV**	**PE-V**	**PE-VI**	**PE-VII**
Data Science	Mathematical Modeling for Data Science		Big Data Analytics		Business Intelligence		Social Media Analytics
AI	Machine Vision	ANN			Mining of Massive Datasets		
		Speech Processing		Compiler Design	Information Retrieval Systems	Natural Language Processing	
Cloud				Distributed system	Cloud Computing		Edge Computing
Web Programming	Web Technologies		Semantic Web & Social Networks	Web Services			
Networks	Foundation of Cryptography	Digital Forensics	Blockchain Technologies			Cyber Security	
General		Internet of Things	Design Thinking	Soft Computing		Human Computer Interaction	Fundamentals of AR and VR
SE						Software Quality Testing	Knowledge Management

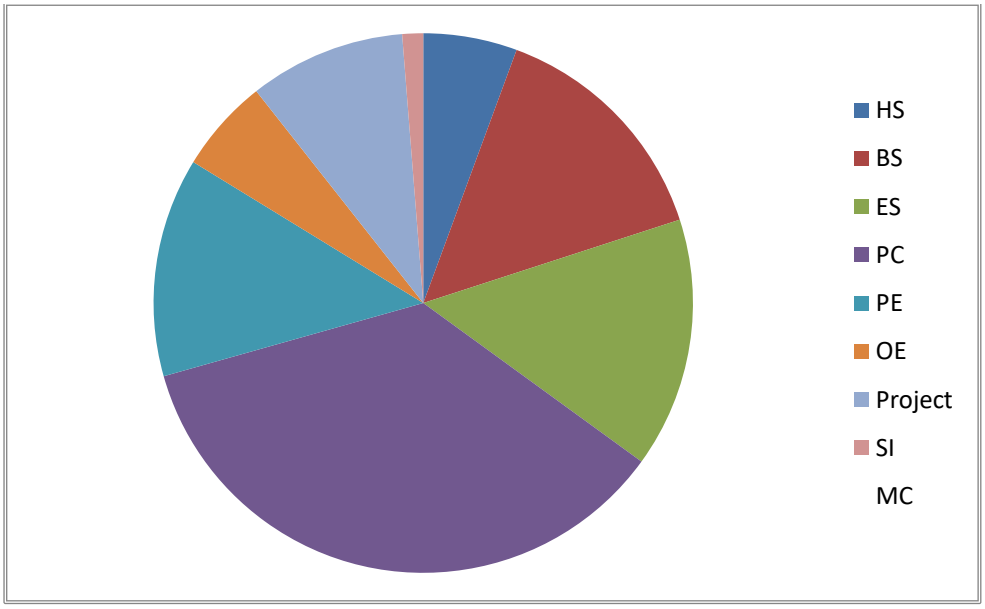
Credit Contribution Group-wise

Semester	CIE	SEE	Credits
1	275	550	20.5
2	250	500	17.5
3	255	575	22
4	280	620	20
5	280	640	24
6	280	620	20
7	250	450	20
8	160	240	16
Total	2030	4195	160
	6225		

Category	Sem1	Sem2	Sem3	Sem4	Sem-5	Sem-6	Sem-7	Sem-8	Total Credits group wise
HS	1+1(3)			1(3)	1(3)				9
BS	2+1(9.5)	2+1(10.5)	1(3)						23
ES	1+2(8)	1+2(7)	2+1(9)						24
PC			3+2(10)	5+3(17)	4+2(13)	3+3(9)	2+2(8)		57
PE					2(6)	2(6)	2(6)	1(3)	21
OE						1(3)	1(3)	1(3)	9
Project					1(2)		1(3)	2(10)	15
SI						1(2)			2
MC	1	2							
Totalsemwise	9(20.5)	10(17.5)	9(22)	10(20)	9(24)	9(20)	8(20)	3(14)	160

	MJCET	AICTE
Humanities & Social Science including Management Courses	9	12
Basic Sciences Courses	22.5	26
Engineering Sciences Courses including workshop, drawing, basics of electrical/ mechanical / computer etc	24	29
Professional Core Courses	57	47
Professional Elective Courses relevant to Chosen specialization/branch	21	23
Open Subjects- Electives form Courses technical and /or emerging subjects	9	11
Project work, seminar and internship in industry or elsewhere	13	12
Mandatory courses(Environmental Sciences, induction program, Indian constitution , Essence of Indian Knowledge Tradition)		
Total	160	160

Number of Credits with Respective to Category Pie Chart



Course Code	Course Title						Core/Elective
BS207MT	**Mathematics – III (Probability & Statistics)**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To introduce the solution methodologies for second order Partial Differential Equations with applications in engineering
- To provide an overview of probability and statistics to engineers

Course Outcomes

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

1. Solve field problems in engineering involving PDEs.
2. They can also formulate and solve problems involving random variables and apply statistical methods for analyzing experimental data.

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

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

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

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

UNIT-V: Test for single mean, difference of means and correlation coefficients, test for ratio of variances, Chi-square test for goodness of fit and independence of attributes, -control charts for measurements (X and R charts) – Control charts for attributes (p, c and np charts) – Tolerance limits – Acceptance sampling

Suggested Readings:

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

8. Ross, S.M., “Introduction to Probability and Statistics”, Academic Foundation, 2011.
9. Papoulis, A. and Pillai, S.U, “Probability, Random Variables and Stochastic Processes”, TMH, 2010

1. Verilog HDL A Guide to Digital Design and Synthesis,” 2nd Edition, Pearson Education, 2006.

Course Code	Course Title						Core/Elective
PC221CS	**Data Structures and Algorithm**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To teach the importance of structuring the data for easy access and storage.
- To teach the implementation of various data structures.
- To acquire skills in using generic principles for data representation and manipulation with a view for efficiency, maintainability and code reuse.
- To introduce the basic concepts of advanced data structures.

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. Develop applications using Linear and Non-linear data structures.
5. Apply the suitable data structure for a real world problem and think critically for improvement in solutions.
6. Determine the suitability of the standard algorithms: Searching, Sorting and Traversals.

UNIT-I

Introduction Algorithms: Introduction, Algorithm Specifications, Recursive Algorithms, Performance Analysis of an algorithm- Time and Space Complexity, Asymptotic Notations, Amortized analysis

UNIT-II

Stacks and Queues: ADT Stack and its operations: Algorithms and their complexity analysis, **Applications of Stacks:** Expression Conversion and evaluation –corresponding algorithms and complexity analysis, Queue ADT and its operations: Linear Queue, Circular Queue, Algorithms and their analysis.

UNIT-III

Linked Lists: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, **Doubly linked list:** Operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.

UNIT-IV

Trees: Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis, Heaps.

UNIT-V

Sorting and Searching: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort,

Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Linear and Binary Search algorithms, and their complexity analysis, Hashing

Graph: Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.

Suggested Readings:

1. "Fundamentals of Data Structures in C++", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Dinesh Mehta, 2nd Edition, Universities Press.
2. Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, 3rd Edition, Pearson India.
3. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.
4. "How to Solve it by Computer", 2nd Impression by R.G. Dromey, Pearson Education.

Course Code	Course Title						Core/Elective
PC222CS	**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
PC231CS	**OOPS 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, StringBuffer, StringBuilder

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
ES214EC	**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
ES216EC	**Digital Electronics**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

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

Course Outcomes

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

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

UNIT – I

Design Concepts: Digital Hardware, Design process, Design of digital hardware. Introduction to logic circuits – Variables and functions, Logic gates and networks. Boolean algebra, Synthesis using gates, Design examples. Optimized implementation of logic functions using K-Map and Quine-McCluskey Tabular method

UNIT – II

Number representation: Addition and Subtraction of signed and unsigned numbers.

Combinational circuit building blocks: Half adder, Full adder, Multiplexers. Decoders. Encoders. Code converters, BCD to 7-segment converter, Arithmetic comparator circuits.

UNIT – III

Design of combinational circuits using Programmable Logic Devices (PLDs): General structure of a Programmable Array Logic (PAL), Programmable Logic Arrays (PLAs), Structure of CPLDs and FPGAs, 2-input and 3-input lookup tables (LUTs)

Introduction to Verilog HDL: Verilog code for basic logic gates, adders, decoders.

UNIT – IV

Sequential Circuits: Basic Latch, Gated SR Latch, gated D Latch, Master-Slave edge triggered flip-flops, T Flip-flop, JK Flip-flop, Excitation tables. Registers, Counters, Verilog code for flip-flops

UNIT – V

Synchronous Sequential Circuits: Basic Design Steps, Finite State machine (FSM) representation using Moore and Mealy state models, State minimization, Design of FSM for Sequence Generation and Detection, Algorithmic State Machine charts.

Suggested Readings:

2. Moris Mano and Michael D CIletti, Digital Design, Pearson, fourth edition, 2008
3. Zvi Kohavi, Switching and Finite Automata Theory, 3rd ed., Cambridge University Press-New Delhi,

2011.
4. R. P Jain, Modern Digital Electronics,4th ed., McGraw Hill Education (India) Private Limited, 2003
5. Ronald J.Tocci, Neal S. Widmer &Gregory L.Moss, "Digital Systems: Principles and Applications," PHI, 10/e, 2009.

Samir Palnitkar, "

Course Code	Course Title						Core/Elective
PC251EC	**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.

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

Course Code	Course Title	Core/Elective
PC252CS	**Data Structures and Algorithm Lab**	**Core**

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

Course Objectives

- Design and construct simple programs by using the concepts of structures as abstract data type.
- To have a broad idea about how to use pointers in the implement of data structures.
- To enhance programming skills while improving their practical knowledge in data structures.
- To strengthen the practical ability to apply suitable data structure for real time applications.

Course Outcomes

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

1. Implement the abstract data type and reusability of a particular data structure.
2. Implement linear data structures such as stacks, queues using array and linked list.
3. Understand and implements non-linear data structures such as trees, graphs.
4. Implement various kinds of searching, sorting and traversal techniques and know when to choose which technique.
5. Understanding and implementing hashing techniques.
6. Decide a suitable data structure and algorithm to solve a real world problem.

Programming Exercise using C:

1. Implementation of Stacks, Queues (using both arrays and linked lists).
2. Implementation of Singly Linked List, Doubly Linked List and Circular List.
3. Implementation of Infix to Postfix conversion and evaluation of postfix expression.
4. Implementation of Polynomial arithmetic using linked list.
5. Implementation of Linear search and Binary Search
6. Implementation of Hashing Technique
7. Implementation of Binary Tree and Binary tree traversal techniques (inorder, preorder, postorder, level-order)
8. Implementation of Binary search tree and its operations
9. Implementation of Insertion Sort, Selection Sort, Bubble Sort, Merge Sort, Quick Sort, Heap Sort
10. Implementation of operations on AVL trees.
11. Implementation of Graph Search Methods.

Note: It is recommended to use a debugging tool to debug the programs.

Course Code	Course Title						Core/Elective
PC262CS	**OOPs 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
PC401AD	**Computer Organization & Microprocessor**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To understand basic components of computers
- To explore the I/O organizations in depth.
- To explore the memory organization.
- To understand the basic chip design and organization of 8086 with assembly language.

Course Outcomes

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

1. After this course students understand in a better way the I/O and memory organization in depth.
2. Ability to understand the merits and pitfalls in computer performance measurements.
3. Identify the basic elements and functions of 8086 microprocessors.
4. Understand the instruction set of 8086 and use them to write assembly language programs.
5. Demonstrate fundamental understanding on the operation between the microprocessor and its interfacing devices.

UNIT-I

Basic Computer Organization: Functions of CPU, I/O Units, Memory: Instruction: Instruction FormatsOne address, two addresses, zero addresses and three addresses and comparison; addressing modes with numeric examples: Program Control- Status bit conditions, conditional branch instructions, Program Interrupts: Types of Interrupts.

UNIT-II

Input-Output Organizations: I/O Interface, I/O Bus and Interface modules: I/O Vs Memory Bus, Isolated Vs Memory-Mapped I/O, Asynchronous data Transfer- Strobe Control, Hand Shaking: Asynchronous Serial transfer- Asynchronous Communication interface, Modes of transfer Programmed I/O, Interrupt Initiated I/O, DMA; DMA Controller, DMA Transfer, IOP-CPU-IOP Communication, Intel 8089 IOP.

UNIT-III

Memory Organizations: Memory hierarchy, Main Memory, RAM, ROM Chips, Memory Address Map, Memory Connection to CPU, associate memory, Cache Memory, Data Cache, Instruction cache, Miss and Hit ratio, Access time, associative, set associative, mapping, waiting into cache, Introduction to virtual memory.

UNIT-IV

8086 CPU Pin Diagram: Special functions of general purpose registers, Segment register, concept of pipelining, 8086 Flag register, Addressing modes of 8086.

8086-Instruction formats: assembly Language Programs involving branch & Call instructions, sorting, evaluation of arithmetic expressions. Interfacing with peripherals.

UNIT-V

Interfacing: 8255.8253,8257, 8259, RS-232, 555 Timer

Suggested Readings:

References:

1. E. Horowitz, S. Sahni, Fundamentals of Computer Algorithms.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, “Data Structures and Algorithms”, Pearson Education, Reprint 2006.
3. M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley, 2002.
4. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, Third Edition, PHI Learning Private Limited, 2012.

Course Code	Course Title						Core/Elective
PC402AD	**Design and Analysis of Algorithms**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

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

Course Outcomes

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

7. Ability to analyze the performance of algorithms.
8. Ability to choose appropriate algorithm design techniques for solving problems.
9. Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs

UNIT-I

Introduction: Algorithm definition, and specification, asymptotic analysis – best, average, and worst-case behavior; Performance measurements of Algorithms, Time and Space complexities, Analysis of recursive algorithms.

Basic Data Structures: Disjoint set operations, union and find algorithms, Dictionaries, Graphs, Trees.

UNIT-II

Divide and Conquer: General method, Control abstraction, Merge sort, Quick Sort – Worst, Best and average case. Binary search.

Brute Force: Computing an– String Matching – Closest-Pair and Convex-Hull Problems - Exhaustive Search – Travelling Salesman Problem – Knapsack Problem – Assignment problem.

Greedy method: General method, applications- Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Single source shortest path problem.

UNIT-III

Dynamic Programming: General Method, applications- All pairs shortest path problem, Optimal binary search trees, 0/1 knapsack problem, Reliability design, Traveling sales person problem.

Backtracking: General method, Recursive backtracking algorithm, Iterative backtracking method. 8-Queen problem, Hamiltonian Cycle, 0/1 Knapsack Problem. With effect from the academic year 2020-21

Branch and Bound: Control abstractions for Least Cost Search, Bounding, FIFO branch and bound, LC branch and bound, 0/1 Knapsack problem – LC branch and bound and FIFO branch and bound solution, Traveling sales person problem.

UNIT-IV

Graph Algorithms: Graph Traversals DFS, BFS, Transitive Closure, Directed Acyclic Graphs - Topological Ordering, Network Flow algorithms.

Tries: Standard Tries, Compressed Tries, Suffix Tries, Search Engine Indexing.

External Searching and B-Trees: (a, b) Trees, B-Trees

UNIT-V

Computational Complexity:

Non Deterministic algorithms, The classes: P, NP, NP Complete, NP Hard, Satisfiability problem, Proofs for **NP Complete Problems**: Clique, Vertex Cover.

Parallel Algorithms: Introduction, models for parallel computing, computing with complete binary tree.

Course Code	Course Title						Core / Elective
PC 702 CS / OE 872 CS	**Data Science Using R Programming**						**Core / Open Elective-II**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To learn basics of R Programming environment: R language, R- studio and R packages
- To learn various statistical concepts like linear and logistic regression, cluster analysis, time series forecasting
- To learn Decision tree induction, association rule mining and text mining

Course Outcomes:

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

1. Use various data structures and packages in R for data visualization and summarization
2. Use linear, non-linear regression models, and classification techniques for data analysis
3. Use clustering methods including K-means and CURE algorithm

UNIT – I

Data Science: Introduction to data science, Linear Algebra for data science, Linear equations, Distance, Hyper planes, Half spaces, Eigen values, Eigenvectors.

UNIT II

Statistical Modelling, Random variables, Probability mass/density functions, sample statistics, hypothesis testing.

UNIT III

Predictive Modelling: Linear Regression, Simple Linear Regression model building, Multiple Linear Regression, Logistic regression

UNIT IV

Introduction to R Programming, getting started with R: Installation of R software and using the interface, Variables and data types, R Objects, Vectors and lists, Operations: Arithmetic, Logical and Matrix operations, Data frames, functions, Control structures, Debugging and Simulation in R.

UNIT V

Classification: performance measures, Logistic regression, K-Nearest neighbors (KNN), Clustering: K-Means Algorithm. Case Study

Suggested Readings:

1. Nina Zumel, Practical Data Science with R, Manning Publications, 2014.
2. Peter Bruce and Andrew Bruce, Practical Statistics for Data Scientists, O'Reilly, 2017.
3. Hadley Wickham and Garrett Grolemund, R for Data Science, O'Reilly, 2017.
4. Roger D Peng, R Programming for Data science, Lean Publishing, 2016.
5. Rafael A Irizarry, Introduction to Data Science, Lean Publishing,2016.
6. Vishwa Vishwanathan and Shanthi Vishwanathan, R Data Analysis cookbook 2015

Course Code	Course Title						Core/ Elective
PC 404 AD	**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
- To know the components and management aspects of concurrency management

Course Outcomes

- Identify System calls and evaluate process scheduling criteria of OS.
- Develop procedures for process synchronization of an OS.
- Demonstrate the concepts of memory management and of disk management
- Solve issues related to file system interface and implementation, I/O systems
- 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, Microkemel 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: Principleof 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 Replacementalgorithms, 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 books:

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

Course Code	Course Title						Core/ Elective
PC405AD	**COMPUTER NETWORKS**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	p			
	3	1			30	70	

Course 0/yectives

- To develop an understanding of communication in modern network architectures from a design and performance perspective.
- To understand Data Transmission standards and MAC protocols.
- To introduce the protocols functionalities in Network Layer and Transport Layer.
- To understand DNS and supportive application protocols.
- To provide basic concepts of Cryptography.

Course Outcomes

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

- Explain the functions of the different layer of the OSI and TCP/IP Protocol.
- Understand wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) describe the function of each block.
- Illustrate network layer and transport layer protocols. For a given problem related TCP/IP protocol developed the network programming.
- Configure DNS , EMAIL, SNMP, Bluetooth, Firewalls using open source available software and tools.
- Identify the types of encryption techniques.

UNIT-I

Data communication Components: Representation of data communication, flow of Networks,Layered architecture, OSI and TCP/IP model, Transmission Media. (William stalling)

Techniques for Bandwidth utilization: Line configuration, Multiplexing - Frequency division, Time division and Wave division, Asynchronous and Synchronous transmission,XDSL , Introduction to Wired and Wireless LAN

UNIT-II

Data Link Layer and Medium Access Sub Layer: Error Detection and Error Correction -

Fundamentals, Block coding, Hamming Distance, CRC;

Flow Control and Error control protocols: Stop and Wait, Go back - N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking.

Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA/CD,CDMA/CA

UNIT-III

Network Layer: Switching techniques (Circuit and Packet) concept **,Logical addressing:** IPV4(Header), IPV6(Header), NAT, Sub-Netting concepts.

Inter-Networking:Tunnelling, Fragmentation, congestion control (Leaky Bucket and Token Bucket algorithm), Internet control protocols: ARP, RARP, BOOTP and DHCP.

Network Routing Algorithms: Delivery, Forwarding and Unicast Routing protocol,Gateway protocols.

UNIT-IV

Transport Layer: Process to Process Communication, Elements of transport protocol ,

Internet Transport Protocols: UDP, TCP.

Congestion and Quality of Service, QoS improving techniques.

UNIT-V

Application Layer: Domain Name Space (DNS), EMAIL, SNMP, Bluetooth.

Basic concepts of Cryptography: Network Security Attacks, firewalls, symmetric encryption, Data encryption Standards, public key Encryption (RSA), Hash function,Message authentication, Digital Signature.

Suggested books:

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan,McGrawHill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.
3. W. Richard Stevens, Unix Network Programming, Prentice Hall/ Pearson Education, 2009

Course Code	Course Title						Core/Elective
HS105CM	**Finance and Accounting**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

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 analyze 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. Analyze 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 annuitiesFinancial 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
PC 452 AD	**COMPUTER NETWORKS & OPERATING SYSTEMS LAB**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	p			
DC				2	30	70	

Course 0/yectives

- Learn to communicate between two desktop computers.
- Learn to implement the different protocols
- Be familiar with socket programming.
- Be familiar with the various routing algorithms
- Be familiar with simulation tools.
- To use simulation tools to analyze the performance of various network protocols
- 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 that occur in them and various page replacement policies
- Understand Banker's algorithm used for deadlock avoidance
- Learn various disk scheduling algorithms.

Course Outcomes

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

- Implement various protocols using TCP and UDP.
- Program using sockets.
- Use simulation tools to analyze the performance of various network protocols.
- Implement and Analyze various routing algorithms.
- Evaluate the performance of different types of CPU scheduling algorithms
- Implement producer-consumer problem, reader-writers problem, Dining philosopher's problem
- Simulate Banker's algorithm for deadlock avoidance
- Implement paging replacement and disk scheduling techniquesUse different system calls for writing application programs.

Computer Networks Lab

Part – A

1. Configuration of router, hub, switch etc. (using real devices or simulators)
2. Running and using services/commands like ping, traceroute, nslookup, arp, telnet, ftp, etc.
3. Network packet analysis using tools like Wireshark, tcpdump, etc.
4. Network simulation using tools like Cisco Packet Tracer, NetSim, OMNeT++, NS2, NS3, etc.
5. Socket programming using UDP and TCP (e.g., simple DNS, data & time client/server, echo client/server, iterative & concurrent servers)
6. Programming using raw sockets
7. Programming using RPC

Part -B

Operating Systems Lab:

1. Write C programs to Simulate the following CPU scheduling algorithms
   a) FCFS b) SJF c)Round Robin d) Priority
2. Write C programs to Simulate IPC techniques
   a) Pipes b) Message Queues c) Shared Memory
3. Write C Programs to Simulate Classical Problems of Synchronization
   a) Readers-Writers b) Producers-Consumers C) Dining Philosophers
4. Write C Program to simulate Bankers Algorithm for Dead Lock Avoidance
5. Write C Programs to Simulate all page replacement algorithms
   a) FIFO b) LRU c) Optimal Etc...
6. Write C program to Simulate Disk Scheduling Algorithms
   a) FCFS b) SSTF etct.
7. Write Unix Shell Programs