

ACADEMIC CURRICULA

BACHELOR'S DEGREE PROGRAMME

B.Tech. in Information Technology (IT)

Course Structure and Detailed Syllabi
for students admitted in
2025–26
Academic Session



HALDIA INSTITUTE OF TECHNOLOGY
Haldia, West Bengal, India

Semester III (Second year)							
Sl. No.	Type of course	Code	Course Title	Hours per week			Credits
				L	T	P	
Theory							
1	Engineering Science Course	ESC-IT-301	Digital Electronics	3	0	0	3
2	Professional Core Courses	PCC-IT-301	Data Structure & Algorithms	3	0	0	3
3	Professional Core Courses	PCC-IT-302	Object Oriented Programming	3	0	0	3
4	Professional Core Courses	PCC-IT-303	Database Management System	3	0	0	3
5	Basic Science course	BSC-IT-301	Mathematics-III (Probability & Statistics)	3	0	0	3
6	Humanities & Social Sciences including Management courses	HSMC-IT-301	Economics for Engineers (Humanities-II)	3	0	0	3
7	Mandatory Courses	MC-IT-301	Environmental Sciences	1	-	-	0
Practical							
8	Engineering Science Course	ESC-IT-391	IT Workshop I (SciLab/MATLAB/Python/R)	0	0	4	2
9	Professional Core Courses	PCC-IT-391	Data Structure & Algorithms Lab	0	0	4	2
10	Professional Core Courses	PCC-IT-392	Object Oriented Programming Lab	0	0	4	2
11	Professional Core Courses	PCC-IT-393	Database Management System Lab	0	0	4	2
			Total Credit				26

Semester IV (Second year)							
Sl. No.	Type of course	Code	Course Title	Hours per week			Credits
				L	T	P	
Theory							
1	Basic Science course	BSC-IT-401	Discrete Mathematics	3	0	0	3
2	Professional Core Courses	PCC-IT-402	Computer Organization & Architecture	3	0	0	3
3	Professional Core Courses	PCC- IT-403	Formal Language & Automata Theory	3	0	0	3
4	Professional Core Courses	PCC- IT-404	Design & Analysis of Algorithms	3	0	0	3
5	Professional Core Courses	PCC- IT-405	Operating System	3	0	0	3
6	Professional Core Courses	PCC- IT-406	Artificial Intelligence	3	0	0	3
Practical							
7	Professional Core Courses	PCC-IT-492	Computer Organization & Architecture Lab	0	0	4	2
8	Professional Core Courses	PCC-IT-494	Algorithms Lab	0	0	4	2

9	Professional Core Courses	PCC-IT-495	Operating System Lab	0	0	4	2
10	Professional Core Courses	PCC-IT-496	Artificial Intelligence Lab	0	0	4	2
			Total Credit				26

Semester V (Third year)							
Sl. No.	Type of course	Code	Course Title	Hours per week			Credits
				L	T	P	
Theory							
1	Professional Core Courses	PCC- IT-501	Machine Learning	3	0	0	3
2	Professional Core Courses	PCC- IT-502	Computer Networks	3	0	0	3
3	Professional Elective Courses	PEC-IT-501 (A/B/C)	(Elective-I) Image Processing/ Pattern Recognition / Microprocessor & Microcontroller	3	0	0	3
4	Humanities &Social Sciences including Management courses	HSMC-IT-501	Introduction to Industrial Management (Humanities III)	3	0	0	3
5	Professional Elective courses	PEC-IT-502 (A/B/C)	(Elective-II) Compiler Design/ Micro-electronics and VLSI Design / Bioinformatics	3	0	0	3
6	Mandatory Courses	MC- IT-501	Constitution of India/ Essence of Indian Knowledge Tradition	1	-	-	0
Practical							
7	Professional Core Courses	PCC-IT-591	Machine Learning Lab	0	0	4	2
8	Professional Core Courses	PCC- IT-592	Computer Networks Lab	0	0	4	2
9	Professional Core Courses	PEC- IT-593 (A/B)	Web Technology Lab/ Mobile App Development	0	0	4	2
			Total Credit				21

Semester VI (Third year)							
Sl. No.	Type of course	Code	Course Title	Hours per week			Credits
				L	T	P	
Theory							
1	Professional Core Courses	PCC- IT-601	Deep Learning	3	0	0	3
2	Professional Core Courses	PCC- IT-602	Cloud and Edge Computing	3	0	0	3
3	Professional Core Courses	PCC-IT-603	Internet of Things	3	0	0	3
4	Professional Elective courses	PEC- IT-601 (A/B/C)	Cryptography and Network Security/Ad-hoc Sensor Network/ wireless Sensor Network	3	0	0	3

5	Professional Elective courses	PEC- IT-602 (A/B/C/D)	(Elective-II) Blockchain Technology/ Computer Vision / Natural Language Processing /Embedded System	3	0	0	3
6	Open Elective courses	OEC- IT-601 (A/B/C)	(Open Elective-I) Operations Research /Introduction to Philosophical Thoughts/Soft Skill & Interpersonal Communication	3	0	0	3
Practical							
7	Professional Core Courses	PCC-IT-691	Deep Learning Lab	0	0	4	2
8	Professional Core Courses	PCC-IT-692	Cloud and Edge Computing Lab	0	0	4	2
9	Professional Core Courses	PCC-IT-693	Internet of Things Lab	0	0	4	2
			Total Credit				24

Semester VII (Fourth year)

Semester VII (Fourth Year)							
Sl. No.	Type of course	Code	Course Title	Hours per week			Credits
				L	T	P	
Theory							
1	Professional Elective Courses	PEC- IT-701 (A/B/C/D)	(Elective-III) Big Data Analytics/ Cyber Physical System / Generative AI/ Human Computer Interaction	3	0	0	3
2	Open Elective courses	OEC- IT-701 (A/B)	Project Management and Entrepreneurship/ Organizational Behavior	3	0	0	3
3	Professional Sessional Courses	PSC-IT-781	Industrial Training-I/ Internship-I	0	0	0	3
4	Project	PROJ-IT-781	Project-I	0	0	12	6
			Total Credit				15

Semester VIII (Fourth year)

Semester VIII (Fourth year)							
Sl. No.	Type of course	Code	Course Title	Hours per week			Credits
				L	T	P	
Theory							
1	Professional Elective courses	PEC- IT-801 (A/B /C/D)	(Elective-IV) Augmented & Virtual Reality/Cyber Security and Ethical Hacking / Computational Geometry/ Social Network Analysis	3	0	0	3
2	Professional Sessional Courses	PSC-IT-881	Industrial Training-II/ Internship-II	0	0	0	3
3	Project	PROJ-IT-881	Project-II	0	0	12	6
			Total Credit				12

Year wise Credit Point Distribution:

YEAR	CREDIT POINT
1st YEAR	40
2nd YEAR	52
3rd YEAR	45
4th YEAR	27
TOTAL CREDIT POINT	164

Third Semester (2nd Year)

Digital Electronics

Code: ESC-IT-301

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To acquire the basic knowledge of different analog components and their applications
2.	To acquire the basic knowledge of digital logic levels and application of knowledge to understand digital electronics circuits.
3.	To prepare students to perform the analysis and design of various digital electronic circuits

Pre-Requisite:	
1.	Basic Electronics Parts I & II learned in the First year, semesters 1 & 2.
2.	Basic concept of the working of P-N diodes, Schottky diodes,
3.	Basic FETs and OPAMP as a basic circuit component. Concept of Feedback

Unit	Content	Hrs.
1.	Different Classes of Amplifiers - (Class-A, B, AB and C - basic concepts, power, efficiency; Recapitulation of basic concepts of Feedback and Oscillation, Phase Shift, Wein Bridge oscillators Astable & Monostable Multi-vibrators; Schmitt Trigger circuits, 555 Timer.	9
2.	Binary Number System & Boolean Algebra (recapitulation); BCD, ASCII, EBDIC, Gray codes and their conversions; Signed binary number representation with 1's and 2's complement methods, Binary arithmetic, Venn diagram, Boolean algebra (recapitulation); Representation in SOP and POS forms; Minimization of logic expressions by algebraic method, Combinational circuits - Adder and Subtractor circuits (half & full adder & subtractor); Encoder, Decoder, Comparator, Multiplexer, De-Multiplexer and Parity Generator.	11
3.	Sequential Circuits - Basic Flip-flop & Latch, Flip-flops -SR, JK, D, T and JK Master-slave Flip Flops, Registers (SISO, SIPO, PIPO, PISO) Ring counter, Johnson counter Basic concept of Synchronous and Asynchronous counters (detail design of circuits excluded), Design of Mod N Counter.	10
4.	A/D and D/A conversion techniques – Basic concepts (D/A:R-2-R only [2L]A/D: successive approximation [2L]) Logic families-TTL, ECL, MOS and CMOS - basic concepts (2L).	6

Textbook and Reference books:

1. I.G. Nagrath, Analog Electronics, PHI
2. Analog Electronics, A.K. Maini, Khanna Publishing House
3. Microelectronics Engineering –Sedra& Smith-Oxford.
4. Principles of Electronic Devices & circuits—B L Thereja & Sedha—SChand
5. Digital Electronics – Kharate –Oxford
6. Digital Electronics – Logic & Systems by J.Bigmeil & R. Donovan; Cambridge Learning.
7. Digital Logic and State Machine Design (3rd Edition) – D.J. Comer, OUP
8. Electronic Devices & Circuit Theory – Boyelstad & Nashelsky -PHI
9. Bell-Linear IC & OPAMP—Oxford
10. P. Raja- Digital Electronics- Scitech Publications

Course Outcome:	
On completion of the course students will-be able to	
ESC-IT 301.1	Define the fundamental Analog circuits such as Amplifiers, Wein Bridge Oscillator, Multi-vibrators, Schmitt Trigger, and 555 timers.
ESC-IT 301.2	Distinguish between analog and digital system with the basic about binary number system and Boolean algebra.
ESC-IT 301.3	Demonstrates the fundamental combinational and sequential logic circuits and counters and registers.
ESC-IT 301.4	Discuss the basic concepts of logic families and realize the basic A/D and D/A conversion techniques.
ESC-IT 301.5	Formulate the combinational and sequential circuit design and minimization techniques.
ESC-IT 301.6	Validate the circuit design theory for model development of logic circuits.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	-	-	1	-	-	2	3	2	2
CO2	3	3	3	1	2	-	-	1	-	-	2	3	3	3
CO3	3	3	3	2	3	-	-	1	-	-	2	3	3	3
CO4	3	3	3	2	3	-	-	1	-	-	2	3	3	3
CO5	3	3	3	2	3	-	-	1	-	-	2	3	3	3
CO6	3	3	3	2	3	-	-	1	-	-	2	3	3	3

Data Structure & Algorithms

Code: PCC-IT-301

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To find the Time Complexity and Space Complexity for algorithm.
2.	To understand the various techniques of sorting and searching.
3.	To design and implement arrays, stacks, queues, and linked lists.
4.	To understand the complex data structures such as trees and graphs.
5.	To solve real life problems.

Pre-Requisite:	
1.	Basic Computation and Principles of C, basics of set theory, recurrence relation.

Unit	Content	Hrs.
1.	Introduction: Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space tradeoff. Searching: Linear Search and Binary Search Techniques and their complexity analysis.	8
2.	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. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis. Principles of recursion – use of stack, recursion vs iteration, tail recursion. Applications - Tower of Hanoi.	8
3.	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.	6
4.	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. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis	8
5.	Sorting and Hashing: 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, Hashing. Graph: Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.	6

Textbook and Reference books:

1. “Data Structures and Program Design In C”, 2/E by Robert L. Kruse, Bruce P. Leung.
2. “Fundamentals of Data Structures of C” by Ellis Horowitz, Sartaj Sahni, Susan Anderson- freed.
3. “Data Structures in C” by Aaron M. Tenenbaum.
4. “Data Structures” by S. Lipschutz.
5. “Data Structures” by R.S. Salaria, Khanna Publishing House
6. “Data Structures through C” by Yashwant Kanitkar, BPB House
7. “Data Structures Using C” by Reema Thareja.
8. “Data Structure Using C”, 2/e by A.K. Rath, A. K. Jagadev.
9. “Introduction to Algorithms” by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein

Course Outcome:	
On completion of the course students will-be able to	
PCC IT301.1	Recall the concepts of data structure, data types, abstract data type and various operations to develop solutions for engineering problems.
PCC IT301.2	Develop programs to implement linear data structures such as stacks, queues, linked lists etc.
PCC IT301.3	Apply the concept of trees and graph data structures in real world scenarios.
PCC IT301.4	Comprehend the implementation of sorting and searching algorithms.
PCC IT301.5	Compare Time Complexity and Space Complexity for algorithm.
PCC IT301.6	Effectively choose the data structure that efficiently models data to solve complex problem.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	-	-	-	-	-	-	2	3	2	2
CO2	3	3	2	2	2	-	-	-	-	-	2	3	3	2

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO3	3	2	2	2	2	-	-	-	-	-	2	3	2	2
CO4	3	2	2	1	2	-	-	-	-	-	2	3	2	2
CO5	3	2	2	2	1	-	-	-	-	-	2	3	2	2
CO6	3	3	2	2	2	2	-	-	-	-	2	3	3	2

Object Oriented Programming

Code: PCC-IT-302

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1.	To understand the fundamental concepts of Object-Oriented Programming (OOP) such as objects, classes, inheritance, polymorphism, abstraction, and encapsulation.
2.	To develop problem-solving skills using object-oriented design and programming techniques.
3.	To implement real-world applications using an object-oriented programming language (e.g., C++/Java/Python).

Pre-Requisite:	
1.	Basic knowledge of programming concepts (variables, data types, control structures, functions).
2.	Familiarity with any procedural programming language (such as C or Python).

Unit	Content	Hrs.
1.	Object oriented design: Concepts of object-oriented programming language, Major and minor elements, Object, Class, relationships among objects, aggregation, links, relationships among classes-association, aggregation, using, instantiation, meta-class, grouping constructs.	10
2.	Object oriented concepts: Difference between OOP and other conventional programming – advantages and disadvantages. Class, object, message passing, inheritance, encapsulation, polymorphism.	4
3.	Class & Object properties: Basic concepts of java programming – advantages of java, byte-code & JVM, data types, access specifiers, operators, control statements & loops, array, creation of class, object, constructor, finalize and garbage collection, use of method overloading, this keyword, use of objects as parameter & methods returning objects, call by value & call by reference, static variables & methods, garbage collection, nested & inner classes, basic string handling concepts- String methods, concept of mutable and immutable string, command line arguments, basics of I/O operations – keyboard input using Buffered Reader & Scanner classes.	6
4.	Reusability properties: Super class & subclasses including multilevel hierarchy, process of constructor calling in inheritance, use of super and final keywords with super () method, dynamic method dispatch, use of abstract classes & methods, interfaces. Creation of packages, importing packages, member access for packages.	6
5.	Exception handling & Multithreading: Exception handling basics, different types of exception classes, use of try & catch with throw, throws & finally, creation of user defined exception classes. Basics of multithreading, main thread, thread life cycle, creation of multiple threads, thread priorities, thread synchronization, inter-thread communication, deadlocks for threads, suspending & resuming threads.	6
6.	Collection Framework: Introduction to Collection Framework, need for collections, hierarchy of Collection interfaces and classes, commonly used classes – Array List, Linked List, Hash Map, Tree Map.	4

Textbook and Reference books:

1. Deitel and Deitel - "Java How to Program" (6th Ed.) - Pearson
2. Patrick Naughton, Herbert Schildt - "The complete reference-Java2" - TMH
3. E. Balagurusamy - " Programming with Java: A Primer" (3rd Ed.)- TMH
4. Kathy Sierra, Bert Bates (2006). Head First Java (2nd ed.). O'Reilly Media.
5. Cay Horstmann (2020). Core Java: Fundamentals (Volume I). Pearson Publisher.
6. R.K Das - "Core Java for Beginners" - Vikas Publishing

Course Outcome:	
On completion of the course students will-be able to	
PCC IT302.1	To understand and illustrate concepts of Object-Oriented Programming (OOP) language.
PCC IT302.2	To define and investigate fundamental OOP concepts such as classes, objects, inheritance, polymorphism, encapsulation, and abstraction.
PCC IT302.3	To investigate various I/O techniques and interpret string operations.
PCC IT302.4	To classify and select different types of inheritance, polymorphism, abstract classes, interfaces and packages.
PCC IT302.5	To develop the concepts of exception handling and multithreading.
PCC IT302.6	Evaluate and design efficient Java applications using appropriate Collection Framework interfaces and classes for efficient data management.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	3	3	3	2	1	2	2	2	2	2	2	3	3
CO2	3	3	3	2	1	2	-	-	1	2	2	3	3	3
CO3	3	3	2	2	1	1	-	-	-	-	2	3	3	2
CO4	3	2	2	2	2	1	-	-	1	2	2	3	2	2
CO5	3	3	3	3	2	2	-	2	1	2	2	3	3	3
CO6	3	3	3	3	2	2	2	3	2	3	2	3	3	3

Database Management System

Code: PCC-IT-303

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand the different issues involved in the design and implementation of a database system.
2.	To study the physical and logical database designs, database modeling, relational, hierarchical, and network models
3.	To understand and use data manipulation language to query, update, and manage a database
4.	To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
5.	To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.
6.	To understand the different issues involved in the design and implementation of a database system.

Pre-Requisite:	
1.	Basic knowledge of computer fundamentals and data organization.
2.	Understanding of programming concepts (variables, data types, loops, functions).

Unit	Content	Hrs.
1.	Introduction: Overview of DBMS, DBMS vs. file system, Data Abstraction, Data Independence, DDL, DML, DBMS architectures, various users of database, role of DBA	2

Unit	Content	Hrs.
2.	Database Design: Entity-relationship model, network model, relational and object-oriented data models, integrity constraints, data manipulation operations,	5
3.	Relational query languages: Relational algebra, Tuple and domain relational calculus, evaluation of relational algebra expressions, SQL, Query equivalence, Join strategies, Query optimization algorithms.	7
4.	Relational database design: Domain and data dependency, Armstrong's axioms, Functional dependency, Normal forms, Dependency preservation, Lossless design.	8
5.	Transaction processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp-based schedulers, Multi- version and optimistic Concurrency Control schemes.	9
6.	Database Recovery and File Organization: Database recovery, Indices, B-trees, Hashing.	5

Textbook and Reference books:

1. "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.
2. "Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.
3. "Fundamentals of Database Systems", 5th Edition by R. Elmasri and S. Navathe,
4. Pearson Education "Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley
5. Database Management Systems, R.P. Mahapatra, Khanna Publishing House

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT303.1	Define and understand the concept of DBMS, RDBMS, and Architecture of Database System.
PCC-IT303.2	Implement the different Database Models(ER-Model, N/W-Model, Relational Model) and apply and design ER-Model to design database system
PCC-IT303.3	Formulate and solution of Relational Algebra, Tuple & Domain Calculus, Query optimization with evaluation of relational algebra expression and apply SQL for data manipulation.
PCC-IT303.4	Implement Normalization Technique for designing database system for different applications and analyze the suitable level of Normal Form of the database system.
PCC-IT303.5	Understand and justify the aspect of Transaction processing and Concurrency control of transactions.
PCC-IT303.6	Understand, implement and evaluate the performance of Database Recovery, Database storage structures and access technique.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1	-	-	1	-	-	-	-	-	1	3	1	-
CO2	3	3	3	2	2	1	-	2	1	1	2	3	3	3
CO3	3	3	2	2	3	-	-	2	1	1	2	3	3	2
CO4	3	3	3	3	3	2	1	2	2	2	3	3	3	3
CO5	3	3	2	2	1	2	1	2	1	2	3	3	3	2
CO6	3	3	2	2	1	1	1	1	2	2	2	3	3	2

Mathematics-III (Probability & Statistics)

Code:BSC-IT-301

Contacts Hours:

3

Credit point: 3

Course Objective:

1.	To understand the fundamental concepts of probability theory and random variables.
2.	To apply statistical methods for data analysis and interpretation in engineering and IT applications.
3.	To develop analytical and problem-solving skills using probability distributions and statistical techniques.

Pre-Requisite:

1.	Knowledge of Mathematics–I (Calculus and Algebra).
----	--

Unit	Content	Hrs.
1.	Probability spaces, Axiomatic definition of Probability, Conditional probability, Independent events, Bayes theorem (Proof not required). Probability Distributions: Discrete and continuous and their properties, Distribution functions and densities, Expectation and Variance, Binomial, Poisson, Uniform, Exponential, Normal distributions. Binomial and Poisson approximation to Normal distribution. t , χ^2 and F-distribution (Definition only). Transformation of random variables. Central Limit Theorem, Law of large numbers (statement only) and their applications. Tchebychev inequalities (statement only) and its application.	12
2.	Bivariate Distributions: Two-dimensional random variable, Distribution function and its properties: Discrete and continuous, Marginal distribution, Conditional distribution, Mathematical expectation of bi-variate distribution, Conditional expectation, Correlation and regression - Rank correlation coefficient, Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas and more general curves.	10
3.	Random sampling, Parameter, Statistic and its Sampling distribution. Standard error of statistic. Sampling distribution of sample mean and variance in random sampling from a normal distribution (statement only) and related problems. Estimation of parameters: Unbiased and consistent estimators. Point estimation. Interval estimation. Maximum likelihood estimation of parameters (Binomial, Poisson and Normal). Confidence intervals and related problems.	8
4.	Simple and Composite hypothesis. Critical region. Level of significance. Type I and Type II errors. One sample and two sample tests for means and proportions. χ^2 - test for goodness of fit.	6

Textbook and Reference books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. Kapoor, V. K and Gupta, S.C.: Fundamental of Mathematical Statistics, Sultan Chand and Sons.
3. S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.
4. W. Feller, An Introduction to Probability Theory and its Applications, Vol. 1, 3rd Ed., Wiley, 1968.
5. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.

Course Outcome:	
On completion of the course students will-be able to	
BSC-IT 301.1	Recite concept of permutation and combination, concept of statistics.
BSC-IT 301.2	Discuss the concept probability distribution, statistical inference and hypothesis testing.
BSC-IT 301.3	Demonstrate computational modelling of biological phenomena and applies techniques from areas such as artificial intelligence, data base, software engineering, machine learning, image processing.
BSC-IT 301.4	Illustrate physical scenario and classify them to recognize the best fit physical and logical models.
BSC-IT 301.5	Compare different mathematical results during the process of problem analysis.
BSC-IT 301.6	Design models to demonstrate industrial problem for emerging trend in information technology.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2	2	1	1	1	1	1	1	2	3	3
CO2	3	3	3	2	2	1	1	1	1	1	1	2	3	2

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO3	3	3	3	2	2	1	1	1	1	1	1	2	3	2
CO4	3	3	3	2	2	1	1	1	1	1	1	2	3	3
CO5	3	3	3	2	2	1	1	1	1	1	1	2	3	3
CO6	3	3	3	2	2	1	1	1	1	1	1	2	3	2

Economics for Engineers (Humanities-II)

Code: HSMC-IT-301

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	Understand basic economic concepts and their relevance to engineering decision-making.
2.	Apply economic principles to analyze engineering projects and assess their feasibility.
3.	Develop skills in cost estimation, project evaluation, and risk analysis.
4.	Gain insights into the economic implications of engineering decisions on society and the environment.

Pre-Requisite:	
1.	Basic understanding of Mathematics (algebra and graphs).
2.	General awareness of economic and business environments.

Unit	Content	Hrs.
1.	Introduction to Engineering Economy: Origin of Engineering Economy, Principles of Engineering Economy, Role of Engineers in Decision Making Inflation and Price Change: Definition, Effects, Causes, Price Change with Indexes, Types of Index, Composite vs. Commodity Indexes, Use of Price Indexes in Engineering Economic Analysis, Cash Flows that inflate at different Rates.	10
2.	Time Value of Money: Introduction to Time Value of Money, Simple Interest, Compound Interest, Nominal Interest rate, Effective Interest rate, Continuous Compounding, Economic Equivalence, Development of Interest Formulas, The Five Types of Cash flows, Single Cash flow Formulas, Uneven Payment Series, Equal Payment Series.	7
3.	Methods of comparison of alternatives: NPV, Profitability Index or Benefit Cost Ratio, Payback Period Method, Equivalent Worth Methods, Present Worth Method, Future Worth Method, Annual Worth Method, Rate of Return Methods (IRR and ARR)	7
4.	Engineering Costs & Estimation: Elements of cost (Fixed, Variable, Marginal & Average Costs, Sunk Costs, Opportunity Costs, Recurring and Nonrecurring Costs, Incremental Costs, Cash Costs vs Book Costs, Life-Cycle Costs) and cost estimation models (Per-Unit Model, Segmenting Model, Cost Indexes, Power-Sizing Model, Improvement & Learning Curve), Concept of Revenue, Break even analysis, Cost sheet.	6

Text book and Reference books:

1. Donald Newnan, Ted Eschembach, Jerome Lavelle: Engineering Economics Analysis, OUP
2. R. Paneer Seelvan: Engineering Economics, PHI
3. Sullivan and Wicks: Engineering Economy, Pearson
4. John A. White, Kenneth E. Case, David B. Pratt: Principle of Engineering Economic Analysis, John Wiley
5. James L. Riggs, David D. Bedworth, Sabah U. Randhawa: Economics for Engineers 4e, Tata Mc Graw - Hill

Course Outcome:	
On completion of the course students will-be able to	
CO 1	Students will recall and explain fundamental concepts of engineering economics.

CO 2	Students will apply economic principles and techniques to analyze engineering projects and make informed decisions based on economic criteria.
CO 3	Students will analyze project cost structures, estimate costs using appropriate methods, and evaluate cost-effectiveness of the engineering projects using NPV, IRR, BCR etc.
CO 4	Students will integrate economic sustainability considerations into engineering design and decision-making processes by assessing project risk through sensitivity analysis.
CO 5	Students will engage in critical analysis of economic challenges, synthesizing information to devise innovative solutions for engineering problems.
CO 6	Students will be grounded in ethical considerations, addressing both economic constraints and societal needs.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	-	1	1	1	1	1	1	1	1	1	1	-	1	1
CO2	2	2	2	2	2	2	2	-	2	-	2	2	2	2
CO3	3	2	2	2	2	2	2	-	2	2	-	3	2	2
CO4	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO6	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Environmental Sciences

Code:MC-IT-301

Contacts Hours:

1

Credit point: 0

Course Objective:	
1.	Be able to understand the natural environment and its relationships with human activities.
2.	Be able to apply the fundamental knowledge of science and engineering to assess environmental and health risk.
3.	Be able to understand environmental laws and regulations to develop guidelines and procedures for health and safety issues.
4.	Be able to solve scientific problem-solving related to air, water, noise & land pollution

Pre-Requisite:	
1.	Basic knowledge of Environmental science

Unit	Content	Hrs.
1.	Basic ideas of environment, basic concepts, man, society & environment, their interrelationship (1L) Mathematics of population growth and associated problems, Importance of population study in environmental engineering, definition of resource, types of resource, renewable, non-renewable, potentially renewable, effect of excessive use vis-à-vis population growth, Sustainable Development. (1L) Environmental degradation: Natural environmental Hazards Anthropogenic degradation, Nature and scope of Environmental Science and Engineering. (1L)	3
2.	Elements of ecology: System, open system, closed system, definition of ecology, species, population, community, definition of ecosystem- components types and function. (1L) Structure and function of the following ecosystem: Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems, Mangrove ecosystem (special reference to Sundar ban); Food chain [definition and one example of each food chain], Food web (1L). Biodiversity- types, importance, Endemic species, Biodiversity Hot-spot, Threats to biodiversity, Conservation of biodiversity (1L).	3
3.	Atmospheric Composition: Troposphere, Stratosphere, Mesosphere, Thermosphere, Tropopause and Mesopause. Green house effects: Definition, impact of greenhouse gases on the global climate and consequently on sea water level, agriculture and marine food. Global warming and its consequence, Control of Global warming. Earth's heat budget(1L)	4

Unit	Content	Hrs.
	Lapse rate: Ambient lapse rate Adiabatic lapse rate, atmospheric stability, temperature inversion (radiation inversion) (1L). Definition of pollutants and contaminants, Primary and secondary pollutants: emission standard, criteria pollutant. Sources and effect of different air pollutants Smog Photochemical smog, and London smog. Depletion Ozone layer: CFC, destruction of ozone layer by CFC, impact of other green-house gases, effect of ozone modification (1L). Standards and control measures: Industrial, commercial and residential air quality standard, control measure (ESP. cyclone separator, bag house, catalytic converter, scrubber (ventury), Statement with brief reference). (1L)	
4.	Hydrosphere, Hydrological cycle and Natural water. Pollutants of water, their origin and effects: Oxygen demanding wastes, pathogens, nutrients, Salts, thermal application, heavy metals, pesticides, volatile organic compounds. (1L) River/Lake/ground water pollution: River: DO, 5-day BOD test, Seeded BOD test, BOD reaction rate constants, Effect of oxygen demanding wastes on river [deoxygenation, reaeration], COD, Oil, Greases, pH (2L). Lake: Eutrophication [Definition, source and effect]. (1L) Ground water: Aquifers, hydraulic gradient, ground water flow (Definition only) (1L) Standard and control: Waste water standard [BOD, COD, Oil, Grease], Water Treatment system [coagulation and flocculation, sedimentation and filtration, disinfection, hardness and alkalinity, softening] Waste water treatment system, primary and secondary treatments [Trickling filters, rotating biological contractor, Activated sludge, sludge treatment, oxidation ponds] tertiary treatment definition. (2L) Water pollution due to the toxic elements and their biochemical effects: Lead, Mercury, Cadmium, and Arsenic (1L)	8
5.	Lithosphere; Internal structure of earth, rock and soil (1L)	1

Textbook and Reference books:

1. Masters, G. M., "Introduction to Environmental Engineering and Science", Prentice- Hall of India Pvt.Ltd.,1991.
2. Erach Bharucha, Environmental Studies, University Press
3. M.P. Poonia, Environmental Studies, Khanna Publishing House
4. De, A. K., "Environmental Chemistry", New Age International
5. Rajagopalan, Environmental Studies, Oxford University Press

Course Outcome:	
On completion of the course students will-be able to	
MC-IT 301.1	Articulate the interconnected and interdisciplinary nature of environmental studies
MC-IT 301.2	Demonstrate an integrative approach to environmental issues with a focus on sustainability
MC-IT 301.3	Use critical thinking, problem-solving, and the methodological approaches in environmental problem solving
MC-IT301.4	Communicate complex environmental information to both technical and non-technical audiences.
MC-IT 301.5	Understand and evaluate the global scale of environmental problems.
MC-IT 301.6	Reflect critically on their roles, responsibilities, and identities as citizens, consumers and environmental actors in a complex, interconnected world.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
CO2	2	2	2	2	2	2	2	1	2	1	2	2	2	2
CO3	3	1	2	2	2	2	2	1	2	2	1	3	1	2
CO4	1	1	2	1	2	1	-	3	1	3	1	1	1	2
CO5	3	1	1	3	1	2	2	2	1	1	3	3	1	1
CO6	3	1	1	2	2	1	1	3	3	1	2	3	1	1

IT Workshop I (Python)

Code: ESC-IT-391

Contacts Hours:

4

Credit point: 2

Course Objective:	
1.	To introduce students to programming using Python and/or R for problem solving and data analysis.
2.	To develop practical skills in writing, debugging, and executing programs.
3.	To apply programming concepts to real-world applications such as data handling, visualization, and basic statistical analysis.

Pre-Requisite:	
1.	Knowledge of Programming Logic
2.	Experience with a high-level language (C/C++,) is suggested.
3.	Prior knowledge of a scripting language and Object-Oriented concepts is helpful but not mandatory.

Laboratory Experiments:	
1.	Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.
2.	Types, Operators and Expressions: Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elseif-else, for, while, break, continue, pass.
3.	Data Structures Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.
4.	Functions - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function - Global and Local Variables. Modules: Creating modules, import statement, from. Import statement, name spacing, Python packages Introduction to PIP, Installing Packages via PIP, Using Python Packages.
5.	Object Oriented Programming OOP in Python: Classes, 'self-variable', Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding, Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions.
6.	Brief Tour of the Standard Library - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics Testing: Why testing is required? Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

Textbook and Reference books:

1. Python Crash Course – Eric Matthes, No Starch Press.
2. Programming in Python 3 – Mark Summerfield, Addison-Wesley Professional.
3. Learning Python – Mark Lutz, O'Reilly Media.
4. Core Python Programming – R. Nageswara Rao, Dreamtech Press.
5. Automate the Boring Stuff with Python – Al Sweigart, No Starch Press.
6. Think Python – Allen B. Downey, O'Reilly Media.

Course Outcome:	
On completion of the course students will-be able to	

ESC-IT391.1	Recall the fundamental concepts of Python programming, including its history, need, features, syntax, data types, and basic input/output operations.
ESC-IT391.2	Explain the behavior of Python control structures, operators, and expressions in various scenarios to predict program output.
ESC-IT391.3	Apply built-in Python data structures (lists, tuples, sets, dictionaries) and control structures to solve computational problems.
ESC-IT391.4	Analyze Python functions and modular programming constructs to identify scope, argument passing, and code organization using packages and modules.
ESC-IT391.5	Evaluate object-oriented principles and exception handling techniques in Python to design secure and reusable software components.
ESC-IT391.6	Design and develop complete Python programs using standard libraries for multithreading, GUI, testing, and file handling to address real-world applications.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1	-	-	2	-	-	-	1	-	2	3	1	-
CO2	3	2	1	1	2	-	-	-	2	-	2	3	2	1
CO3	3	2	3	2	3	1	-	1	2	-	3	3	2	3
CO4	3	3	3	2	3	1	-	2	2	1	3	3	3	3
CO5	3	3	3	2	3	1	1	2	2	1	3	3	3	3
CO6	3	3	3	2	3	2	1	2	2	2	3	3	3	3

Data Structure & Algorithms Lab

Code: PCC-IT-391

Contacts Hours:

4

Credit point: 2

Course Objective:	
1.	To provide practical exposure in implementing fundamental data structures and algorithms using C programming.
2.	To develop the ability to design and analyze efficient solutions using appropriate data structures for real-world problems.
3.	To enhance programming, logical reasoning, and problem-solving skills through hands-on laboratory practice.

Pre-Requisite:	
1.	Pre-requisites as in PCC-IT301

Laboratory Experiments:	
1.	Implementations of different operations on array.
2.	Operations on stacks, queues and linked lists.
3.	Applications of stack to convert infix to postfix and evaluation of postfix or prefix expressions.
4.	Application of linked lists- polynomial representation, addition.
5.	Implementation of priority queue.
6.	Implementation of Binary Tree, Binary Search Tree, recursive and non-recursive traversal of trees.
7.	Implementation of BFS, DFS – Application of stack, queues.

8.	Implementation of sorting and searching algorithms.
----	---

Textbook and Reference books:

1. Data Structures and Algorithms Made Easy – Narasimha Karumanchi, CareerMonk Publications.
2. Data Structures Using C – Reema Thareja, Oxford University Press.

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT391.1	Apply and implement the learned algorithms related to array, stack, queue for problem solving.
PCC-IT391.2	Identify the data structure to develop program for real time applications.
PCC-IT391.3	Design and develop optimal algorithms using appropriate data structures.
PCC-IT391.4	Implementation of priority queue for optimal algorithms.
PCC-IT391.5	Implement Binary Tree, Binary Search Tree, AVL Trees for problem solving.
PCC-IT391.6	Implement sorting techniques for real time applications.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	-	-	-	-	-	2	3	2	2
CO2	3	2	2	2	2	-	-	-	-	-	2	3	2	2
CO3	3	2	2	2	2	-	-	-	-	-	2	3	2	2
CO4	3	2	2	1	2	-	-	-	-	-	2	3	2	2
CO5	3	3	3	2	2	-	-	-	-	-	3	3	3	3
CO6	3	3	3	3	2	-	-	-	-	-	3	3	3	3

Object Oriented Programming Lab

Code: PCC-IT-392

Contacts Hours:

4

Credit point: 2

Course Objective:	
1.	To provide hands-on experience in implementing Object Oriented Programming concepts such as classes, objects, inheritance, polymorphism, encapsulation, and abstraction.
2.	To develop programming skills for designing and building modular, reusable, and maintainable software applications.
3.	To apply object-oriented techniques to solve real-world problems through practical laboratory exercises.

Pre-Requisite:	
1.	Pre-requisites as in PCC-IT302

Laboratory Experiments:	
1.	Assignments on class, constructor, overloading, inheritance, overriding.

2.	Assignments on wrapper class, arrays.
3.	Assignments on developing interfaces- multiple inheritance, extending interfaces.
4.	Assignments on creating and accessing packages.
5.	Assignments on multi-threaded programming.
6.	Implementation of the Java Collection Framework, including Collection hierarchy, ArrayList, LinkedList, HashMap, and TreeMap for efficient data organization and processing.

Textbook and Reference books:

1. Kathy Sierra, Bert Bates (2006). Head First Java (2nd ed.). O'Reilly Media.
2. Cay Horstmann (2020). Core Java: Fundamentals (Volume I). Pearson Publisher.

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT392.1	To apply the ideas of class, object and constructor.
PCC-IT392.2	To implement different types of inheritance and polymorphism.
PCC-IT392.3	To assemble a variety of I/O techniques and string operations.
PCC-IT392.4	To create abstract class, interface and package related programs.
PCC-IT392.5	To develop the concepts of exception handling and multithreading.
PCC-IT392.6	Evaluate and develop Java applications using appropriate Collection Framework interfaces and classes.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	-	3	-	-	-	-	-	2	3	2	2
CO2	3	2	3	-	3	-	-	-	-	-	2	3	2	3
CO3	3	2	2	-	3	-	-	-	-	-	2	3	2	2
CO4	3	2	3	-	3	-	-	2	2	-	2	3	2	3
CO5	3	2	3	2	3	-	-	2	2	-	2	3	2	3
CO6	3	2	3	2	3	-	-	2	3	1	2	3	2	3

Database Management System Lab

Code: PCC-IT-393

Contacts Hours:

4

Credit point: 2

Course Objective:	
1.	To provide hands-on experience in designing, creating, and managing databases using SQL.
2.	To develop skills in writing queries, procedures, triggers, and views for data manipulation and retrieval.
3.	To implement database concepts such as normalization, constraints, indexing, and transaction management through practical exercises.

Pre-Requisite:

1.	Basic knowledge of Database Management System concepts.
2.	Familiarity with any programming language and basic understanding of data structures.

Unit	Content
1.	Creating Database: Creating a Database, Creating a Table, Specifying Relational Data Types, Specifying Constraints, and Creating Indexes.
2.	Table and Record Handling: INSERT statement, Using SELECT and INSERT together, DELETE, UPDATE, TRUNCATE statements, DROP, ALTER statements.
3.	Retrieving Data from a Database: (a) The SELECT statement, Using the WHERE clause, Using Logical Operators in the WHERE clause, Using IN, BETWEEN, LIKE, ORDER BY, GROUP BY and HAVING (b) Clause: Using Aggregate Functions, Combining Tables Using JOINS, Subqueries
4.	Database Management: Creating Views, Creating Column Aliases, Creating Database Users, Using GRANT and REVOKE
5.	Cursors in Oracle PL / SQL
6.	Writing Oracle PL / SQL Stored Procedures

Textbook and Reference books:

1. SQL, PL/SQL: The Programming Language of Oracle – Ivan Bayross, BPB Publications.
2. SQL in 10 Minutes, Sams Teach Yourself – Ben Forta, Pearson.

Course Outcome:	
On completion of the course students will-be able to	
PCC IT393.1	Understand through laboratory activities to solve problems related to key concepts taught in the classroom.
PCC-IT393.2	Create and populate a RDBMS, using SQL.
PCC-IT393.3	Implement queries in SQL to retrieve any type of information from a database.
PCC-IT393.4	Design conceptual models of a database using ER modelling for real life applications and also construct queries in Relational Algebra.
PCC-IT393.5	Create and demonstrate various activities on tables using PL/SQL
PCC-IT393.6	Formulate problem definition statements for real life applications and implement a database for the same.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2	2	1	-	-	1	1	1	2	2	2	2
CO2	2	2	2	2	2	-	-	2	1	1	2	2	2	2
CO3	2	2	2	2	2	-	-	2	1	1	2	2	2	2
CO4	2	2	2	3	1	-	-	1	1	2	3	2	2	2
CO5	3	3	3	2	1	-	-	1	1	1	2	3	3	3
CO6	3	3	3	3	3	-	1	3	3	3	3	3	3	3

Fourth Semester (2nd Year)

Discrete Mathematics

Code: BSC-IT-401

Contacts

Hours: 3

Credit point: 3

Course Objective:

1.	Use mathematically correct terminology and notation.
2.	Construct correct direct and indirect proofs.
3.	To know Syntax, Semantics, Validity and Satisfiability, Graphs and Trees
4.	Use counterexamples. Apply logical reasoning to solve a variety of problems.

Pre-Requisite:

1.	Some concepts from basic math – algebra, geometry
----	---

Unit	Content	Hrs.
1.	Sets, Relation and Function: Operations and Laws of Sets, Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Image of a Set, Sum and Product of Functions, Bijective functions, Inverse and Composite Function, Size of a Set, Finite and infinite Sets, Countable and uncountable Sets, Cantor's diagonal argument and The Power Set theorem, Schroeder-Bernstein theorem. Principles of Mathematical Induction: The Well- Ordering Principle, Recursive definition, The Division algorithm: Prime Numbers, The Greatest Common Divisor: Euclidean Algorithm, The Fundamental Theorem of Arithmetic.	8
2.	Basic counting techniques-inclusion and exclusion, pigeon-hole principle, permutation and combination	5
3.	Propositional Logic: Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, The use of Quantifiers. Proof Techniques: Some Terminology, Proof Methods and Strategies, Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency.	8
4.	Algebraic Structures and Morphism: Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups, Algebraic Structures with two Binary Operation, Rings, Integral Domain and Fields. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form	7
5.	Graphs and Trees: Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Colouring, Colouring maps and Planar Graphs, Colouring Vertices, Colouring Edges, List Colouring, Perfect Graph, definition properties and Example, rooted trees, trees and sorting, weighted trees and prefix codes, Bi-connected component and Articulation Points, Shortest distances.	8

Textbook and Reference books:

1. Russell Merris, Combinatorics, Wiley-Interscience series in Discrete Mathematics and Optimization
2. N. Chandrasekaran and M. Umaparvathi, Discrete Mathematics, PHI
3. Gary Chartrand and Ping Zhang – Introduction to Graph Theory, TMH
4. J.K. Sharma, Discrete Mathematics, Macmillan
5. Winfried Karl Grassmann and Jean-Paul Tremblay, Logic and Discrete Mathematics, PEARSON.
6. S. K. Chakraborty and B. K. Sarkar, Discrete Mathematics, OXFORD University Press.

Course Outcome:

On completion of the course students will be able to	
BSC-IT401.1	Represent, solve and formulate systems of linear equations, which are fundamental in engineering for modelling various physical problems; eigen values /eigen vectors to understand dynamic behavior of systems and analyze their stability, multivariate statistics system analysis.
BSC-IT401.2	Arrange and assess knowledge of characteristics of function at intermediate points, continuity pertaining to proper and improper integrals leading to convergence, convergence of sequence and series.
BSC-IT401.3	Model complex systems with several variables to understand their interactions; comprehend optimization in multidimensional spaces.
BSC-IT401.4	Describe, analyze and compose physical phenomena that involve quantities with both magnitude and direction amalgamated with the concept of gradient, divergence, and curl to ascertain how quantities change in space and time.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	-	1	1	1	1	1	1	1	1	1	1	-	1	1
CO2	2	2	2	2	2	2	2	-	2	-	2	2	2	2
CO3	3	2	2	2	2	2	2	-	2	2	-	3	2	2
CO4	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO6	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Computer Organization & Architecture

Code: PCC-IT-402

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1	To prepare students to perform the analysis and design of various digital electronic circuits.
2	To know how Computer systems, work & its basic principles
3	To know how I/O devices are being accessed and its principles etc.
4	To learn the basics of stored program concepts.
5	To learn the principles of pipelining
6	To learn mechanism of data storage
7	To distinguish between the concepts of serial, parallel, pipeline architecture.

Pre-Requisite:	
1.	Concept of basic components of a digital computer, Basic concept of Fundamentals & Programme structures. Boolean Algebra
2.	Concept of basic components of a digital computer, Basic concept of Fundamentals & Programme structures. Boolean Algebra

Unit	Content	Hrs.
1	Introduction to Computer and Computer Arithmetic: Von Neumann and Harvard Architecture, Computer organization vs Computer Architecture, Instruction format, Addressing modes, Half adder, Full adder, Ripple carry adder, Carry Look-ahead adder, Design of ALU. Multiplication algorithm, Division algorithm, Floating point number representation - IEEE 754 standard.	8
2	Memory Organization and I/O techniques: Memory organization, static and dynamic memory, memory hierarchy, associative memory. Inclusion, coherence and locality properties, Cache memory organization, Memory replacement policies, Techniques for reducing cache misses, Virtual memory organization, Mapping and management techniques, Modes of transfer, Handshaking and DMA.	8

Unit	Content	Hrs.
3	Pipeline and ILP: Introduction to pipeline, Instruction pipeline, Arithmetic pipeline, processor pipeline, Types of Pipeline hazards and its countermeasures, Super-pipeline, Superscalar and VLIW architecture. Introduction to ILP and techniques to improve ILP, Array and Vector processor.	10
4	Multiprocessor Architecture and Control Unit: Taxonomy of parallel architectures, Types of Multiprocessor architectures, Cache inconsistency, Centralized and Distributed shared memory architecture, Memory Consistency, Cluster computers, RISC and CISC architecture. Introduction to Control unit, Hardwired CU and Micro programmed CU	10

Text book and Reference books:

1. Computer Organization and Architecture: Designing for Performance – William Stallings, Pearson.
2. Computer System Architecture – M. Morris Mano, Pearson.
3. Computer Organization and Design – David A. Patterson and John L. Hennessy, Morgan Kaufmann.

Course Outcome:	
On completion of the course students will be able to	
PCC IT402.1	Describe and explain the difference between computer organization and computer architecture.
PCC IT402.2	Design the ALU for different arithmetical and logical problems and apply the knowledge of different multiplication and division algorithm.
PCC IT402.3	Formulate design methodology for using various types of instructions.
PCC IT402.4	Differentiate between different Memory hierarchy (Primary, Secondary, Cache). Able to solve different kind of numerical based on memory technologies and page replacement techniques.
PCC IT402.5	Differentiate between types of pipeline, hazards and selecting remedial techniques to handle the hazards. able to distinguish between parallel architectures. Compare performance parameters of pipelines and deduce derivations to demonstrate change in performance parameters when branching is introduced. Able to solve numerical based on pipeline concepts
PCC IT402.6	Comparing techniques of ILP, types of CU, types of shared memory architectures. Distinguish between different multiprocessor architectures, Data Flow architecture, RISC and CISC architecture

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	-	1	1	-	-	-	-	-	-	1	3	-	1
CO2	2	2	2	2	1	-	-	-	-	-	1	2	2	2
CO3	2	3	2	1	1	-	-	-	-	-	2	2	3	2
CO4	3	3	3	2	2	-	-	-	-	-	2	3	3	3
CO5	3	3	3	3	2	-	-	-	-	1	1	3	3	3
CO6	3	3	3	3	3	2	1	-	-	1	2	3	3	3

Formal Language & Automata Theory

Code: PCC-IT-403

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1	To know about Chomsky hierarchy for organizing languages
2	To introduce concepts of formal language and automata theory
3	To identify different formal language classes and their relationships
4	To design grammars and recognizers for different formal languages

5	To understand undecidability and decide on languages that are undecidable
---	---

Pre-Requisite:	
1.	Basics of Discrete Mathematics-Sets, Functions and Relations.

Unit	Content	Hrs.
1	Mathematical Preliminaries: Sets, Relations, Functions, Logical statements, Proof Techniques, mathematical induction, recursive definitions.	2
2	Finite Automata: alphabets, strings, languages, grammars, automata, Chomsky classification, deterministic finite automata (DFA), formal definition, transition graph, transition table, language of DFA, non-deterministic finite automata (NFA), NFA with epsilon transition, language of NFA, regular languages, equivalence of NFA and DFA, minimization of FA, FA with output: Moore and Mealy machine, Equivalence of Moore and Mealy Machine, applications and limitation of FA.	8
3	Regular Expression (RE): definition, operators and their precedence, algebraic laws for RE, Kleen's Theorem, RE to FA, FA to RE, Arden's theorem, non-regular languages, pumping lemma for RL and its application, closure properties of RL, applications of RE. Regular Grammar: Constituents, definition, right linear and left linear grammar, derivations, examples.	7
4	Context Free Grammar (CFG) and Context Free Language (CFL): Definition, examples, derivations, derivation trees, ambiguity in grammar, removing ambiguity in CFG, inherent ambiguity, Simplification of CFGs, normal forms for CFGs: CNF and GNF, Pumping lemma for CFLs, Closure properties of CFLs.	7
5	Push Down Automata (PDA): Description and definition, FA vs PDA, Instantaneous Description, state transition diagram, language of PDA, acceptance by final state, acceptance by empty stack, Deterministic PDA, Conversion of CFG to PDA and PDA to CFG. Concept of context-sensitive grammars and languages, linear bounded automata.	6
6	Turing machines: Basic model, definition and representation, Instantaneous Description, transition graph, language accepted by TM, variants of TM, Universal Turing Machine (UTM), TM as computer of integer functions, Recursive and recursively enumerable languages, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators. Church-Turing thesis, Halting Problem, universal and diagonalization languages, reducibility and undecidable problems, undecidable problems for CFL and recursively enumeration language.	6

Text book and Reference books:

1. Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2001). *Introduction to Automata Theory, Languages, and Computation*. Pearson Education Asia.
2. Lewis, H. R., & Papadimitriou, C. H. (1997). *Elements of the Theory of Computation*. Pearson Education Asia.
3. Kozen, D. C. (1997). *Automata and Computability*. Springer.
4. Sipser, M. (2006). *Introduction to the Theory of Computation*. PWS Publishing.
5. Martin, J. C. (2010). *Introduction to Languages and the Theory of Computation*. Tata McGraw-Hill.
6. Patel, R. B. (2012). *Theory of Computation*. Khanna Publishing House.
7. Mishra. (2008). *Theory of Computers*. PHI Publications.

Course Outcome:	
On completion of the course students will be able to	
PCCIT403.1	Acquire a fundamental understanding of the core concepts in automata theory and formal languages.
PCCIT403.2	Design finite automata or regular expression or grammar for any tokenization task.
PCCIT403.3	Construct a context free grammar, PDA for parsing any language.
PCCIT403.4	Design Turing machine for any language.
PCCIT403.5	Conclude the decidable / un decidable nature of any language.
PCCIT403.6	Apply mathematical and formal techniques for solving real-world problems

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	-	-	-	-	-	-	2	3	2	1
CO2	2	3	2	1	-	-	-	-	-	-	2	2	3	2
CO3	3	3	2	3	-	-	-	-	-	-	1	3	3	2
CO4	3	3	3	3	-	-	-	-	-	-	1	3	3	3
CO5	3	2	3	3	-	-	-	-	-	-	2	3	2	3
CO6	3	3	3	3	-	-	-	-	-	-	2	3	3	3

Design & Analysis of Algorithms

Code: PCC-IT-404

Contacts

Hours: 3

Credit point: 3

Course Objective:

1.	The aim of this module is to learn how to develop efficient algorithms for simple computational tasks and reasoning about the correctness of them
2.	Through the complexity measures, different range of behaviors of algorithms and the notion of tractable and intractable problems will be understood.

Pre-Requisite:

1.	To know data-structure and basic programming ability
----	--

Unit	Content	Hrs.
1	Introduction: Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem	8
2	Fundamental Algorithmic Strategies: Brute-Force, Greedy, Dynamic Programming, Branch and- Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack TSP. Heuristics –characteristics and their application domains.	8
3	Graph and Tree Algorithms: Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.	6
4	Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques.	10
5.	Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE	4

Textbook and Reference books:

1. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
2. Fundamentals of Algorithms – E. Horowitz et al.
3. Algorithm Design, 1ST Edition, Jon Kleinberg and Éva Tardos, Pearson.
4. Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley.

Course Outcome:

On completion of the course students will be able to

PCC IT404.1	To understand and develop the ability to analyze the running time and verify the correctness of basic algorithms.
PCC IT404.2	To implement major algorithm paradigms such as divide and conquer, greedy method, dynamic programming, branch and bound and backtracking, to a variety of real-world problems to design a good algorithm.
PCC IT404.3	To argue the correctness and efficiency of algorithms such as DFS, BFS, Shortest path, Minimum Spanning Tree and Network Flow.
PCC IT404.4	To be able to apply and design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques.
PCC IT404.5	To assemble the notions of P, NP, NP-complete, and NP-hard and to compare and differentiate with deterministic algorithms.
PCC IT404.6	To be able to develop the concept of simple approximation algorithms.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	1	-	1	2	-	2	3	3	2
CO2	3	3	3	2	2	1	-	1	2	-	2	3	3	3
CO3	3	3	2	2	2	1	-	1	2	-	2	3	3	2
CO4	3	3	3	2	2	1	-	1	2	-	2	3	3	3
CO5	3	3	2	2	1	1	-	1	2	-	2	3	3	2
CO6	3	3	3	2	2	1	-	1	2	-	2	3	3	3

Operating System

Code: PCC-IT-405

Contacts

Hours: 3

Credit point: 3

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

Pre-Requisite:	
1.	Computer Organization & Architecture

Unit	Content	Hrs.
1	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. Case study on UNIX and WINDOWS Operating System.	3
2	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: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.	10
3.	Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing. Classical IPC Problems: Reader's & Writer Problem, Dining Philosopher Problematic.	5

Unit	Content	Hrs.
4.	Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.	5
5.	Memory Management: Basic concepts; logical and physical address mapping. Memory Allocation: Contiguous memory allocation – fixed and variable partitioning; internal and external fragmentation; compaction. Paging: Principle of operation; page allocation; hardware support for paging; protection and sharing; disadvantages of paging. Virtual Memory: Basics of virtual memory; hardware and control structures; locality of reference; page faults; working set; dirty page/dirty bit; demand paging. Page Replacement Algorithms: Optimal; First-In, First-Out (FIFO); Second Chance (SC); Not Recently Used (NRU); Least Recently Used (LRU).	8
6.	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, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance. Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks	6

Text book and Reference books:

1. Operating System Concepts Essentials, 9th Edition by Avi Silberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.
3. Operating System Concepts, Ekta Walia, Khanna Publishing House, New Delhi (AICTE Recommended Textbook – 2018)
4. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
5. Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison- Wesley
6. Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India
7. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

Course Outcome:	
On completion of the course students will be able to	
PCC-IT405.1	Describe the structure, functionalities, and types of operating systems along with system calls and architectures
PCC-IT405.2	Explain process states, threads, context switching, and illustrate different CPU scheduling and real-time scheduling algorithms.
PCC-IT405.3	Analyze inter-process communication mechanisms and evaluate synchronization techniques including semaphores and monitors.
PCC-IT405.4	Model deadlock handling strategies and apply resource allocation techniques such as Banker's algorithm and detection/recovery models.
PCC-IT405.5	Compare memory allocation techniques and develop solutions using virtual memory and page replacement algorithms.
PCC-IT405.6	Design file systems, directories, disk management, and construct efficient access and allocation methods.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	1	-	-	1	-	2	3	2	-
CO2	3	3	-	-	2	-	-	-	1	-	1	3	3	-
CO3	2	3	2	2	2	-	-	-	-	-	2	2	3	2
CO4	2	3	2	2	2	1	-	-	-	-	2	2	3	2
CO5	2	2	3	2	2	-	-	-	-	-	2	2	2	3
CO6	2	1	3	2	2	2	-	1	1	1	2	2	1	3

Artificial Intelligence

Code: PCC-IT-406

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1.	To understand the fundamental concepts and techniques of Artificial Intelligence including search strategies, knowledge representation, and reasoning.
2.	To develop problem-solving skills using AI algorithms such as informed search, machine learning basics, and heuristic methods.
3.	To apply AI techniques for building intelligent systems in real-world applications.

Pre-Requisite:	
1.	Basic knowledge of Data Structures and Algorithms.
2.	Understanding of Probability, Statistics, and discrete mathematics.

Unit	Content	Hrs.
1	Introduction: Overview of Artificial intelligence- Problems of AI, AI technique, Tic - Tac - Toe problem. Intelligent Agents: Agents & environment, nature of environment, structure of agents, goal-based agents, utility-based agents, learning agents. Problem Solving: Problems, Problem Space & search: Defining the problem as state space search, production system, problem characteristics, issues in the design of search programs.	6
2.	Search techniques: Solving problems by searching: problem solving agents, searching for solutions; uniform search strategies: breadth first search, depth first search, depth limited search, bidirectional search, comparing uniform search strategies. Heuristic search strategies: Greedy best-first search, A* search, memory bounded heuristic search: local search algorithms & optimization problems: Hill climbing search, simulated annealing search, local beam search, genetic algorithms; constraint satisfaction problems, local search for constraint satisfaction problems. Adversarial search: Games, optimal decisions & strategies in games, the minimax search procedure, alpha-beta pruning, additional refinements, iterative deepening.	13
3	Knowledge & reasoning: Knowledge representation issues, representation & mapping, approaches to knowledge representation, issues in knowledge representation.	3
4	Using predicate logic: Representing simple fact in logic, representing instant & ISA relationship, computable functions & predicates, resolution, natural deduction. Probabilistic reasoning: Representing knowledge in an uncertain domain, the semantics of Bayesian networks, Dempster-Shafer theory, Fuzzy sets & fuzzy logics.	6
5	Natural Language processing: Introduction, Syntactic processing, semantic analysis, discourse & pragmatic processing. Learning: Forms of learning, inductive learning, learning decision trees, explanation-based learning, learning using relevance information, Introduction to neural network & genetic Algorithm. Expert Systems: Representing and using domain knowledge, expert system shells, knowledge acquisition.	6

Textbook and Reference books:

1. Artificial Intelligence, Ritch & Knight, TMH
2. Artificial Intelligence A Modern Approach, Stuart Russel Peter Norvig Pearson
3. Artificial Intelligence, A Classical Approach, Munish Chandra Trivedi, Khanna Publishing
4. Introduction to Artificial Intelligence & Expert Systems, Patterson, PHI
5. Poole, Computational Intelligence, OUP

6. Logic & Prolog Programming, Saroj Kaushik, New Age International

7. Expert Systems, Giarranto, VIKAS

Course Outcome:	
On completion of the course students will be able to	
PCC IT 406.1	To understand and develop the fundamental concepts of AI with the ability of problem solving.
PCC IT 406.2	To classify the basic search techniques with examples and to investigate valid solutions for problems involving different search techniques.
PCC IT 406.3	To formulate valid solutions for problems involving uncertain inputs or outcomes by using decision making techniques.
PCC IT 406.4	To understand and evaluate different NLP applications referring AI and ML technique for this process.
PCC IT 406.5	To Examine the issues involved in knowledge bases, reasoning systems and to design intelligent expert models for perception and prediction from intelligent environment.
PCC IT 406.6	To formulate complex problems and able to solve them using various AI techniques.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	-	-	-	1	-	2	3	2	2
CO2	3	3	2	2	2	-	-	1	1	-	2	3	3	2
CO3	3	3	2	2	2	-	-	-	1	1	2	3	3	2
CO4	3	3	2	2	3	1	1	-	2	1	2	3	3	2
CO5	3	3	3	2	3	1	1	1	2	1	2	3	3	3
CO6	3	3	3	2	3	1	1	1	2	1	2	3	3	3

Computer Organization & Architecture Lab

Code: PCC-IT-492

Contacts

Hours: 4

Credit point: 2

Course Objective:	
1.	To provide practical understanding of computer organization concepts such as instruction sets, memory organization, CPU architecture, and I/O systems.
2.	To develop skills in implementing and analyzing digital logic circuits, assembly language programs, and microprocessor-based systems.
3.	To enable students to simulate and evaluate the performance of basic computer architecture components through laboratory experiments.

Pre-Requisite:	
1.	The hardware-based design has been done in 1.the Analog & Digital Electronics laboratory

Laboratory Experiments:	
1.	HDL introduction.
2.	Basic digital logic base programming with HDL
3.	8-bit Addition, Multiplication, Division
4.	8-bit Register design

5.	Memory unit design and perform memory operations.
6.	8-bit simple ALU design
7.	8-bit simple CPU design
8.	Interfacing of CPU and Memory.

Text book and Reference books:

1. Computer Organization and Architecture: Designing for Performance – William Stallings, Pearson.
2. Computer Organization and Design – David A. Patterson and John L. Hennessy, Morgan Kaufmann.

Course Outcome:	
On completion of the course students will be able to	
PCC-IT492.1	To apply concepts and methods of digital system design techniques through hands-on projects.
PCC-IT492.2	To analyze the results of logic and timing simulations and to use these simulation results to debug digital systems.
PCC-IT492.3	To learn to design combinational and sequential digital systems starting from a word description that performs a set of specified tasks and functions.
PCC-IT492.4	To develop skills, techniques and learn state-of-the art engineering tools (such as VHDL, Xilinx ISE simulator etc) to design, implement and test modern day digital systems on FPGAs.
PCC-IT492.5	To learn by using Xilinx Foundation tools and Hardware Description Language (VHDL).
PCC-IT492.6	To learn through hands-on experimentation the Xilinx tools for FPGA design as well as the basics of VHDL design and simulate digital systems.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	3	-	-	-	1	1	2	3	2	2
CO2	3	3	2	3	3	-	-	-	2	1	2	3	3	2
CO3	3	3	3	2	3	-	-	-	2	1	1	3	3	3
CO4	3	2	3	2	3	-	-	-	2	2	2	3	2	3
CO5	3	2	3	2	3	-	-	-	2	2	2	3	2	3
CO6	3	3	3	2	3	-	-	-	2	2	2	3	3	3

Algorithms Lab

Code: PCC-IT-494

Contacts

Hours: 4

Credit point: 2

Course Objective:	
1.	To provide hands-on experience in designing and implementing fundamental algorithms such as sorting, searching, recursion, and graph algorithms.
2.	To develop analytical skills to evaluate algorithm efficiency using time and space complexity analysis.
3.	To apply algorithmic problem-solving techniques to real-world computational problems through programming exercises.

Pre-Requisite:	
1.	Pre-Requisite as in PCC-IT404

Laboratory Experiments:	
Divide and Conquer:	
1	Implement Binary Search using Divide and Conquer approach Implement Merge Sort using Divide and Conquer approach
2	Implement Quick Sort using Divide and Conquer approach Find Maximum and Minimum element from an array of integer using Divide and Conquer approach
Dynamic Programming:	
3	Find the minimum number of scalar multiplications needed for chain of matrix
4	Implement Single Source Shortest Path for a graph (Dijkstra Bellman Ford Algorithm)
5	Implement all pair of Shortest paths for a graph (Floyd-Warshall Algorithm) Implement Traveling Salesman Problem
6	Implement 0/1 Knapsack problem using Dynamic Programming
Branch and Bound:	
7	Implement 15 Puzzle Problem
Backtracking:	
8	Implement 8 Queen problem
9	Graph Coloring Problem Hamiltonian Problem
Greedy method	
10	Fractional Knapsack Problem Job sequencing with deadlines
11	Minimum Cost Spanning Tree by Prim's Algorithm Minimum Cost Spanning Tree by Kruskal's Algorithm
Graph Traversal Algorithm:	
12	Implement Breadth First Search (BFS) Implement Depth First Search (DFS)

Text book and Reference books:

1. Morris Mano, Digital Logic and computer Design, PHI, 2017
2. R.P.Jain, Modern Digital Electronics, 4/e, McGraw Hill, 2008.
3. Leach & Malvino, Digital Principles & Application, 5/e, McGraw Hill
4. Floyd & Jain, Digital Fundamentals, Pearson.
5. S. Salivahanan, S. Arivazhgan, Digital Circuits and Design, 5th edition, Oxford University Press, 2018.

Course Outcome:	
On completion of the course students will be able to	
PCC-IT494.1	To analyze the complexities of various problems in different domains and to prove the correctness and analyze the running time of the basic algorithms for those classic problems in various domains.
PCC-IT494.2	To understand methods for analyzing the efficiency and correctness of algorithms (such as exchange arguments, recurrence, induction, and complexity analysis)
PCC-IT494.3	To design algorithms using the divide and conquer, dynamic programming, greedy method, Backtracking algorithms, etc that employ this strategy.

PCC-IT494.4	To compare, contrast, and choose appropriate algorithmic design techniques to present an algorithm that solves a given problem.
PCC-IT494.5	To develop the efficient algorithms for the new problem with suitable designing techniques.
PCC-IT494.6	To know the appropriate algorithmic design technique to specific problems.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	3	2	-	-	-	1	-	2	3	3	2
CO2	3	3	1	2	2	-	-	-	1	-	2	3	3	1
CO3	2	2	2	2	3	-	-	-	1	1	2	2	2	2
CO4	3	3	3	3	3	-	-	-	1	1	2	3	3	3
CO5	3	3	3	2	3	-	-	-	1	1	3	3	3	3
CO6	3	2	2	1	2	-	-	-	1	-	2	3	2	2

Operating System Lab

Code: PCC-IT-495

Contacts

Hours: 4

Credit point: 2

Course Objective:	
1.	To provide foundational knowledge of Unix/Linux operating systems, shell programming, file management, process and thread handling, and container basics.
2.	To develop the ability to design and implement core operating system algorithms including scheduling, memory management, deadlock handling, virtual memory, and disk scheduling techniques.
3.	To equip students with practical skills in process synchronization and inter-process communication using semaphores, pipes, message queues, and shared memory.

Pre-Requisite:	
1.	Basic knowledge of Operating System concepts such as processes, memory management, file systems, and scheduling.
2.	Familiarity with programming in C/C++ or any system-level programming language.

Laboratory Experiments:	
1.	Managing Unix/Linux Operating System: Basics of UNIX commands and Implementation of Shell Programming.
2.	Process: Implementation of Process and thread (Life cycle of process): (i) Process creation and Termination; (ii) Thread creation and Termination.
3.	Scheduling: Implementation of CPU Scheduling. (i) FCFS, (ii) SJF, (iii) Round Robin.
4.	Semaphore: Programming with semaphores (Producer-Consumer Problem using Semaphores and Reader Writer Problem).
5.	Deadlock: Simulate algorithm for deadlock prevention and detection, Simulate the algorithm for Deadlock avoidance and study about deadlock recovery.
6.	Memory Allocation: Simulate memory allocation methods: (i) Best Fit, (ii) Worst Fit and (iii) First Fit. Docker Container setup and Basic implementation.
7.	Virtual Memory: (i) FIFO (ii) LRU and (iii) Optimal.
8.	Disk Scheduling: Implementation of Disk Scheduling using FCFS, SCAN and C-SCAN Algorithm

9	Inter-process communication: pipes (use functions pipe, popen, pclose), named pipes (FIFOs, accessing FIFO), message passing & shared memory (IPC version V).
---	---

Text book and Reference books:

1. Operating Systems – William Stallings
2. Advanced Programming in the UNIX Environment – W. Richard Stevens (for IPC & system calls)

Course Outcome:	
On completion of the course students will be able to	
PCC-IT495.1	Describes the concept of Unix and Linux operating system and its associate's commands.
PCC-IT495.2	Explains the accessibility of files, create and change the permissions associated with files and several other tasks.
PCC-IT495.3	Develop and demonstrate shell script programming for several problems.
PCC-IT495.4	Designing the programs of process creation, replacing or duplicating a process image.
PCC-IT495.5	Experiment with the concept of mutual exclusion and formulate solutions using semaphore in modern operating system.
PCC-IT495.6	Develop programming solutions for inter process communication

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	-	-	-	-	-	-	-	-	-	2	1	-	-
CO2	2	2	3	2	1	-	-	-	-	-	2	2	2	3
CO3	3	3	3	2	1	-	-	-	-	-	3	3	3	3
CO4	3	3	3	1	1	-	-	1	1	-	3	3	3	3
CO5	2	3	3	2	1	-	-	1	1	-	2	2	3	3
CO6	3	3	3	2	2	-	-	2	1	-	2	3	3	3

Artificial Intelligence Lab

Code: PCC-IT-496

Contacts

Hours: 4

Credit point: 2

Course Objective:	
1.	To understand and implement fundamental Artificial Intelligence algorithms.
2.	To develop problem-solving skills using search techniques, machine learning, and reasoning methods.
3.	To gain hands-on experience with AI tools, programming environments, and real-world AI applications.

Pre-Requisite:	
1.	Basic knowledge of programming (preferably Python).
2.	Fundamental understanding of Data Structures and Algorithms.

Laboratory Experiments:	
1	Basic search strategies: Tic-Tac-Toe game, 8-Puzzle problem, Water Jug Problem, Crypt-arithmetic problem

2	Heuristic search: Best First Search, A* search, Genetic algorithm, Hill Climbing Search, constraint satisfaction problems
3	Adversarial search: Minimax algorithm, Alpha-Beta pruning
4	Probabilistic reasoning: Build naïve Bayes model, Implement Bayesian networks and perform inferences
5	Fuzzy Logic: Fuzzy set basic operation, Dempster-Shafer theory
6	Mini Project

Text book and Reference books:

1. Patrick Henry Winston, *Artificial Intelligence*, Addison-Wesley.
2. Prateek Joshi, *Artificial Intelligence with Python*, Packt Publishing.

Course Outcome:	
On completion of the course students will be able to	
PCC IT496.1	Students will demonstrate the ability to solve problems collaboratively.
PCC-IT496.2	Students will demonstrate knowledge of artificial intelligence concepts.
PCC-IT496.3	An understanding of fundamental concepts and methods of searching Strategy with its applications.
PCC-IT496.4	An ability to analyze and evaluate simple algorithms with the coding of Python programming language.
PCC-IT496.5	An ability to design simple algorithms for probabilistic reasoning and Fuzzy operation with Python programming language and test them with benchmark data sets.
PCC-IT496.6	Practically establish, refine and implement strategies to take the idea in to students and faculty fraternity.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3	3	1	1	2	1	3	3	3	3	3
CO2	3	3	3	3	3	1	1	1	1	2	3	3	3	3
CO3	3	3	3	2	2	2	-	1	1	2	2	3	3	3
CO4	3	3	3	3	3	2	1	2	2	2	2	3	3	3
CO5	3	3	3	3	3	2	2	2	2	2	3	3	3	3
CO6	3	3	3	3	2	2	3	3	3	3	3	3	3	3

Fifth Semester (3rd Year)

Machine Learning

Code: PCC-IT-501

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1.	Understand theoretical foundations of Machine Learning.
2.	Evaluate and optimize ML models for real-world applications.
3.	Prepare students for industry, higher studies, and AI/ML research

Pre-Requisite:	
1.	Engineering Mathematics
2.	Proficiency in Python programming

Unit	Content	Hrs.
1.	Introduction to Machine Learning: Definition and scope of ML, Concept Learning, Hypothesis, Version Space, Loss Function, Cost Function, Gradient Descent, Types of Learning, Applications of Machine Learning, AI vs ML vs Deep Learning.	4
2.	Data Preprocessing & Feature Engineering: Data Cleaning, Removed duplicates, Handling Missing Values, Handle outliers, Encoding Categorical Variables. Feature Creation, Feature Transformation, Feature Selection, Feature Extraction, Train/Test Split, Handle imbalanced datasets (SMOTE), Feature Scaling.	6
3.	Supervised Learning: Regression: Simple vs. Multiple Linear Regression, Polynomial Regression. Evaluation Metrics: MSE, RMSE, R ² Score. Classification: Logistic Regression, Naïve Bayes, K-Nearest Neighbours (KNN), Decision Trees (ID3), Support Vector Machines (SVM). Evaluation Metrics: Confusion Matrix, Precision, Recall, F1 Score, AUC-ROC Curve.	8
4.	Unsupervised Learning: Clustering: K-Means Clustering, Hierarchical Clustering, DBSCAN. Evaluation Metrics: Silhouette Score, Adjusted Rand Index (ARI). Association Rule Mining: Itemset generation, Support, Confidence, Lift, Applications in market basket analysis. Dimensionality Reduction: PCA (Principal Component Analysis).	6
5.	Reinforcement Learning & Ensemble Learning: Basics of RL framework, Markov Decision Process, Q-learning Bagging (Random Forest), Boosting (AdaBoost, Gradient Boosting, XGBoost).	6
6.	Model Optimization & ML Pipeline: Bias–Variance Tradeoff, Cross-Validation, Hyperparameter Tuning, Overfitting and Underfitting, Regularization.	4

Unit	Content	Hrs.
	Machine Learning Pipeline, Introduction to MLOps, MLOps principles and life-cycle, tools and frameworks for MLOps, real-world ML workflow integration and monitoring.	

Text book and Reference books:

1. Kevin Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Springer 2009 (freely available online).
3. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2007.
4. Rajiv Chopra, Machine Learning, Khanna Publishing House, 2018.

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT-501.1	Define the fundamental concepts, types, and mathematical foundations of Machine Learning.
PCC-IT-501.2	Apply data preprocessing and feature engineering techniques to prepare datasets for machine learning tasks.
PCC-IT-501.3	Implement and compare supervised learning algorithms for regression and classification problems.
PCC-IT-501.4	Apply unsupervised learning techniques including clustering, dimensionality reduction, and association rule mining to discover hidden patterns.
PCC-IT-501.5	Understanding of basic reinforcement learning concepts and design and implement ensemble learning methods.
PCC-IT-501.6	Evaluate and optimize end-to-end machine learning pipelines by applying MLOps methodologies for scalable and reliable deployment.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3	3	2	-	2	1	-	2	3	3	3
CO2	3	2	3	2	2	-	-	1	1	1	2	3	2	3
CO3	3	3	3	3	3	2	-	2	2	2	2	3	3	3
CO4	3	3	3	3	3	2	1	-	-	-	2	3	3	3
CO5	3	3	3	3	3	2	-	2	2	2	2	3	3	3
CO6	3	3	2	2	3	2	-	1	1	2	2	3	3	2

Computer Networks

Code: PCC-IT-502

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1	To develop an understanding of modern network architectures from design and performance perspective.
2	To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
3	To provide an opportunity to do network programming
4	To provide a WLAN measurement idea.

Pre-Requisite:	
1.	Basic knowledge of Computer Organization and Operating Systems concepts (memory, processes, I/O operations).
2.	Understanding of Data Communication fundamentals and basic programming concepts

Unit	Content	Hrs.
1.	Data communication Components: Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spread spectrum	9
2.	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, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA	8
3.	Network Layer: Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.	14
4.	Transport Layer: Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.	8
5.	Application Layer: Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography.	8

Text book and Reference books:

1. B. A. Forouzan - “Data Communications and Networking (3rd Ed.)” – TMH
2. A. S. Tanenbaum - “Computer Networks (4th Ed.)” - Pearson Education/PHI
3. W. Stallings - “Data and Computer Communications (5th Ed.)” - PHI/ Pearson Education
4. B. Sidhu, An Integrated Approach to Computer Networks, Khanna Publishing House. (AICTE Recommended)
5. Zheng & Akhtar, Network for Computer Scientists & Engineers, OUP
6. Black, Data & Computer Communication, PHI 6. Miller, data Communication & Network, Vikas

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT502.1	Understand and distinguish between different data communication components and demonstrate the ability to analyze and implement different types of network models.
PCC-IT502.2	Explain different data link layer utilities, functions, control and protocols and Describe with their uses and applications.
PCC-IT502.3	Analyze and apply Network Layer switching techniques and logical addressing mechanisms for IPv4 and IPv6.
PCC-IT502.4	Design and implement process-to-process communication solutions and apply congestion control strategies and enhance Quality of Service (QoS) to ensure reliable and efficient communication in various professional contexts.
PCC-IT502.5	Estimates the functions of Application layer and Presentation layer paradigms and Protocols.
PCC-IT502.6	Justifies network security, cryptography, data integrity working concept.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	-	-	-	-	-	-	-	3	2	1
CO2	3	3	2	1	2	-	-	-	-	-	-	3	3	2
CO3	3	3	2	2	2	-	-	-	-	-	-	3	3	2
CO4	3	3	2	2	2	-	-	1	-	-	-	3	3	2
CO5	3	2	2	-	3	-	-	1	-	-	-	3	2	2
CO6	3	3	3	2	3	-	-	1	1	-	-	3	3	3

Image Processing

Code: PEC-IT-501A

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1	To understand the fundamentals of digital image processing, including image acquisition, representation, and enhancement techniques.
2	To learn various image transformation methods such as filtering, segmentation, compression, and restoration.
3	To develop the ability to design and implement image processing algorithms using tools like MATLAB or Python for real-world applications.

Pre-Requisite:	
1	Basic knowledge of mathematics (linear algebra, matrices, calculus, and probability).
2	Understanding of programming (commonly Python, MATLAB, or C++) and data structures.

Unit	Content	Hrs.
1.	Introduction: Background, Digital Image Representation, Fundamental steps in Image Processing, Elements of Digital Image Processing - Image Acquisition, Storage, Processing, Communication, Display.	3
2.	Digital Image Formation: A Simple Image Model, Geometric Model- Basic Transformation (Translation, Scaling, Rotation), Perspective Projection, Sampling & Quantization – Uniform & Non-uniform.	4
3.	Mathematical Preliminaries: Neighbour of pixels, Connectivity, Relations, Equivalence & Transitive Closure; Distance Measures, Arithmetic/Logic Operations, Fourier Transformation, Properties of The Two-Dimensional Fourier Transform, Discrete Fourier Transform, Discrete Cosine & Sine Transform.	9
4.	Image Enhancement: Spatial Domain Method, Frequency Domain Method, Contrast Enhancement -Linear & Nonlinear Stretching, Histogram Processing; Smoothing - Image Averaging, Mean Filter, Lowpass Filtering; Image Sharpening. High- pass Filtering, High- boost Filtering, Derivative Filtering, Homomorphic Filtering; Enhancement in the frequency domain - Low pass filtering, High-pass filtering.	8
5.	Image Restoration: Degradation Model, Discrete Formulation, Algebraic Approach to Restoration - Unconstrained & Constrained; Constrained Least Square Restoration, Restoration by homo-morphic Filtering, Geometric Transformation - Spatial Transformation, Gray Level Interpolation	8
6.	Image Segmentation: Point Detection, Line Detection, Edge detection, Combined detection, Edge Linking & Boundary Detection - Local Processing, Global Processing via The Hough Transform; Thresholding -Foundation, Simple Global Thresholding, Optimal Thresholding; Region Oriented Segmentation- Basic Formulation, Region Growing by Pixel Aggregation, Region Splitting & Merging.	7

Text book and Reference books:

1. Digital Image Processing, by Rafael C. Gonzalez & Richard E. Woods Publisher: Pearson Education, 4th edition 2017, ISBN 978-1292223049
2. Digital Image Processing and Analysis, by Bhabatosh Chanda, Dwijesh Dutta Majumder, Publisher PHI, 2nd Edition, ISBN: 9788120343252
3. Fundamentals of Digital Image Processing, by Anil K. Jain, Publisher: Pearson Education, 1st Edition 2015, ISBN: 9789332551916
4. Digital Image Processing, by Jayaraman S, Esakkirajan, Veerakumar T, Publisher Mc GRAW HILL, 2nd Edition 2020, ISBN: 978-9389811926

Course Outcome:
On completion of the course students will-be able to

PEC-IT501A.1	Understand the fundamental concepts of digital image processing, including image representation, acquisition, storage, processing, and display systems.
PEC-IT501A.2	Explain and apply digital image formation models, geometric transformations, and sampling and quantization techniques for image analysis.
PEC-IT501A.3	Apply mathematical preliminaries such as pixel relationships, distance measures, arithmetic and logical operations, and frequency-domain transforms (Fourier, DCT, DST) to image processing tasks.
PEC-IT501A.4	Implement image enhancement techniques in spatial and frequency domains for contrast improvement, noise reduction, and image sharpening.
PEC-IT501A.5	Analyze image degradation models and apply image restoration techniques, including constrained and unconstrained restoration and homomorphic filtering.
PEC-IT501A.6	Apply image segmentation techniques such as edge detection, thresholding, and region-based methods to extract meaningful information from digital images.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	-	-	-	-	-	1	2	1	1
CO2	3	3	2	1	2	-	-	-	-	-	1	3	2	1
CO3	3	3	2	2	2	-	-	-	-	-	1	3	2	1
CO4	2	3	3	2	3	-	-	-	-	-	2	2	3	2
CO5	2	3	3	3	2	-	-	-	-	-	2	3	2	2
CO6	2	3	3	2	3	-	-	-	-	-	2	3	3	2

Pattern Recognition

Code: PEC-IT-501B

Contacts Hours: 3

Credit point: 3

Course Objective:	
1.	To understand the fundamental concepts and principles of pattern recognition and statistical decision theory.
2.	To study feature extraction, feature selection, dimensionality reduction, and classification techniques.
3.	To develop the ability to apply pattern recognition methods to real-world problems involving text, images, and signals.
Pre-Requisite:	
1.	Basic knowledge of Probability and Statistics.
2.	Fundamental understanding of Linear Algebra, Machine Learning, and Python programming.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Introduction to Pattern Recognition: Definition, scope and applications; pattern recognition systems; pattern representation; supervised and unsupervised recognition; statistical and structural approaches; Bayes decision theory; discriminant functions; decision boundaries; minimum error-rate classification.	8
2.	Feature Extraction and Selection: Feature vectors; distance measures; similarity measures; normalization; correlation; principal component analysis (PCA); linear discriminant analysis (LDA); feature selection methods; dimensionality reduction; transformation of feature spaces.	7
3.	Statistical Pattern Recognition: Probability density estimation; parametric and non-parametric estimation; maximum likelihood estimation; Bayesian classification; nearest-neighbour classification; minimum distance classifiers; Gaussian classifiers; decision trees and ensemble classifiers.	8
4.	Clustering and Structural Methods: Similarity-based clustering; K-means and hierarchical clustering; mixture models; expectation-maximization; graph-based recognition; syntactic and structural pattern recognition; grammar-based approaches; template matching.	7
5.	Neural and Modern Pattern Recognition: Perceptron and multilayer neural networks; support vector machines; kernel methods; ensemble learning; convolutional neural networks for recognition; performance evaluation using	6

Unit	Content	Hrs.
	confusion matrix, precision, recall, F1-score and ROC; applications in OCR, biometrics, medical diagnosis, speech and image recognition.	

Textbook and Reference books:

1. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
2. Richard O. Duda, Peter E. Hart and David G. Stork, Pattern Classification, 2nd Edition, Wiley-Interscience, 2000.
3. Robert P. W. Duin and Elzbieta Pekalska, Pattern Recognition: Introduction and Applications, Springer.
4. Sergios Theodoridis and Konstantinos Koutroumbas, Pattern Recognition, Academic Press.
5. Andrew Webb and Keith Copsey, Statistical Pattern Recognition, Wiley.

Course Outcome:														
On completion of the course students will-be able to														
PEC-IT-501B.1	Explain the basic concepts, approaches, and mathematical foundations of pattern recognition.													
PEC-IT-501B.2	Apply feature extraction, feature selection, and dimensionality reduction techniques to pattern recognition problems.													
PEC-IT-501B.3	Analyze and implement statistical classification methods for supervised recognition tasks.													
PEC-IT-501B.4	Apply clustering and structural pattern recognition methods to identify meaningful groups and patterns.													
PEC-IT-501B.5	Evaluate pattern recognition models using suitable performance measures and validation techniques.													
PEC-IT-501B.6	Design pattern recognition solutions for practical applications involving image, text, speech, or biomedical data.													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	-	-	-	-	-	1	2	1	1
CO2	3	3	2	1	2	-	-	-	-	-	1	3	2	1
CO3	3	3	2	2	2	-	-	-	-	-	1	3	2	1
CO4	2	3	3	2	3	-	-	-	-	-	2	2	3	2
CO5	2	3	3	3	2	-	-	-	-	-	2	3	2	2
CO6	2	3	3	2	3	-	-	-	-	-	2	3	3	2

Microprocessor & Microcontroller

Code: PEC-IT-501C

Contacts Hours: 3

Credit point: 3

Course Objective:	
1.	To understand the architecture, instruction set, programming model, and interfacing of microprocessors and microcontrollers.
2.	To develop skills in assembly language and embedded C programming for microcontroller-based systems.
3.	To analyze and design microprocessor/microcontroller-based solutions using memory, I/O, timers, interrupts, and communication interfaces.
Pre-Requisite:	
1.	Digital Electronics and basic Computer Organization.
2.	Basic knowledge of programming concepts and C language.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Introduction to Microprocessors: Evolution of microprocessors; 8-bit and 16-bit microprocessors; 8085 architecture; registers, ALU, buses and flags; pin configuration; memory organization; instruction cycle; addressing modes; instruction set and assembly language programming.	7
2.	8086 Microprocessor: Architecture and programming model; register organization; memory segmentation; addressing modes; instruction set; assembly language programming; procedures and macros; interrupts; minimum and maximum mode; basic memory and I/O interfacing.	7
3.	Microcontrollers: Introduction and applications; 8051 architecture; memory organization; SFRs; I/O ports; timers/counters; serial communication; interrupt structure; addressing modes; instruction set; assembly language programming; interfacing LEDs, switches, displays and keyboards.	8

Unit	Content	Hrs.
4.	Interfacing and Peripheral Devices: ADC and DAC; programmable peripheral interface; keyboard and display interfacing; stepper motor and DC motor control; sensor interfacing; memory interfacing; serial interfaces UART, SPI and I2C; basic communication protocols.	8
5.	Embedded Applications: Embedded C programming; real-time programming concepts; watchdog timers; low-power operation; PWM; interrupt-driven programming; microcontroller-based data acquisition and control; case studies in automation, IoT and consumer electronics.	6

Textbook and Reference books:

1. Ramesh S. Gaonkar, Microprocessor Architecture, Programming and Applications with the 8085, Penram International.
2. Yu-Cheng Liu and Glenn A. Gibson, Microcomputer Systems: The 8086/8088 Family, Prentice Hall.
3. Kenneth J. Ayala, The 8051 Microcontroller, Cengage Learning.
4. Muhammad Ali Mazidi, Janice Gillispie Mazidi and Rolin D. McKinlay, The 8051 Microcontroller and Embedded Systems, Pearson.
5. Krishna Kant, Microprocessors and Microcontrollers, PHI Learning.

Course Outcome:														
On completion of the course students will-be able to														
PEC-IT-501C.1	Explain the architecture, programming model, and operation of representative microprocessors and microcontrollers.													
PEC-IT-501C.2	Analyze instruction sets, addressing modes, memory organization, and timing characteristics of microprocessor systems.													
PEC-IT-501C.3	Develop assembly language and embedded C programs for microprocessor and microcontroller applications.													
PEC-IT-501C.4	Design memory, I/O, timer, interrupt, and peripheral interfacing for embedded systems.													
PEC-IT-501C.5	Analyze serial communication and sensor/actuator interfacing techniques used in microcontroller-based systems.													
PEC-IT-501C.6	Design and evaluate microprocessor/microcontroller-based solutions for practical automation and embedded applications.													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	-	1	-	3	2	1
CO2	3	3	2	2	2	-	-	-	1	1	-	3	3	2
CO3	3	3	2	2	3	-	-	-	1	1	-	3	3	2
CO4	3	3	3	2	3	-	-	-	2	1	-	3	3	2
CO5	2	3	2	2	2	1	-	-	1	1	-	2	3	2
CO6	3	3	3	2	3	1	1	1	2	2	1	3	3	3

Micro-electronics and VLSI Design

Code: PEC-IT-502B

Contacts Hours: 3

Credit point: 3

Course Objective:	
1.	To understand semiconductor devices, digital integrated circuits, and the principles of VLSI system design.
2.	To study CMOS circuit design, fabrication technology, layout, delay, power dissipation, and reliability issues.
3.	To develop the ability to analyze and design basic VLSI building blocks and digital systems.
Pre-Requisite:	
1.	Basic knowledge of Semiconductor Devices and Digital Electronics.
2.	Fundamental understanding of Boolean Algebra, logic gates, and circuit analysis.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Introduction to VLSI: VLSI design hierarchy and design flow; Moore's and Mealy's concepts; levels of abstraction; semiconductor materials; PN junction and MOS structures; MOS capacitor; threshold voltage; CMOS technology overview; scaling trends and challenges.	8

Unit	Content	Hrs.
2.	MOS and CMOS Circuit Design: MOSFET operation; CMOS inverter; voltage transfer characteristics; noise margins; static and dynamic power dissipation; propagation delay; fan-in and fan-out; CMOS combinational logic; transmission gates; pass-transistor logic; tri-state circuits.	7
3.	Digital VLSI Building Blocks: CMOS realization of NAND, NOR, XOR and complex gates; multiplexers, decoders and adders; arithmetic circuits; latches and flip-flops; sequential circuits; memory cells; SRAM and DRAM basics; clocking and clock distribution.	8
4.	VLSI Fabrication and Layout: IC fabrication steps; oxidation, diffusion, ion implantation, deposition and lithography; well formation; metallization; CMOS process technologies; design rules; stick diagrams; layout techniques; lambda-based design; layout verification and parasitics.	6
5.	VLSI Design Methodologies: FPGA and ASIC concepts; standard-cell design; PLDs; hardware description languages; synthesis; place and route; timing analysis; low-power VLSI; testing and fault models; design-for-testability; reliability and emerging VLSI technologies.	7

Textbook and Reference books:

1. Neil H. E. Weste and David Money Harris, CMOS VLSI Design: A Circuits and Systems Perspective, Pearson.
2. Jan M. Rabaey, Anantha Chandrakasan and Borivoje Nikolic, Digital Integrated Circuits: A Design Perspective, Pearson.
3. John P. Uyemura, CMOS Logic Circuit Design, Springer.
4. John E. Ayers, Digital Integrated Circuits: Analysis and Design, CRC Press.
5. Wayne Wolf, Modern VLSI Design: System-on-Chip Design, Pearson.

Course Outcome:														
On completion of the course students will-be able to														
PEC-IT-502B.1	Explain semiconductor, MOS, CMOS, and VLSI concepts and the overall IC design flow.													
PEC-IT-502B.2	Analyze CMOS inverter characteristics, power dissipation, noise margins, and propagation delay.													
PEC-IT-502B.3	Design basic CMOS combinational and sequential building blocks used in VLSI systems.													
PEC-IT-502B.4	Explain VLSI fabrication processes, design rules, layout techniques, and parasitic effects.													
PEC-IT-502B.5	Analyze ASIC, FPGA, synthesis, timing, testing, and low-power design methodologies.													
PEC-IT-502B.6	Design and evaluate VLSI building blocks considering performance, power, area, and reliability constraints.													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	-	-	1	-	-	2	3	2	2
CO2	3	3	3	1	2	-	-	1	-	-	2	3	3	3
CO3	3	3	3	2	3	-	-	1	-	-	2	3	3	3
CO4	3	3	3	2	3	-	-	1	-	-	2	3	3	3
CO5	3	3	3	2	3	-	-	1	-	-	2	3	3	3
CO6	3	3	3	2	3	-	-	1	-	-	2	3	3	3

Bioinformatics

Code: PEC-IT-502C

Contacts Hours: 3

Credit point: 3

Course Objective:	
1.	To introduce computational methods used for analysis and interpretation of biological data.
2.	To study sequence alignment, biological databases, genome analysis, and phylogenetic methods.
3.	To develop the ability to apply bioinformatics tools and algorithms to biological and biomedical problems.
Pre-Requisite:	
1.	Basic knowledge of Biology and Genetics.
2.	Basic understanding of Probability, Statistics, algorithms, and programming.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Introduction to Bioinformatics: Scope and history; biological information and computational biology; molecular biology basics; DNA, RNA and proteins; central dogma; genes and genomes; biological data types; primary and secondary databases; GenBank, EMBL, DDBJ, PDB and UniProt; data retrieval and formats.	8
2.	Sequence Analysis: Sequence similarity and homology; pairwise alignment; global and local alignment; dynamic programming; Needleman-Wunsch and Smith-Waterman algorithms; scoring matrices; BLAST and FASTA; multiple sequence alignment; sequence identity and similarity.	7
3.	Genome and Protein Analysis: Genome organization; gene prediction; ORFs; promoter and regulatory region analysis; protein sequence analysis; motifs and domains; protein structure basics; secondary and tertiary structure prediction; molecular visualization; structure databases and applications.	7
4.	Phylogenetics and Systems Analysis: Molecular evolution; distance methods; character-based methods; phylogenetic trees; UPGMA and Neighbor-Joining; maximum parsimony; bootstrap analysis; comparative genomics; transcriptomics and proteomics; introduction to biological network analysis.	6
5.	Computational Tools and Applications: Bioinformatics workflows; Python-based sequence analysis; data preprocessing and visualization; molecular docking overview; drug discovery applications; disease and biomarker analysis; personalized medicine; machine learning in bioinformatics; ethical, privacy, and reproducibility issues in biological data analysis.	8

Textbook and Reference books:

1. Arthur M. Lesk, Introduction to Bioinformatics, Oxford University Press.
2. David W. Mount, Bioinformatics: Sequence and Genome Analysis, Cold Spring Harbor Laboratory Press.
3. Durbin, Eddy, Krogh and Mitchison, Biological Sequence Analysis, Cambridge University Press.
4. Andreas D. Baxevanis and B. F. Francis Ouellette, Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, Wiley.
5. Richard Durbin, Sean R. Eddy, Anders Krogh and Graeme Mitchison, Biological Sequence Analysis, Cambridge University Press.

Course Outcome:														
On completion of the course students will-be able to														
PEC-IT-502C.1	Explain the biological foundations, data types, databases, and major applications of bioinformatics.													
PEC-IT-502C.2	Apply sequence retrieval, pairwise alignment, scoring, and database-search techniques to biological sequences.													
PEC-IT-502C.3	Analyze genome and protein sequence features using suitable computational tools and algorithms.													
PEC-IT-502C.4	Construct and interpret phylogenetic relationships using standard computational methods.													
PEC-IT-502C.5	Apply computational and machine learning approaches to genomics, proteomics, and biomedical data.													
PEC-IT-502C.6	Design reproducible bioinformatics workflows for practical biological or biomedical problems while considering data ethics and privacy.													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	0	0	0	0	0	1	2	1	1
CO2	3	3	2	1	2	0	0	0	0	0	1	3	2	1
CO3	3	3	2	2	2	0	0	0	0	0	1	3	2	1
CO4	2	3	3	2	3	0	0	0	0	0	2	2	3	2
CO5	2	3	3	3	2	0	0	0	0	0	2	3	2	2
CO6	2	3	3	2	3	0	0	0	0	0	2	3	3	2

Mobile App Development

Code: PEC-IT-593B

Contacts Hours: 4P

Credit point: 2

Course Objective:	
1.	To understand the principles, architecture, tools, and development life cycle of mobile applications.
2.	To develop interactive Android applications using modern programming and user-interface techniques.
3.	To integrate persistent storage, networking, APIs, multimedia, and cloud services into mobile applications.

Pre-Requisite:	
1.	Basic programming knowledge in Java/Kotlin or an equivalent object-oriented language.
2.	Fundamental knowledge of databases, computer networks, and software development.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Mobile Development Environment: Mobile computing concepts; mobile platforms; Android architecture; Android Studio; project structure; Gradle; emulator and device configuration; Kotlin fundamentals; activities, lifecycle, intents and basic application components.	4
2.	User Interface and Interaction: Layouts and views; Material Design principles; widgets; menus; dialogs; fragments; RecyclerView; adapters; resources; themes and styles; responsive layouts; accessibility; touch and gesture handling.	5
3.	Data Storage and Application State: SharedPreferences/DataStore; SQLite and Room database; CRUD operations; JSON and serialization; application state management; file storage; permissions; background processing and notifications.	4
4.	Networking and Services: HTTP/HTTPS; RESTful APIs; asynchronous programming; coroutines; JSON parsing; consuming web services; authentication basics; Firebase services; cloud data and push notifications; location and map services.	5
5.	Multimedia, Security and Deployment: Camera and media APIs; image/audio/video handling; security and secure storage; common mobile vulnerabilities; testing and debugging; profiling; performance and battery optimization; signing; packaging; deployment and app-store considerations.	3

Textbook and Reference books:

1. Bill Phillips, Chris Stewart and Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch.
2. Robert C. Martin, Clean Architecture: A Craftsman's Guide to Software Structure and Design, Pearson.
3. Google Android Developers Documentation, Android Developers, developer.android.com.
4. Neil Smyth, Android Studio 4.2 Development Essentials, Payload Media.
5. Erik Hellman, Android Programming: Pushing the Limits, Wiley.

Course Outcome:														
On completion of the course students will-be able to														
PEC-IT-593B.1	Explain mobile application platforms, architecture, development environments, and application life-cycle concepts.													
PEC-IT-593B.2	Develop responsive mobile user interfaces using layouts, widgets, fragments, and event-driven programming.													
PEC-IT-593B.3	Implement persistent storage and state management using suitable local data-storage mechanisms.													
PEC-IT-593B.4	Integrate RESTful services, JSON data, authentication, notifications, and cloud services into mobile applications.													
PEC-IT-593B.5	Apply testing, debugging, security, performance, and deployment practices to mobile applications.													
PEC-IT-593B.6	Design and implement a functional mobile application that addresses a practical user or organizational requirement.													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	-	1	-	3	2	1
CO2	3	3	2	2	2	-	-	-	1	1	-	3	3	2
CO3	3	3	2	2	3	-	-	-	1	1	-	3	3	2
CO4	3	3	3	2	3	-	-	-	2	1	-	3	3	2
CO5	2	3	2	2	2	1	-	-	1	1	-	2	3	2
CO6	3	3	3	2	3	1	1	1	2	2	1	3	3	3

Introduction to Industrial Management (Humanities III)

Code: HSMC-IT-501

Contacts

Hours: 3

Credit point: 3

Course Objective:

1.	Gain a comprehensive understanding of industrial systems, management principles, and key concepts such as production planning, quality management, and supply chain operations.
2.	Develop practical skills in operations management, leadership, decision-making, and communication to effectively manage industrial processes, teams, and resources.
3.	Stay updated on emerging trends, technologies, and best practices in industrial management to adapt, innovate, and lead change initiatives in dynamic industrial environments.

Pre-Requisite:

1.	Mathematical and analytical ability
----	-------------------------------------

Unit	Content	Hrs.
1.	Introduction to Industrial Management: Industrial Management –Concept Meaning and Definitions, Scope of Industrial Management, School of management thoughts – H. Fayol & F.W. Taylor	4
2.	Production, Planning and Control: Concept and Meaning of PPC, Job shop scheduling	4
3.	Plant Location, Plant Layout and Work Study: Product design process, Process selection, Types of production system (Job, Batch and Mass Production), Concept and Meaning of Plant Location, Theories of Plant Location, Plant location-factors-Urban & Rural sites comparison, Concept and Meaning of Plant Layout, Factors Affecting Plant Layout, Advantages of Plant Layout, Techniques of Plant Layout, Types of plant layouts, Work study.	6
4.	Statistical Quality Control: Variables - attributes, Shewart control chart for variables-X chart, R chart, - Attributes-Defective-Defect Charts for attributes-p chart, c chart (simple problems).	4
5.	Material Management and Purchase Management: A) Material management-definition, functions, importance, relationship with other departments. B) Purchase - objectives, purchasing systems, purchase procedure, terms and forms used in purchase department. C) Storekeeping- functions, classification of stores as centralized and decentralized with their advantages, disadvantages and application in actual practice. D) Inventory control: i. Definition. ii. Objectives. iii. Economic Order Quantity (EOQ) and numeric examples. iv. ABC analysis and other modern methods of analysis. E) Material Requirement Planning (MRP) concept, applications and brief details about software packages available in market.	8
6.	Recent trends in Industrial Management: A) ERP (Enterprise resource planning) - concept, features and applications. B) Important features of MS Project. C) Logistics- concept, need and benefits. D) Just in Time (JIT)-concept and benefits. E) Supply chain management-concept and benefits.	4

Textbook and Reference books:

1. O. P. Khanna – “Industrial Engineering & Management”, Dhanpat Rai Publications.
2. S.C. Sharma - “Engineering Management – Industrial Engineering & Management”, Khanna Book Publishing Company, New Delhi
3. A.P. Verma and K.L. Maheshwari – “Industrial Engineering and Management", Sultan Chand & Sons
4. N. Nair – “Materials Management”, Tata McGraw Hill Education

Course Outcome:

On completion of the course students will-be able to

CO 1	Understand and recall the fundamental concepts, principles, and theories of industrial management.
CO 2	Apply industrial management tools and techniques using software and quality management strategies.
CO 3	Evaluate benefits and weaknesses of management strategies and assess market trends.
CO 4	Assess whether strategies are effective in achieving organizational goals.
CO 5	Create innovative solutions to face market challenges.
CO 6	Develop knowledge for real-time implementation and lifelong learning.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	1	2	-	-
CO2	2	3	2	2	3	-	-	-	2	2	2	3	2	-
CO3	2	3	2	2	-	2	1	1	2	2	2	2	2	-
CO4	2	3	2	3	-	2	1	1	2	2	2	2	2	-
CO5	2	2	3	2	2	1	-	1	3	3	2	2	3	2
CO6	1	2	2	2	2	2	2	2	2	3	3	2	2	3

Compiler Design

Code: PEC-IT-502A

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1	To understand and list the different stages in the process of compilation.
2	Identify different methods of lexical analysis
3	Design top-down and bottom-up parsers
4	Identify synthesized and inherited attributes
5	Develop syntax directed translation schemes
6	Develop algorithms to generate code for a target machine

Pre-Requisite:	
1	Strong knowledge of Data Structures (trees, stacks, graphs, hashing) and algorithms.
2	Understanding of Formal Languages & Automata Theory (regular expressions, finite automata, context-free grammars).

Unit	Content	Hrs.
1	Introduction to Compiling [3L]: Compilers, Analysis of the source program, The phases of the compiler, Cousins of the compiler.	3
2	Lexical Analysis [6L]: The role of the lexical analyzer, Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of a tokens, Finite automata, From a regular expression to an NFA, From a regular expression to DFA, Design of a lexical analyzer generator (Lex). **Demonstration of LEX Analysis during classes.	6
3	Syntax Analysis [9L]: The role of a parser, Context free grammars, writing a grammar, Top- down Parsing, Non- recursive Predictive parsing (LL), Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR), Parser generators (YACC). Error Recovery Strategies for different parsing techniques. ** Demonstration of YACC during classes	9
4	Syntax directed translation [5L]: Syntax director definitions, Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes.	5
5	Type checking [4L]: Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions	4
6	Run time environments [5L]: Source language issues (Activation trees, Control stack, scope of declaration, Binding of names), Storage organization (Subdivision of run-time memory, Activation records), Storage allocation strategies, Parameter passing (call by value, call by reference, copy restore, call by name), Symbol tables, dynamic storage allocation techniques.	5
7	Intermediate code generation [4L]: Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).	4

Unit	Content	Hrs.
8	Code optimization [5L]: Introduction, Basic blocks & flow graphs, Transformation of basic blocks, Dag representation of basic blocks, The principle sources of optimization, Loops in flow graph, Peephole optimization.	5
9	Code generations [4L]: Issues in the design of code generator, a simple code generator, Register allocation & assignment.	4

Text book and Reference books:

1. Aho, Sethi, Ullman - "Compiler Principles, Techniques and Tools" - Pearson Education.
2. Holub - "Compiler Design in C" - PHI.

Course Outcome:	
On completion of the course students will-be able to	
PEC-IT502A.1	Describe the different phases of compiler and identify different possible errors detected by different phases.
PEC-IT502A.2	Explain the mean of token and distinguish between the NFA and DFA. Recommend a DFA to recognize partial keywords of programming language.
PEC-IT502A.3	Demonstrate the role of a parser. Understand the top-down and bottom-up parsing techniques and its design issues.
PEC-IT502A.4	Differentiate the role of semantic analysis phase to syntax analysis phase. Be aware of how data type issues are handled in semantic analysis phase.
PEC-IT502A.5	Constructs the significance of intermediate code generation phase. Revise the different ways of intermediate code generation techniques and run-time environment issues in compilation
PEC-IT502A.6	Discriminates the knowledge of code optimization and code generation issues. Make DAG representation of basic blocks and flow graphs.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	2	-	-	-	1	-	-	3	2	-
CO2	3	3	-	-	3	-	-	-	1	-	-	3	3	-
CO3	3	3	2	2	2	-	-	-	1	-	-	3	3	2
CO4	3	3	-	2	2	-	-	-	-	-	-	3	3	-
CO5	3	2	2	3	3	-	-	-	-	-	2	3	2	2
CO6	3	2	2	3	3	-	-	-	-	-	2	3	2	2

Constitution of India

Code:MC-IT-501

Contacts

Hours: 1

Credit point: 0

Course Objective:	
1	To understand the historical background and salient features of the Constitution of India.
2	To study Fundamental Rights, Duties, Directive Principles, and the structure of government.
3	To create awareness about constitutional values, citizenship responsibilities, and democratic governance in India.

Pre-Requisite:

1	Basic understanding of Indian history, especially the freedom movement and framing of the Constitution.
2	General awareness of political science concepts like democracy, rights, duties, and governance.

Unit	Content	Hrs.
1	Introduction: Constitution' meaning of the term, Indian Constitution: Sources and constitutional history, Features: Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy	3
2	Union Government and its Administration: Structure of the Indian Union: Federalism, Centre- State relationship, President: Role, power and position, PM and Council of ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha	6
3.	State Government and its Administration Governor: Role and Position, CM and Council of ministers, State Secretariat: Organization, Structure and Functions	6
4.	Local Administration District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation, Pachayat raj: Introduction, PRI: Zila Pachayat, Elected officials and their roles, CEO Zila Pachayat: Position and role, Block level: Organizational Hierarchy (Different 4. departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy	8
5.	Election Commission Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women	6

Text book and Reference books:

1. 'Indian Polity' by Laxmikanth
2. 'Indian Administration' by Subhash Kashyap
3. 'Indian Constitution' by D.D. Basu
4. 'Indian Administration' by Avasti and Avasti

Course Outcome:	
On completion of the course students will-be able to	
MC-IT501.1	Understand the basic structure of Indian Constitution, remember and apply Preamble, Fundamental Rights and Duties
MC-IT501.2	Understand the structure and functioning of Union Government and its administration.
MC-IT501.3	Understand the structure and functioning of State Government and its administration.
MC-IT501.4	Understand and differentiate aspect of Local Administration and its related bodies.
MC-IT501.5	Understand and apply the knowledge of the constitutional institutions like Election Commission.
MC-IT501.6	

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	-	-	-	2	2	3	-	2	-	-	-	-
CO2	2	3	-	2	-	2	2	3	1	2	-	-	-	-
CO3	2	3	-	2	-	2	2	3	1	2	-	-	-	-
CO4	2	3	1	2	-	3	3	3	2	2	-	-	-	-
CO5	2	2	-	2	-	2	2	3	2	2	-	-	-	-
CO6	1	2	2	2	-	3	3	3	2	3	-	-	-	-

Machine Learning Lab

Code: PCC-IT-591

Contacts

Hours: 4

Credit point: 2

Course Objective:	
1.	Understand core concepts of machine learning
2.	Implement machine learning algorithms using Python

Pre-Requisite:	
1.	Machine learning algorithms
2.	Python programming

Laboratory Experiments:	
1.	Introduction to Python scientific computing with implementation of array manipulation and matrix operations using NumPy, Data handling and visualization using Pandas, Matplotlib, and Seaborn libraries.
2.	Implementation of data preprocessing techniques including handling missing data, encoding categorical variables, feature scaling, and train-test split.
3.	Development and evaluation of regression models: Simple Linear Regression, Multiple Linear Regression, and Polynomial Regression with performance metrics (MSE, RMSE, R ²).
4.	Implementation of classification algorithms: Logistic Regression, K-Nearest Neighbours (KNN), Naïve Bayes, Decision Tree, Random Forest and Support Vector Machine (SVM) with comparative performance analysis.
5.	Implementation of clustering techniques: K-Means (with Elbow method), Hierarchical Clustering and DBSCAN; application of Principal Component Analysis (PCA) for dimensionality reduction.
6.	Model validation using Cross-Validation techniques and hyperparameter tuning using Grid SearchCV.
7.	Mini case study on a real-world dataset integrating preprocessing, model building, evaluation, and interpretation of results.
8.	Implement the Q-Learning algorithm and train an agent to learn the optimal path in a Grid World environment using reward-based learning.

Text book and Reference books:

1. Sebastian Raschka and Vahid Mirjalili, *Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow*, Packt Publishing.
2. Andreas C. Müller and Sarah Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly Media.

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT-591.1	Implement Python-based scientific computing tools (Num Py, Pandas, Matplotlib, Sea born) for data manipulation and visualization.
PCC-IT-591.2	Apply data preprocessing techniques including data cleaning, encoding, feature scaling, and dataset splitting for machine learning tasks.
PCC-IT-591.3	Develop and evaluate regression and classification models using appropriate performance metrics.
PCC-IT-591.4	Implement clustering and dimensionality reduction techniques for unsupervised learning problems.

PCC-IT-591.5	Perform model validation and hyperparameter tuning using cross-validation techniques to improve model performance.
PCC-IT-591.6	Design and implement a complete machine learning solution for a real-world problem, including reinforcement learning concepts such as Q-Learning.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	3	1	1	2	2	1	2	3	2	2
CO2	3	3	2	3	3	1	1	2	2	1	2	3	3	2
CO3	3	3	2	3	3	1	1	2	2	1	2	3	3	2
CO4	3	3	2	3	3	1	1	2	2	1	2	3	3	2
CO5	3	3	2	3	3	1	1	2	2	1	2	3	3	2
CO6	3	3	3	3	3	2	2	3	3	3	2	3	3	3

Computer Networks Lab

Code: PCC-IT-592

Contacts

Hours: 4

Credit point: 2

Course Objective:	
1.	To provide practical exposure to networking concepts through implementation and configuration of network protocols and services.
2.	To develop skills in socket programming and simulation tools for analyzing network behavior and performance.
3.	To enable students to design, configure, and troubleshoot small-scale computer networks.

Pre-Requisite:	
1.	Basic knowledge of Computer Networks concepts such as OSI model, TCP/IP, routing, and switching.
2.	Basic programming knowledge (C/C++/Java/Python) and familiarity with Operating Systems fundamentals.

Laboratory Experiments:	
1.	NIC Installation & Configuration (Windows/Linux)
2.	Understanding IP address, subnet etc. Familiarization with - Networking cables (CAT5, UTP) - Connectors (RJ45, T-connector) - Hubs, Switches
3.	TCP/UDP Socket Programming - Simple, TCP based, UDP based - Multicast & Broadcast Sockets Implementation of a Prototype Multithreaded Server
4.	Implementation of Data Link Layer Flow Control Mechanism (Stop & Wait, Sliding Window) Data Link Layer Error Detection Mechanism (Cyclic Redundancy Check) Data Link Layer Error Control Mechanism (Selective Repeat, Go Back N)
5.	Server Setup/Configuration FTP, Tel Net, NFS, DNS, Firewall

Textbook and Reference books:

- Behrouz A. Forouzan, *Data Communications and Networking*, McGraw-Hill Education.

2. James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach*, Pearson Education.

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT592.1	Defines hardware types related to computer network.
PCC-IT592.2	Explains and evaluate network-based commands and demonstrate their use.
PCC-IT592.3	Demonstrate the performance of network protocol such as message queue, IPC.
PCC-IT592.4	Implement the socket programming for client server architecture.
PCC-IT592.5	Design and compares flow control mechanism stop and wait, Go back N etc.
PCC-IT592.6	Design and Understand Setup /Configuration for several protocols.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	3	3	2	1	1	2	1	1	3	3	2
CO2	3	3	2	2	3	2	1	1	2	-	1	3	3	2
CO3	3	3	2	2	3	2	1	1	2	-	1	3	3	2
CO4	3	3	3	3	2	2	1	1	2	-	1	3	3	3
CO5	3	3	3	3	2	2	1	1	2	1	1	3	3	3
CO6	3	3	3	3	2	2	1	1	1	1	1	3	3	3

Web Technology Lab

Code: PEC-IT-593A

Contacts

Hours: 4

Credit point: 2

Course Objective:	
1	To provide hands-on experience in designing and developing dynamic web applications using HTML, CSS, JavaScript, and server-side technologies.
2	To develop skills in implementing client-server communication, form validations, and database connectivity for web applications.
3	To enable students to deploy and test web applications and troubleshoot common issues in web development.

Pre-Requisite:	
1	Basic knowledge of Programming (C/C++/Java/Python) and problem-solving concepts.
2	Understanding of fundamental web concepts such as HTTP, URLs, web browsers, and basic HTML.

Laboratory Experiments:	
1.	To practice HTML fundamental constructs: Headings, Links, Images, Tables, Frames, Forms etc.
2.	To practice Cascading Style Sheets (CSS): Internal, External. Inline.
3.	To practice JavaScript: Basic Programs, Handling Events, Objects in JavaScript.

4.	To practice Servlet programming: Understanding life cycle methods, Basic servlet programming, Request Dispatcher interface.
5.	To practice JSP: JSP basic tags, Implicit objects, working with Action tags.
6.	To practice JDBC connectivity: Statement, Prepared Statement.

Textbook and Reference books:

1. Web Technologies, Black Book, Kogent Learning Solutions Inc, Dreamtech Press.
2. JDBC, Servlets, and JSP, New Edition, Santhosh Kumar K, Kogent Learning Solutions Inc, Dreamtech Press.

Course Outcome:	
On completion of the course students will-be able to	
PEC-IT593A.1	To design static web pages for a web-based application using features of HTML.
PEC-IT593A.2	To create structured, visually appealing, and user-friendly web pages with CSS.
PEC-IT593A.3	To develop dynamic web pages using JavaScript.
PEC-IT593A.4	To construct dynamic and interactive web applications using Servlet.
PEC-IT593A.5	To create Java Server Pages for designing dynamic and interactive web applications using JSP technology.
PEC-IT593A.6	To construct web applications to connect database and implement basic database operations using Servlet/JSP, JDBC.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	–	–	–	–	–	–	–	3	2	–
CO2	3	2	3	1	–	–	–	–	–	–	–	3	2	–
CO3	3	3	3	2	–	–	–	–	–	–	–	3	2	–
CO4	3	3	3	2	–	–	–	–	–	–	–	3	3	2
CO5	3	3	3	2	–	–	–	–	–	–	–	3	3	2
CO6	3	3	3	2	–	–	–	–	–	–	–	3	3	2

Sixth Semester (3rd Year)

Deep Learning

Code: PCC-IT-601

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To provide students with a strong foundation in neural networks, deep learning architectures, optimization techniques, and training methodologies.
2.	To enable students to design, implement, and evaluate deep learning models using modern frameworks and tools for real-world problem solving.
3.	To equip students with the ability to analyze, deploy, and optimize deep learning solutions for applications such as computer vision, NLP, time-series prediction, and intelligent systems.

Pre-Requisite:	
1.	Basic knowledge of linear algebra, probability, statistics, and calculus (especially derivatives and partial derivatives) to understand learning algorithms and optimization.
2.	Familiarity with Python programming, data structures, and introductory machine learning concepts such as regression, classification, and model evaluation.

Unit	Content	Hrs.
1.	Foundations of Deep Learning & Neural Networks: Biological neuron vs artificial neuron, Perceptron and limitations, Multi-Layer Perceptron (MLP), Activation functions (Sigmoid, Tanh, ReLU, Softmax), Loss functions, Forward propagation & Back propagation (chain rule intuition + derivation), Stochastic Gradient Descent, Vanishing & Exploding gradient problem, Weight initialization techniques, Regularization (L1, L2, Dropout, Batch Normalization).	8
2.	Deep Feed Forward Networks: Deep feed forward architectures, Bias-Variance tradeoff in deep learning, Over fitting and generalization, Hyperparameter tuning strategies, Model evaluation metrics, Introduction to Tensor Flow / Py Torch, GPU-based training.	7
3.	Convolutional Neural Networks (CNN): Convolution operation (mathematical intuition), Padding, Stride, Pooling layers, CNN architecture design, Classic architectures (VGG, Res Net), Transfer learning, Data augmentation, Object detection basics (YOLO / R-CNN).	6
4.	Sequential Models & Modern Architectures: Recurrent Neural Networks (RNN): LSTM & GRU, Sequence-to-Sequence models, Attention mechanism, Introduction to Transformers. <i>Pre-trained Language Models:</i> Encoder vs Decoder architecture, Bidirectional learning concept, Fine-tuning vs Feature extraction, Overview of BERT architecture, MLM (Masked Language Modeling), NSP (Next Sentence Prediction), Limitations of BERT, Applications in NLP, Time-series forecasting using deep learning, Basic introduction to Large Language Models.	9
5.	Industrial Deep Learning & Deployment: End-to-end Deep Learning pipeline, Data preprocessing for large datasets, Model versioning, Model compression (Quantization, Pruning), GAN, Deployment Techniques, Ethical AI & Bias in deep learning, Case studies (Healthcare, Finance, E-commerce, Autonomous systems).	8

Text book and Reference books:

1. Deep Learning — Good fellow, I.; Bengio, Y.; Courville, A. MIT Press, 2016.
2. Pattern Recognition and Machine Learning — Bishop, C. M. Springer, 2006.
3. Artificial Neural Networks — Yegnanarayana, B. PHI Learning Pvt. Ltd., 2009.
4. Matrix Computations — Golub, G. H.; Van Loan, C. F. JHU Press, 2013.
5. Neural Networks: A Classroom Approach — Satish Kumar. Tata McGraw-Hill Education, 2004.
6. Deep Learning — Rajiv Chopra. Khanna Publishing House, 2018.

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT 601.1	Explain the fundamental concepts of deep learning, biological vs artificial neurons, perceptron limitations, activation and loss functions, gradient descent, and regularization techniques.
PCC-IT 601.2	Implement forward and backward propagation algorithms and train neural networks using stochastic gradient descent, proper weight initialization, and regularization methods.
PCC-IT 601.3	Analyze deep feed forward architectures with respect to bias-variance tradeoff, over fitting, hyper parameter tuning, and performance metrics using frameworks such as Tensor Flow or Py Torch.
PCC-IT 601.4	Design and evaluate convolutional neural network architectures for image-based tasks using convolution operations, pooling, transfer learning, and object detection techniques.
PCC-IT 601.5	Evaluate sequential models including RNN, LSTM, GRU, attention mechanisms, transformers, and pre-trained language models for NLP and time-series applications.
PCC-IT 601.6	Develop and deploy an end-to-end deep learning solution integrating preprocessing, training, optimization, compression, deployment, and ethical AI considerations for real-world applications.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	0	0	0	0	0	2	3	1	0

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO2	3	3	2	2	3	0	0	1	0	0	2	3	3	2
CO3	3	3	2	3	3	0	0	0	1	0	2	3	3	2
CO4	3	3	3	2	3	0	0	2	1	0	2	3	3	3
CO5	3	3	2	3	3	1	1	0	0	0	3	3	3	2
CO6	3	3	3	3	3	3	3	2	2	3	3	3	3	3

Cloud and Edge Computing

Code: PCC-IT-602

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand the fundamental concepts, architecture, and service models of Cloud Computing.
2.	To explore Edge Computing paradigms and their role in IoT and real-time applications.
3.	To design and deploy scalable, secure, and efficient cloud-edge-based solutions.

Pre-Requisite:	
1.	Basic knowledge of Computer Networks.
2.	Fundamental understanding of Operating Systems and Virtualization concepts.

Unit	Content	Hrs.
1.	Introduction - Fundamentals of Cloud Computing: Concept of Cloud computing, Characteristics, Features and Applications, Cloud Architecture, Service models, Deployment models, Role of Cloud in Data Science domain.	4
2.	Cloud Computing Architecture: Cloud computing stack Service Models (XaaS): Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service(SaaS) Deployment Models: Public cloud, Private cloud, Hybrid cloud. Data Center Architecture.	10
3.	Infrastructure Services on Cloud: Introduction to Infrastructure, Virtualization, Hypervisors, Server virtualization, Storage Virtualization, Data Center Fundamentals, DB Cluster, Compute, Network and Storage Infrastructure, Database Services, Virtual Firewall and Security Groups, Load Balancing and auto-scaling.	10
4.	Cloud Infrastructure: Cloud Management: An overview of the features of network management systems and a brief introduction of related products from large cloud vendors, Monitoring of an entire cloud computing deployment stack – an overview with mention of some products, Lifecycle management of cloud services (six stages of lifecycle).	4
5.	Cloud Security: Cloud Security Risks, Trust, Operating System Security, VM Security, Security of Virtualization, Security Risks Posted by Shared Images, Security Risks Posted by Management OS, Data privacy and security Issues, Identity & Access Management, Access Control, Authentication in cloud computing, Case Study - Microsoft Azure, Amazon EC2	4
6.	Edge Computing: Need and Model, Use cases, Drivers and Barriers, Edge Platforms & Computing Latency, Edge computing hardware architectures, Edge node architecture, Edge computing frameworks, Edge-to-Cloud Communication, Use Cases and Applications	6

Text book and Reference books:

1. Cloud Computing Bible by Barrie Sosinsky, Wiley India Pvt. Ltd, 2013
2. Mastering Cloud Computing by Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, McGraw Hill Education (India) Private Limited, 2013

3. Cloud computing: A practical approach, Anthony T. Velte, Tata Mcgraw-Hill
4. Cloud Computing, Miller, Pearson
5. Building applications in cloud: Concept, Patterns and Projects, Moyer, Pearson
6. Cloud Computing – Second Edition by Dr. Kumar Saurabh, Wiley India

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT602.1	Define the fundamental concepts of cloud computing
PCC-IT602.2	Explain the cloud computing architecture and demonstrate the cloud stack and different cloud models.
PCC-IT602.3	Interpret infrastructure services and justify the role of virtualization in cloud computing.
PCC-IT602.4	Discuss the entire cloud computing deployment stack and distinguish key insights into cloud management and maintenance.
PCC-IT602.5	Evaluate cloud security principles and design strategies to identify security risks and challenges
PCC-IT602.6	Understand the concept of edge computing and investigate the relation with cloud computing.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	-	-	1	2	-	1	3	2	2
CO2	3	2	1	1	2	-	-	-	1	-	1	3	2	1
CO3	3	2	2	1	1	-	-	1	1	-	1	3	2	2
CO4	3	2	2	2	1	-	-	1	1	1	1	3	2	2
CO5	3	3	2	2	1	-	1	1	1	1	1	3	3	2
CO6	3	2	1	1	1	-	-	1	1	-	1	3	2	1

Internet of Things

Code: PCC-IT-603

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	Able to understand the application areas of IOT
2.	Able to realize the revolution of Internet in Mobile Devices, Cloud & Sensor Networks
3.	Able to understand building blocks of Internet of Things and characteristics

Pre-Requisite:	
1.	Computer Networks, Cloud Computing

Unit	Content	Hrs.
1.	Internet of Things (IoT): Vision, Definition, Conceptual Framework, Architectural view, technology behind IoT, Sources of the IoT, M2M Communication, IoT Examples. Design Principles for Connected Devices: IoT/M2M systems layers and design standardization, communication technologies, data enrichment and consolidation, ease of designing and affordability	6
2.	Hardware for IoT: Sensors, Digital sensors, actuators, radio frequency identification (RFID) technology, wireless sensor networks, participatory sensing technology. Embedded Platforms for IoT: Embedded computing basics, Overview of IOT supported Hardware platforms such as Arduino, NetArduino, Raspberry pi, Beagle Bone, Intel Galileo boards and ARM cortex.	8

Unit	Content	Hrs.
3.	Network & Communication aspects in IoT: Wireless Medium access issues, MAC protocol survey, Survey routing protocols, Sensor deployment & Node discovery, Data aggregation & dissemination	10
4.	IOT APPLICATIONS - IoT applications for industry: Future Factory Concepts, Brownfield IoT, Smart Objects, Smart Applications. Study of existing IoT platforms /middleware, IoT- A, Hydra etc.	7
5.	Challenges in IoT Design challenges: Development Challenges, Security Challenges, Other challenges IoT Applications: Smart Metering, E-health, City Automation, Automotive Applications, home automation, smart cards, communicating data with H/W units, mobiles, tablets, Designing of smart street lights in smart city.	5

Text book and Reference books:

1. Yasuura, H., Kyung, C.-M., Liu, Y., Lin, Y.-L., Smart Sensors at the IoT Frontier, Springer International Publishing
2. Kyung, C.-M., Yasuura, H., Liu, Y., Lin, Y.-L., Smart Sensors and Systems, Springer International Publishing
3. Jeeva Jose, Internet of Things, Khanna Publishing House, 2018.
4. Internet of Things, Arsheep Bahga and Vijay Madisetti

Course Outcome:	
On completion of the course students will-be able to	
PCC-IT604.1	Explain the definition and usage of the term “Internet of Things” in different contexts
PCC-IT604.2	Understand the key components that make up an IoT system
PCC-IT604.3	Differentiate between the levels of the IoT stack and be familiar with the key technologies and protocols employed at each layer of the stack
PCC-IT604.4	Apply the knowledge and skills acquired during the course to build and test a complete, working IoT system involving prototyping, programming and data analysis
PCC-IT604.5	Understand where the IoT concept fits within the broader ICT industry and possible future trends.
PCC-IT604.6	Appreciate the role of big data, cloud computing and data analytics in a typical IoT system

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	1	-	2	-	1	-	3	2	2
CO2	3	3	2	2	2	-	-	2	-	1	-	3	3	2
CO3	3	3	3	2	3	-	-	2	1	2	1	3	3	3
CO4	3	3	3	3	3	1	1	3	2	2	2	3	3	3
CO5	3	2	2	1	2	2	1	3	1	1	1	3	2	2
CO6	3	2	1	1	2	2	2	3	1	2	1	3	2	1

Cryptography and Network Security

Code: PEC-IT-601A

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To introduce the fundamental concepts of computer and network security, including security principles and attack models.
2.	To explain cryptographic concepts and techniques used for secure data communication.
3.	To develop an understanding of symmetric and asymmetric key cryptographic algorithms and their applications.

4.	To familiarize students with digital signatures, message authentication, and public key infrastructure.
5.	To enable learners to analyze internet security protocols, authentication mechanisms, and firewall-based network protection.

Pre-Requisite:

1.	Basics of Discrete Mathematics, Number Theory, Modular Arithmetic, and Probability for understanding encryption, hashing, and public-key algorithms.
2.	Knowledge of C/C++/Java or Python for implementing cryptographic algorithms, along with basic data structures like arrays, matrices, and modular arithmetic.

Unit	Content	Hrs.
1.	Attacks on Computers & Computer Security - Introduction, Need for Security, Security approaches, Principles of Security, Types of attack	5
2.	Cryptography: Concepts & Techniques- Introduction, Plaintext & Cipher text, Substitution Techniques, Transposition Techniques, Encryption & Decryption, Symmetric & Asymmetric key Cryptography, Key Range & Key Size	7
3.	Symmetric Key Algorithm - Introduction, Algorithm types & Modes, Overview of Symmetric Key Cryptography, DES(Data Encryption Standard) algorithm, IDEA(International Data Encryption Algorithm) algorithm, RC5(Rivest Cipher5) algorithm.	8
4.	Asymmetric Key Algorithm, Digital Signature and RSA - Introduction, Overview of Asymmetric key Cryptography, RSA algorithm, Symmetric & Asymmetric key Cryptography together, Digital Signature, Basic concepts of Message Digest and Hash Function (Algorithms on Message Digest and Hash function not required).	5
5.	Internet Security Protocols, User Authentication - Basic Concepts, SSL protocol, Authentication Basics, Password, Authentication Token, Certificate based Authentication, Biometric Authentication.	6
6.	Electronic Mail Security - Basics of mail security, Pretty Good Privacy, S/MIME.Firewall - Introduction, Types of firewall, Firewall Configurations, DMZ Network	7

Text book and Reference books:

1. "Cryptography and Network Security", William Stallings, 2nd Edition, Pearson Education Asia
2. "Network Security private communication in a public world", C. Kaufman, R. Perlman and M. Speciner, Pearson
3. Cryptography & Network Security: Atul Kahate, TMH.
4. "Network Security Essentials: Applications and Standards" by William Stallings, Pearson.
5. "Designing Network Security", Merike Kaeo, 2nd Edition, Pearson Books
6. "Building Internet Firewalls", Elizabeth D. Zwicky, Simon Cooper, D. Brent Chapman, 2nd Edition, Oreilly.
7. "Practical Unix & Internet Security", Simson Garfinkel, Gene Spafford, Alan Schwartz, 3rd Edition, Oreilly
8. "Cryptography and Network Security", V.K. Jain, Khanna Publishing House, 2017.

Course Outcome:	
On completion of the course students will-be able to	
PEC-IT601A.1	Explain the principles of computer security and classify different types of attacks.
PEC-IT601A.2	Analyze cryptographic techniques to distinguish their roles in securing data.
PEC-IT601A.3	Apply symmetric key algorithms like DES, IDEA, and RC5.
PEC-IT601A.4	Evaluate asymmetric cryptography, RSA, and digital signatures.
PEC-IT601A.5	Demonstrate knowledge of internet security protocols and authentication methods.
PEC-IT601A.6	Design secure mail service and networks using firewalls and DMZ.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	-	-	-	-	-	-	2	3	2	1
CO2	3	3	2	1	2	-	-	-	-	-	2	3	3	2
CO3	3	3	2	2	2	-	-	-	-	-	1	3	3	2
CO4	3	3	2	2	2	-	-	1	-	-	1	3	3	3
CO5	3	2	2	-	3	-	-	1	-	-	1	3	3	3
CO6	3	3	3	2	3	-	-	1	1	-	1	3	3	3

Ad-hoc Sensor Network

Code: PEC-IT-601B

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand the fundamentals, architecture, and characteristics of Ad-hoc and Wireless Sensor Networks (WSNs).
2.	To analyze routing protocols, MAC protocols, and energy-efficient communication techniques in ad-hoc sensor networks.
3.	To design and evaluate secure and scalable sensor network solutions for real-world applications.

Pre-Requisite:	
1.	Basic knowledge of Computer Networks.
2.	Fundamental understanding of Wireless Communication and Operating Systems.

Unit	Content	Hrs.
1.	Introduction to Ad-Hoc Networks: Characteristics of MANETs, Applications of MANETs and Challenges of MANETs. Routing in MANETs - Criteria for classification, Taxonomy of MANET routing algorithms, Topology-based routing algorithms-Proactive: DSDV; Reactive: DSR, AODV; Hybrid: ZRP; Position-based routing algorithms Location Services- DREAM, Quorum-based.	7
2.	Data Transmission: Broadcast Storm Problem, Rebroadcasting Schemes Simple-flooding, Probability-based Methods, Area-based Methods, Neighbor Knowledge-based: SBA, Multipoint Relaying, AHBP. Multicasting: Tree based: AMRIS, MAODV; Mesh-based: ODMRP, CAMP; Hybrid: AM Route, MCEDAR.	7
3.	Geocasting: Data-transmission Oriented-LBM; Route Creation Oriented Geo TORA, MGR.TCP over Ad-Hoc TCP protocol Overview: TCP Basics, TCP Header Format, Congestion Control; TCP and Manets: Effects of Partition on TCP, Impact Of Lower Layers On TCP.	7
4.	Basics of Wireless, Sensors and Lower Layer Issues: Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer, Routing Layer.	7
5.	Upper Layer Issues of WSN: Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs, Sensor Networks and mobile robots.	8

Text book and Reference books:

1. Ad-Hoc and Sensor Networks–Theory and Applications, Carlos Corderio Dharma P. Aggarwal, World Scientific Publications, March 2006, ISBN–981256–681–3.
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonid as Guibas, Elsevier Science, ISBN –978-1-55860-914-3 (Morgan Kauffman).

Course Outcome:

On completion of the course students will-be able to	
PEC-IT-601B.1	Explain the fundamentals, characteristics, challenges, and applications of MANETs and Wireless Sensor Networks.
PEC-IT-601B.2	Analyze and compare various routing protocols in MANETs including proactive, reactive, hybrid, and position-based routing algorithms.
PEC-IT-601B.3	Evaluate different broadcasting, multicasting, and geo-casting techniques used in ad-hoc networks.
PEC-IT-601B.4	Examine the impact of TCP over ad-hoc networks and analyze lower-layer issues including physical, MAC, and link layer challenges in WSNs.
PEC-IT-601B.5	Analyze upper-layer issues in Wireless Sensor Networks including transport layer support and dynamic adaptation mechanisms.
PEC-IT-601B.6	Design and assess secure, efficient, and scalable solutions for real-world ad-hoc and sensor network applications.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	2	1	1	-	-	3	2	3	2	2
CO2	3	3	3	3	2	1	1	-	-	3	2	3	3	3
CO3	3	3	2	3	2	1	1	-	-	3	2	3	3	2
CO4	3	3	2	2	2	1	1	-	-	3	2	3	3	2
CO5	3	3	3	3	2	1	1	-	-	3	2	3	3	3
CO6	3	3	3	3	2	1	1	-	-	3	2	3	3	3

Wireless Sensor Network

Code: PEC-IT-601C

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand the architecture, characteristics, and design challenges of wireless sensor networks.
2.	To analyze communication protocols, routing strategies, and energy-efficient mechanisms in WSNs.
3.	To develop the ability to design WSN-based applications for real-world scenarios.

Pre-Requisite:	
1.	Computer Networks
2.	Digital Communication Basics

Unit	Content	Hrs.
1.	Introduction to WSN: Overview, applications, design challenges, node architecture, sensing and actuation, hardware components, energy consumption model	7
2.	Communication in WSN: Wireless channel, MAC protocols (S-MAC, T-MAC), error control, data aggregation, energy-efficient communication	9
3.	Routing Protocols: Flooding, Gossiping, Hierarchical routing (LEACH), Data-centric routing, Location-based routing, QoS in WSN	8
4.	Localization & Time Synchronization: Localization techniques, GPS-free methods, time synchronization protocols	5
5.	Security in WSN: Security challenges, key management, secure routing, attacks and countermeasures	4

Unit	Content	Hrs.
6.	WSN Applications: IoT integration, environmental monitoring, smart agriculture, healthcare, industrial applications	4

Text book and Reference books:

1. Kazem Sohrawy, Daniel Minoli, Taieb Znati – Wireless Sensor Networks: Technology, Protocols, and Applications, Wiley.
2. Holger Karl, Andreas Willig – Protocols and Architectures for Wireless Sensor Networks, Wiley.
3. C. S. Raghavendra, Krishna Sivalingam, Taieb Znati – Wireless Sensor Networks, Springer.

Course Outcome:	
On completion of the course students will-be able to	
PEC-IT-601C.1	Explain WSN architecture, components, and design constraints.
PEC-IT-601C.2	Analyze MAC protocols and communication techniques for energy efficiency.
PEC-IT-601C.3	Evaluate and compare routing protocols used in WSNs.
PEC-IT-601C.4	Apply localization and synchronization techniques in WSN environments.
PEC-IT-601C.5	Identify security issues and propose solutions for secure WSN deployment.
PEC-IT-601C.6	Design WSN-based applications for real-world problems.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	-	1	-	3	2	1
CO2	3	3	2	2	2	-	-	-	-	1	-	3	3	2
CO3	3	3	2	2	2	-	-	-	-	1	-	3	3	2
CO4	2	3	2	2	2	-	-	-	-	1	-	2	3	2
CO5	2	3	2	2	2	1	-	2	-	1	-	2	3	2
CO6	3	3	3	2	3	1	1	1	1	2	1	3	3	3

Block chain Technology

Code: PEC-IT-602A

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand the fundamental concepts, architecture, and working principles of Blockchain technology.
2.	To analyze consensus mechanisms, cryptographic techniques, and smart contract frameworks.
3.	To design and evaluate secure, decentralized applications using blockchain platforms.

Pre-Requisite:	
1.	Basic knowledge of Data Structures and Cryptography.
2.	Fundamental understanding of Computer Networks and Distributed Systems.

Unit	Content	Hrs.
1.	Introduction: Introduction to Block chain, Purpose of the Block chain, Types of Block chain. Block chain Architecture, Centralized vs Decentralized System, System Networks, Cryptography, and Advanced Block chain Application.	6
2.	Basic Crypto Primitive: Properties of Hash Function, Cryptographic Hash function, hashes (names, reference and commitments), Blocks, Blocks headers, Merkel Tree, Chain forks, Asymmetric Cryptography, Digital Signature, Basic Crypto Currency, Bitcoin.	8
3.	Consensus Protocols- Proof of Work (PoW), Proof of Stack (PoS), Proof of capacity (PoC), Proof of burn (PoB), Practical Byzantine Fault Tolerance (PBFT), PoET, Scalability aspects of Block chain consensus protocols: Permissioned/Permission less Block chains and their application, Advantage and Disadvantages.	8
4.	Scripting and Implementation: Scripting in Block chain, Ethereum, Smart Contracts, Enterprise Block chain, Hyper ledger fabric, Financial Software and Systems (FSS): -Settlements, KYC, Capital Markets-Insurance Block chain in trade, Block chain In Government and Supply Chain.	8
5.	Privacy, Security issues in Blockchain: Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Block chains – such as Sybil attacks, selfish mining, 51% attacks - -advent of algorand, and Sharding based consensus algorithms to prevent these.	6

Text book and Reference books:

1. Blockchain Technology: Concepts and Applications: by Kumar Saurabh (Author), Ashutosh Saxena (Author), Wiley.
2. Blockchain Technology, by Chandramouli Subramanian (Author), Asha A George (Author), Abhilash K A (Author), Meena Karthikeyan (Author), Universities Press.
3. Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Application, by Lorne Lantz (Author), Daniel Cawrey (Author), O'Reilly Media
4. Mastering Blockchain: Inner workings of blockchain, from cryptography and decentralized identities, to DeFi, NFTs and Web3, by Imran Bashir (Author), packet.
5. Blockchain Basics: A Non-Technical Introduction in 25 Steps, Daniel Drescher (Author), Matthew Boston (Narrator), Gildan Media (Publisher).

Course Outcome:	
On completion of the course students will be able to	
PEC-IT602A.1	Understanding basic concepts of Block chain Technology along with basic crypto primitives.
PEC-IT602A.2	Demonstrate and Explain various Block chain consensus protocols
PEC-IT602A.3	Explain and Analyze different hyper ledger fabric components.
PEC-IT602A.4	Interpretation and Assessment application of Block chain Technology like Financial Software and Systems(FSS).
PEC-IT602A.5	Formulating and Interpreting Block chain services for Government services as well as for public interests.
PEC-IT602A.6	Investigate different privacy issues and Interpret security aspects related to Block chain Technology.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	1	1	-	1	1	-	-	-	-	-	2	1	1
CO2	2	3	2	1	1	1	-	-	-	-	1	2	3	2
CO3	2	1	1	-	1	1		-	-	-	1	2	1	1
CO4	3	3	3	2	2	1	1	-	-	1	2	3	3	3
CO5	3	3	3	2	3	1	2	-	1	2	2	3	3	3
CO6	3	3	3	3	3	2	2	1	1	2	3	3	3	3

Computer Vision

Code: PEC-IT-602B

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand the fundamental concepts of image processing and computer vision.
2.	To analyze feature extraction, object detection, and image segmentation techniques.
3.	To develop and implement computer vision applications using modern tools and deep learning frameworks.

Pre-Requisite:	
1.	Basic knowledge of Linear Algebra and Probability.
2.	Fundamental understanding of Python programming and Machine Learning.

Unit	Content	Hrs.
1.	Introduction: Image Processing, Computer Vision and Computer Graphics, what is Computer Vision - Low-level, Mid-level, High-level, Overview of Diverse Computer Vision Applications: Document Image, Analysis, Bio-metrics, Object Recognition, Tracking, Medical Image Analysis, Content-Based Image Retrieval, Video Data Processing, Multimedia, Virtual Reality and Augmented Reality.	6
2.	Image Formation Models: Monocular imaging system, Orthographic & Perspective Projection, Camera model and Camera calibration, Binocular imaging systems, Multiple views geometry, Structure determination, shape from shading, Photometric Stereo, Depth from Defocus, Construction of 3D model from images.	8
3.	Image Processing, Feature Extraction, and Motion Estimation: Image pre-processing, Image representations (continuous and discrete), Edge detection, Regularization theory, Optical computation, Stereo Vision, Motion estimation, Structure from motion.	7
4.	Shape Representation and Segmentation: Contour based representation, Region based representation, Deformable curves and surfaces, Snakes and active contours, Level set representations, Fourier, and wavelet descriptors, Medial representations, Multi-resolution analysis, Object recognition.	9
5.	Image Understanding and Computer Vision Applications: Pattern recognition methods, Face detection, Face recognition, 3D shape models of faces Application: Surveillance – foreground-background separation –human gait analysis Application: In-vehicle vision system: locating roadway – road markings – identifying road signs – locating pedestrians.	8

Text book and Reference books:

1. Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Pearson Education.
2. Richard Szeliski, *Computer Vision: Algorithms and Applications*, Springer.
3. Simon J. D. Prince, *Computer Vision: Models, Learning, and Inference*, Cambridge University Press.

Course Outcome:	
On completion of the course students will be able to	
PEC-IT-602B.1	Explain fundamental concepts of image processing, image formation models, and computer vision applications.
PEC-IT-602B.2	Analyze camera models, multiple view geometry, and 3D reconstruction techniques.
PEC-IT-602B.3	Apply image preprocessing, feature extraction, motion estimation, and stereo vision techniques to real-world problems.
PEC-IT-602B.4	Evaluate various shape representation and image segmentation methods for object recognition tasks.
PEC-IT-602B.5	Design computer vision systems for applications such as face recognition, surveillance, and intelligent transportation systems.
PEC-IT-602B.6	Implement and assess deep learning-based computer vision models using modern frameworks.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	2	2	1	-	-	-	2	1	3	2	2
CO2	3	3	2	3	2	1	-	-	-	2	1	3	3	2
CO3	3	3	3	3	3	1	-	-	-	2	1	3	3	3
CO4	3	3	2	3	2	1	-	-	-	2	1	3	3	2
CO5	3	3	3	3	3	1	1	-	1	3	2	3	3	3
CO6	3	3	3	3	3	1	1	-	1	3	2	3	3	3

Natural Language Processing

Code: PEC-IT-602C

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	Understand linguistic and probabilistic foundations of NLP.
2.	Develop statistical and machine learning models for text processing.
3.	Implement deep learning and transformer-based architectures, and to build scalable real-world NLP systems.

Pre-Requisite:	
1.	Data Structures and Algorithms
2.	Probability, Statistics and Linear Algebra
3.	Machine Learning with Python

Unit	Content	Hrs.
1.	Foundations of NLP and Text Processing: Introduction to Natural Language Processing, Applications of NLP, Ambiguity and Challenges in NLP, Levels of Linguistic Analysis (Morphology, Syntax, Semantics, Pragmatics), Text Preprocessing, Regular Expressions, Tokenization, Normalization, Stemming, Lemmatization, N-grams, Corpus Linguistics, Evaluation Metrics (Precision, Recall, F1-score).	3
2.	Language Modeling and Sequence Tagging: N-gram Language Models, Smoothing Techniques (Laplace, Good-Turing, Kneser-Ney), Part-of-Speech Tagging, Hidden Markov Models, Conditional Random Fields, Named Entity Recognition.	4
3.	Syntax and Semantics: Context-Free Grammars, Parsing Algorithms (Top-Down, Bottom-Up, CYK), Dependency Parsing, Transition-Based Parsing, Semantic Analysis, Word Sense Disambiguation, Distributional Semantics, Semantic Similarity, Co reference Resolution.	6
4.	Machine Learning Approaches for NLP: Vector Space Model, Bag of Words, TF-IDF, Feature Engineering, Text Classification, Naïve Bayes, Logistic Regression, Support Vector Machines, Sequence Labeling, Sentiment Analysis, Topic Modeling (Latent Dirichlet Allocation), Information Retrieval Basics.	8
5.	Deep Learning and Transformers in NLP: Word Embeddings (Word2Vec, GloVe, Fast Text), Neural Language Models, Recurrent Neural Networks, LSTM, GRU, Sequence-to-Sequence Models, Attention Mechanism, Transformer Architecture, Pre-trained Language Models (BERT, GPT, RoBERTa), Fine-tuning Techniques.	6
6.	Advanced NLP Applications: Machine Translation, Question Answering Systems, Text Summarization (Extractive and Abstractive), Dialogue Systems and Chatbots, Cross-Lingual Models, Multilingual NLP (Indian Languages), Speech-to-Text Integration, Ethical Issues, Bias and Fairness in NLP, Large Language Models and Prompt Engineering.	5

Text book and Reference books:

1. Jurafsky D. and Martin H. J, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Prentice Hall (2014), 2nd ed.
2. Manning D. C. and Schütze H., Foundations of Statistical Natural Language Processing MIT Press (1999) 1st ed.
3. Dale R., Moisl H. and Somers H., Handbook of Natural Language Processing, CRC Press (2010), 2nd ed.
4. Bird S., Klein E. and Loper E., Natural Language Processing with Python, Oreilly Publication (2009), 2nd ed.

Course Outcome:	
On completion of the course students will be able to	
PEC IT601C.1	Explain linguistic and probabilistic foundations of NLP.
PEC IT601C.2	Build statistical language models and sequence tagging systems.
PEC IT601C.3	Apply syntactic and semantic analysis techniques.
PEC IT601C.4	Implement machine learning models for text analytics.
PEC IT601C.5	Develop deep learning and transformer-based NLP models.
PEC IT601C.6	Design advanced NLP applications considering ethical aspects.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	–	–	–	–	–	–	–	2	2	–
CO2	3	3	2	2	1	–	–	–	–	–	–	3	3	–
CO3	3	3	2	2	1	–	–	–	–	–	–	3	3	–
CO4	3	3	3	2	2	–	–	–	–	–	–	3	3	–
CO5	3	3	3	2	2	–	–	–	–	–	–	3	3	2
CO6	3	3	2	2	2	–	–	–	–	–	–	3	3	2

Embedded System

Code: PEC-IT-602D

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand the architecture, design principles, and components of embedded systems.
2.	To develop skills in programming microcontrollers and interfacing peripherals.
3.	To design and analyze real-time embedded systems for practical applications.

Pre-Requisite:	
1.	Digital Electronics and Microprocessors
2.	Basic knowledge of C programming

Unit	Content	Hrs.
1.	Introduction to Embedded Systems: Definition, characteristics, classification, embedded system design flow, hardware-software co-design, applications	6
2.	Microcontroller Architecture: 8-bit/16-bit microcontrollers (8051/ARM basics), memory organization, I/O ports, timers, interrupts	8

Unit	Content	Hrs.
3.	Embedded C Programming: Data types, control structures, functions, pointers, interrupt handling, embedded C vs standard C	8
4.	Interfacing & Communication: Sensors, actuators, ADC/DAC, serial communication (UART, SPI, I2C), interfacing techniques	6
5.	Real-Time Systems: Concepts of real-time systems, scheduling algorithms, RTOS basics, task management, synchronization	6
6.	Embedded System Design & Applications: Case studies (home automation, automotive systems, IoT devices), design constraints, low power design	6

Text book and Reference books:

1. Raj Kamal – Embedded Systems: Architecture, Programming and Design, McGraw Hill.
2. Frank Vahid, Tony Givargis – Embedded System Design, Wiley.
3. Muhammad Ali Mazidi – The 8051 Microcontroller and Embedded Systems, Pearson.

Course Outcome:	
On completion of the course students will be able to	
PEC IT602D.1	Explain the architecture and design flow of embedded systems.
PEC IT602D.2	Analyze microcontroller architecture and its interfacing mechanisms.
PEC IT602D.3	Develop embedded programs using C for microcontroller-based systems.
PEC IT602D.4	Interface sensors and communication modules in embedded applications.
PEC IT602D.5	Analyze real-time constraints and apply scheduling techniques in embedded systems.
PEC IT602D.6	Design embedded system solutions for real-world applications.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	-	1	-	3	2	1
CO2	3	3	2	2	2	-	-	-	1	1	-	3	3	2
CO3	3	3	2	2	3	-	-	-	1	1	-	3	3	2
CO4	3	3	3	2	3	-	-	-	2	1	-	3	3	2
CO5	2	3	2	2	2	1	-	-	1	1	-	2	3	2
CO6	3	3	3	2	3	1	1	1	2	2	1	3	3	3

Operations Research

Code: OEC-IT-601A

Contacts Hours: 3

Credit point: 3

Course Objective:	
1.	To understand mathematical models and optimization techniques used for decision-making.
2.	To formulate and solve linear programming, transportation, assignment, and network optimization problems.
3.	To apply operations research techniques to scheduling, inventory, queuing, and project management problems.
Pre-Requisite:	
1.	Basic knowledge of Mathematics, algebra, matrices, and probability.
2.	Fundamental understanding of quantitative problem-solving and elementary programming or spreadsheet tools.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Introduction and Linear Programming: Nature, scope and applications of Operations Research; modeling process; assumptions; linear programming formulation; graphical solution; simplex method; Big-M and two-phase methods; duality and sensitivity concepts.	8
2.	Transportation and Assignment Models: Transportation problem formulation; balanced and unbalanced problems; initial basic feasible solutions; North-West Corner, Least Cost and Vogel's Approximation methods; optimality test; MODI method; assignment problem; Hungarian method; maximization and prohibited assignments.	8
3.	Network and Integer Optimization: Shortest path; minimum spanning tree; maximum flow; project network; CPM and PERT; critical path and floats; crashing; integer programming; branch-and-bound; basic goal programming concepts.	8
4.	Inventory, Queuing and Simulation: Inventory models; EOQ with and without shortages; quantity discounts; probabilistic inventory concepts; queuing systems; Kendall notation; M/M/1 and M/M/c models; waiting-time measures; simulation concepts and Monte Carlo method.	8
5.	Decision and Replacement Analysis: Decision theory under certainty, risk and uncertainty; payoff tables; expected value and expected opportunity loss; decision trees; replacement models; sequencing problems; dynamic programming basics; applications of OR using spreadsheets and optimization software.	8

Textbook and Reference books:

1. Hamdy A. Taha, Operations Research: An Introduction, Pearson.
2. Frederick S. Hillier and Gerald J. Lieberman, Introduction to Operations Research, McGraw-Hill.
3. Kanti Swarup, P. K. Gupta and Man Mohan, Operations Research, Sultan Chand & Sons.
4. R. Panneerselvam, Operations Research, PHI Learning.
5. J. K. Sharma, Operations Research: Theory and Applications, Macmillan.

Course Outcome:														
On completion of the course students will-be able to														
OEC-IT-601A.1	Explain the principles, assumptions, and modeling framework of Operations Research.													
OEC-IT-601A.2	Formulate and solve linear programming problems using graphical and simplex-based methods.													
OEC-IT-601A.3	Apply transportation, assignment, network, and integer programming techniques to optimization problems.													
OEC-IT-601A.4	Analyze inventory, queuing, sequencing, and simulation models for operational decision-making.													
OEC-IT-601A.5	Apply CPM, PERT, decision analysis, and replacement techniques to engineering and management problems.													
OEC-IT-601A.6	Evaluate alternative solutions using appropriate OR tools and communicate recommendations based on quantitative analysis.													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2	3	3	1	2	2	2	2	3	3	3
CO2	3	2	2	2	3	1	-	1	2	1	2	3	2	2
CO3	2	2	-	-	-	-	-	2	3	3	2	2	2	-
CO4	1	2	2	2	1	-	-	2	3	2	2	1	2	2
CO5	1	2	-	-	-	2	-	1	2	2	2	1	2	-
CO6	1	-	-	-	-	3	-	3	1	2	2	1	-	-

Introduction to Philosophical Thoughts

Code: OEC-IT-601B

Contacts Hours: 3

Credit point: 3

Course Objective:	
1.	To introduce students to major philosophical questions concerning knowledge, reality, ethics, society, and human existence.
2.	To develop critical, analytical, and reflective thinking through selected Indian and Western philosophical traditions.
3.	To connect philosophical ideas with professional life, technology, society, values, and responsible decision-making.
Pre-Requisite:	
1.	General awareness of society, culture, and contemporary issues.
2.	Willingness to read, discuss, compare, and critically reflect on philosophical ideas.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Nature and Scope of Philosophy: Meaning, origin and branches of philosophy; philosophical method; metaphysics, epistemology, logic, ethics, aesthetics and social philosophy; philosophy and science; philosophy and everyday life.	8
2.	Indian Philosophical Thought: Vedic and Upanishadic ideas; schools of Indian philosophy; Nyaya, Vaisheshika, Samkhya, Yoga, Mimamsa and Vedanta; Buddhist and Jain philosophies; concepts of karma, dharma, liberation and non-violence.	8
3.	Western Philosophical Thought: Socrates, Plato and Aristotle; rationalism and empiricism; Descartes, Locke, Berkeley and Hume; Kant and the critical approach; existentialist ideas; selected modern perspectives on freedom, responsibility and human existence.	8
4.	Ethics and Human Values: Nature of morality; virtue ethics; deontological and consequentialist approaches; justice, rights and duties; truth, non-violence, compassion and social responsibility; professional ethics; ethical dilemmas in science and technology.	8
5.	Philosophy, Technology and Society: Philosophy of science; technology and human values; artificial intelligence and responsibility; privacy and autonomy; environmental ethics; sustainability; social justice; pluralism and dialogue; critical reflection on contemporary technological society.	8

Textbook and Reference books:

1. S. Radhakrishnan, Indian Philosophy, Vol. I & II, Oxford University Press.
2. S. C. Chatterjee and D. M. Datta, An Introduction to Indian Philosophy, University of Calcutta.
3. Bertrand Russell, The Problems of Philosophy, Oxford University Press.
4. William Lawhead, The Philosophical Journey: An Interactive Approach, McGraw-Hill.
5. Peter Singer, Practical Ethics, Cambridge University Press.

Course Outcome:														
On completion of the course students will-be able to														
OEC-IT-601B.1	Explain the meaning, scope, branches, and methods of philosophical inquiry.													
OEC-IT-601B.2	Describe and compare major ideas from selected Indian philosophical traditions.													
OEC-IT-601B.3	Explain and compare important themes in classical and modern Western philosophical thought.													
OEC-IT-601B.4	Analyze ethical theories and apply them to personal, professional, and social situations.													
OEC-IT-601B.5	Evaluate ethical and social implications of science, technology, artificial intelligence, and environmental issues.													
OEC-IT-601B.6	Demonstrate critical, reflective, and reasoned thinking while discussing philosophical and contemporary problems.													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1	1	1	1	-	1	2	3	1	1	2	3
CO2	-	1	1	1	1	2	1	3	2	3	2	1	2	3
CO3	1	2	1	1	1	-	-	-	1	3	2	3	2	2
CO4	-	1	1	1	1	1	1	2	3	3	2	1	2	3
CO5	-	-	-	-	-	2	-	3	2	3	1	1	2	2
CO6	-	1	1	1	1	1	-	2	2	3	1	1	2	3

Soft Skill & Interpersonal Communication

Code: OEC-IT-601C

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To develop effective verbal and non-verbal communication skills for professional and personal contexts.
2.	To enhance interpersonal skills including teamwork, leadership, conflict management, and emotional intelligence.
3.	To build presentation, interview, and workplace communication competencies required for career success.

Pre-Requisite:	
1.	Basic understanding of English language communication.
2.	Willingness to participate in interactive activities, group discussions, and presentations.

Unit	Content	Hrs.
1.	Soft Skills: An Introduction – Definition and Significance of Soft Skills; Process, Importance and Measurement of Soft Skill Development. Self-Discovery: Discovering the Self; Setting Goals; Beliefs, Values, Attitude, Virtue. Positivity and Motivation: Developing Positive Thinking and Attitude; Driving out Negativity; Meaning and Theories of Motivation; Enhancing Motivation Levels.	12
2.	Interpersonal Communication: Interpersonal relations; communication models, process and barriers; team communication; developing interpersonal relationships through effective communication; listening skills; essential formal writing skills; corporate communication styles – assertion, persuasion, negotiation. Public Speaking: Skills, Methods, Strategies and Essential tips for effective public speaking. Group Discussion: Importance, Planning, Elements, Skills assessed; Effectively disagreeing, Initiating, Summarizing and Attaining the Objective. Non-Verbal Communication: Importance and Elements; Body Language. Teamwork and Leadership Skills: Concept of Teams; Building effective teams; Concept of Leadership and honing Leadership skills.	12
3.	Interview Skills: Interviewer and Interviewee – in-depth perspectives. Before, During and After the Interview. Tips for Success. Presentation Skills: Types, Content, Audience Analysis, Essential Tips – Before, During and After, Overcoming Nervousness. 3. Etiquette and Manners – Social and Business. Time Management – Concept, Essentials, Tips. Personality Development – Meaning, Nature, Features, Stages, Models; Learning Skills; Adaptability Skills.	12

Text book and Reference books:

1. Managing Soft Skills for Personality Development – edited by B.N. Ghosh, McGraw Hill India, 2012.
2. English and Soft Skills – S.P. Dhanavel, Orient Blackswan India, 2010.

Course Outcome:	
On completion of the course students will be able to	
OEC-IT601C.1	Communication: Students will maintain open, effective, and professional communications.
OEC-IT601C.2	Professionalism: Students will demonstrate appropriate workplace demeanor and behavior
OEC-IT601C.3	Problem-solving: Students will demonstrate flexibility, desire to meet challenges, and ability to find solutions.
OEC-IT601C.4	Teamwork: Students will develop and maintain constructive working relationships.
OEC-IT601C.5	Demonstration: Demonstrate acknowledgment and validation of the feelings, opinions, and contributions of others.

OE-IT601C.6	Application: Effectively apply active listening skills.
-------------	---

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1	1	1	1	-	1	2	3	1	1	2	3
CO2	-	1	1	1	1	2	1	3	2	3	2	1	2	3
CO3	1	2	1	1	1	-	-	-	1	3	2	3	2	2
CO4	-	1	1	1	1	1	1	2	3	3	2	1	2	3
CO5	-	-	-	-	-	2	-	3	2	3	1	1	2	2
CO6	-	1	1	1	1	1	-	2	2	3	1	1	2	3

Deep Learning Lab

Code: PCC-IT-691

Contacts Hours:

4

Credit point: 2

Course Objective:	
1.	Understand the concepts of deep learning
2.	Implement deep learning algorithms using Python

Pre-Requisite:	
1.	Machine learning algorithms with Python

Laboratory Experiments:	
1.	Introduction to Tensor Flow/PyTorch and GPU setup.
2.	Single-Layer Perceptron for binary classification and Multi-Layer Perceptron (MLP) using Back propagation.
3.	Regularization and Optimization techniques (L1, L2, Dropout, SGD/Adam).
4.	Image Classification using Convolutional Neural Networks (CNN).
5.	Transfer Learning using Pretrained CNN models (e.g., Res Net/VGG).
6.	Text Classification using RNN/LSTM models.
7.	Sentiment Analysis using Transformer-based models (e.g., BERT).
8.	Time-Series Forecasting using LSTM networks.
9.	Object Detection using YOLO or pretrained detection models.
10.	Mini Project using a Real-World Dataset with end-to-end deep learning pipeline.

Text book and Reference books:

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, MIT Press.
2. Francois Chollet, *Deep Learning with Python*, Manning Publications.
3. Nikhil Buduma and Nicholas Locascio, *Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms*, O'Reilly Media.

Course Outcome:	
On completion of the course students will be able to	
PCC- IT 691.1	Develop and train basic neural network models such as Perceptron and MLP.
PCC- IT 691.2	Apply optimization and regularization techniques to improve model performance.
PCC- IT 691.3	Design and evaluate CNN models for image classification tasks.
PCC- IT 691.4	Apply RNN/LSTM and Transformer models for text and sequential data analysis.
PCC- IT 691.5	Utilize transfer learning and pretrained models for real-world datasets.
PCC- IT 691.6	Build and deploy an end-to-end deep learning solution using a real-world dataset.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	–	2	–	–	–	1	–	–	1	3	2
CO2	2	3	2	2	3	–	–	–	1	–	–	1	2	3
CO3	2	3	3	2	3	–	–	–	1	–	–	1	2	3
CO4	2	3	2	2	3	–	–	–	1	–	–	1	2	3
CO5	2	2	3	2	3	–	–	–	1	–	–	1	2	3
CO6	2	3	3	2	3	2	1	2	2	1	2	2	2	3

Cloud and Edge Computing Lab

Code: PCC-IT-692

Contacts Hours:

4

Credit point: 2

Course Objective:	
1.	To develop practical proficiency in provisioning, configuring, and managing cloud infrastructure and services.
2.	To design and deploy containerized and serverless applications using modern cloud-native technologies.
3.	To implement and evaluate edge computing solutions for latency-sensitive and distributed applications.

Pre-Requisite:	
1.	Fundamentals of Computer Networks and Operating Systems.
2.	Basic knowledge of Linux, virtualization, and programming (Python/Java/C).

Laboratory Experiments:	
1.	Create and configure virtual machines on a cloud platform (AWS/Azure/GCP) and perform resource scaling.
2.	Install and configure hypervisors (VirtualBox/VMware) and analyze VM performance.
3.	Build Docker images, run containers, and manage container lifecycle.
4.	Deploy multi-container applications using Docker Compose and introduction to Kubernetes (pods, services).
5.	Implement object storage and managed database services; perform CRUD operations.

6.	Develop and deploy functions using AWS Lambda/Azure Functions and integrate with APIs.
7	Use Raspberry Pi/Edge simulators to run lightweight applications and collect sensor data.
8	Design a basic edge computing pipeline with local processing and cloud synchronization.
9	Measure latency, bandwidth usage, and response time for cloud vs edge deployments.
10	Design and implement a cloud-edge integrated system (e.g., smart monitoring system with real-time alerts).

Text book and Reference books:

1. Rajkumar Buyya et al., Cloud Computing: Principles and Paradigms, Wiley.
2. Thomas Erl et al., Cloud Computing: Concepts, Technology & Architecture, Pearson.
3. Weisong Shi et al., Edge Computing: Vision and Challenges, IEEE.

Course Outcome:	
On completion of the course students will be able to	
PCC- IT 692.1	Provision and manage cloud infrastructure including virtual machines, storage, and networking services.
PCC- IT 692.2	Build and deploy containerized applications using Docker and basic orchestration tools.
PCC- IT 692.3	Develop serverless and cloud-native applications using managed cloud services.
PCC- IT 692.4	Design and implement edge computing workflows for real-time data processing.
PCC- IT 692.5	Analyze performance trade-offs between cloud and edge computing systems using experimental metrics.
PCC- IT 692.6	Integrate cloud and edge platforms to develop scalable and latency-aware distributed applications.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	3	-	-	-	1	1	-	3	2	1
CO2	3	2	3	1	3	-	-	-	2	1	-	3	3	2
CO3	3	3	3	2	3	1	-	-	2	2	-	3	3	2
CO4	2	3	3	2	2	1	1	-	2	1	-	3	3	2
CO5	2	3	2	3	2	1	1	-	1	2	-	2	3	3
CO6	3	3	3	2	3	2	1	1	2	2	1	3	3	3

Internet of Things Lab

Code: PCC-IT-693

Contacts Hours:

4

Credit point: 2

Course Objective:	
1.	To provide hands-on experience in designing and implementing IoT-based systems using sensors, actuators, and microcontrollers.
2.	To enable students to interface hardware devices with software platforms for data acquisition, processing, and communication over the internet.
3.	To develop practical skills in IoT protocols, cloud integration, and real-time monitoring applications.

Pre-Requisite:	
1.	Fundamental understanding of electronics and microcontrollers.

2.	Basic knowledge of programming (C / Python)
----	---

Laboratory Experiments:	
1.	Basic understanding of the sensors
2.	Build a stream reasoner over HTTP/FTP/MQTT
3.	Basic circuit on Aurdino using bread board
4.	Basic circuit on Rasbery Pi using bread board
5.	Convert Time series data obtained from temperature sensor into XML
6.	Convert Time series data obtained from temperature sensor into JSON
7	Store time series data (temperature of college) to cloud
8	Design of time series analysis algorithms (3 methods)
9	Switch on LED based on temparature
10	Send SMS from R-Pi/ Aurdino based on temperature threshold

Text book and Reference books:

1. Arshdeep Bahga and Vijay Madiseti, Internet of Things: A Hands-On Approach, Universities Press, 2015
2. Matt Richardson and Shawn Wallace, Getting Started with Raspberry Pi, 3rd Edition, Maker Media, 2016.
3. Richard Blum, Arduino Programming in 24 Hours, Sams Publishing, 2014.

Course Outcome:	
On completion of the course students will be able to	
PCC- IT 693.1	Identify and interface different sensors to acquire real-time data in IoT applications.
PCC- IT 693.2	Design and implement basic electronic circuits using Arduino and Raspberry Pi with breadboard setup.
PCC- IT 693.3	Develop IoT-based applications using standard communication protocols such as HTTP, FTP, and MQTT.
PCC- IT 693.4	Analyze and convert time-series sensor data into structured formats such as XML and JSON.
PCC- IT 693.5	Implement cloud-based storage and perform basic time-series data management for IoT applications.
PCC- IT 693.6	Design and develop real-time IoT systems with automation and alert mechanisms based on threshold conditions.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	2	3	-	-	-	1	1	-	3	2	1
CO2	3	2	3	2	3	-	-	-	2	1	-	3	3	2
CO3	3	3	2	2	3	1	-	-	2	2	-	3	3	2
CO4	2	3	2	3	2	-	-	-	1	2	-	2	3	2
CO5	2	2	3	2	3	1	1	-	2	2	1	2	3	3
CO6	3	3	3	3	3	2	1	1	2	2	1	3	3	3

Seventh Semester (4th Year)

Big Data Analytics

Code: PEC- IT-701A

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1.	Students will learn to identify trends in unstructured data and apply analytics for real-world scenarios like fraud detection, credit risk management, and algorithmic trading for the diverse industries (healthcare, finance, and marketing).
2.	Students will be able to analyze the design and distribution models of non-relational databases. This includes mastering NoSQL data models (key-value pair, document, and graph).
3.	Students will gain hands-on experience with HDFS for storage, YARN for resource management, and high-level tools like Pig Latin and HiveQL for complex data manipulation and querying.

Pre-Requisite:	
1.	Basic proficiency in Java / Python.
2.	Basic knowledge of DBMS, SQL concepts (DDL, DML, queries)

Unit	Content	Hrs.
1.	Introduction to Big Data: Definition and Characteristics of Big Data, Necessity of Big Data, Convergence of Key Trends (Cloud, Social, Mobile, Analytics). Structured vs. Unstructured Data, Industry Examples of Big Data. Big Data Applications: Web Analytics, Big Data in Marketing and Advertising, Fraud Detection using Big Data, Risk Analytics and Credit Risk Management, Big Data in Algorithmic Trading, Big Data in Healthcare and Medicine. Big Data Technologies and Ecosystem: Big Data Technologies Overview, Introduction to Hadoop, Open-Source Technologies, Cloud and Big Data, Mobile Business Intelligence, Crowd-sourcing Analytics, Inter and Trans Firewall Analytics.	8
2.	NoSQL: Overview of NoSQL databases, Need for NoSQL in Big Data environments. Data Models: Aggregate Data Models, Aggregates, Key–Value Data Model, Document Data Model, Relationships in NoSQL, Graph Databases, Schemaless Databases, Materialized Views. Distribution and Replication Models: Distribution Models, Sharding, Master–Slave Replication, Peer-to-Peer Replication, Sharding and Replication Strategies. Consistency and Data Processing: Consistency Models, Relaxing Consistency (Eventual Consistency), Version Stamps, MapReduce Model, Partitioning and Combining, Composing MapReduce Calculations.	8
3.	Data Handling in Hadoop: Data Formats, Analyzing Data with Hadoop, Scaling Out in Distributed Systems, Hadoop Streaming, Hadoop Pipes. HDFS Architecture and Design: Design of Hadoop Distributed File System (HDFS), HDFS Concepts and Architecture, HDFS Java Interface, Data Flow in HDFS. Hadoop I/O and Storage: Hadoop I/O, Data Integrity Mechanisms, Compression Techniques, Serialization, Avro Data Serialization, File-Based Data Structures.	4
4.	MapReduce Fundamentals: MapReduce Workflows, Anatomy of a MapReduce Job Run, Classic. MapReduce Architecture, MapReduce Types, Input Formats, Output Formats. Execution and Processing: Job Scheduling, Shuffle and Sort Phase, Task Execution. Testing and Development: Unit Testing with MRUnit, Test Data and Local Testing. YARN and Failure Handling: YARN Architecture, Failures in Classic MapReduce, Failures in YARN.	8
5.	HBase and Cassandra: HBase Architecture, HBase Data Model and Implementation, HBase Clients, HBase Examples, Practical Applications (Praxis)	4

Unit	Content	Hrs.
	Cassandra: Cassandra Architecture, Cassandra Data Model, Cassandra Clients, Cassandra Examples, Hadoop Integration with Cassandra.	
6.	Apache Pig: Introduction to Pig, Grunt Shell, Pig Data Model, Pig Latin Programming, Developing and Testing Pig Latin Scripts. Apache Hive: Introduction to Hive, Hive Data Types and File Formats, HiveQL Data Definition Language (DDL), HiveQL Data Manipulation Language (DML), HiveQL Queries.	4

Text book and Reference books:

1. Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, Business Intelligence and Analytic
2. Trends for Today's Businesses, ISBN: 978-1-118-23915-5, Wiley, 2012.
3. Tom White, "Hadoop: The Definitive Guide", ISBN: 978-1-491-90163-2 Fourth Edition, O'Reilly, 2015.
4. Eric Sammer, "Hadoop Operations", ISBN 978-1449327057, O'Reilly, 2012.
5. Martin J. Fowler, Pramod kumar J. Sadalage, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Addison-Wesley Professional, ISBN 978-1449327057 2012.
6. Lars George, "HBase: The Definitive Guide: Random Access to Your Planet-Size Data, (Indian Edition) 1st Edition, ISBN- 10 935023503X, O'Reilly, 2011.
7. Jeff Carpenter Eben Hewitt, Cassandra – The Definitive Guide, ISBN-978-1491933664, O'Reilly, 2015.
8. Subhashini Chellappan Seema Acharya, "Big Data and Analytics" ISBN 978-8126579518, Wiley, 2019.

Course Outcome:	
On completion of the course students will be able to	
PEC-IT701A.1	explain big data concepts, characteristics, unstructured data, and applications in marketing, healthcare, finance, fraud detection, and risk analytics.
PEC-IT701A.2	analyze the Big Data ecosystem, including Hadoop.
PEC-IT701A.3	apply NoSQL data models such as key-value, document, and graph databases, and implement distribution strategies like sharding and replication.
PEC-IT701A.4	design and implement distributed data processing using MapReduce, YARN, Hadoop streaming, and related tools.
PEC-IT701A.5	develop big data solutions using HBase, Cassandra, Pig, and Hive for large-scale data storage and analytics.
PEC-IT701A.6	evaluate performance, consistency models, fault tolerance, and scalability issues in distributed Big Data systems.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	2	1	0	1	0	2	3	1	1
CO2	3	3	1	2	2	1	0	0	1	0	2	3	2	1
CO3	3	3	3	2	3	1	0	1	1	1	2	3	3	2
CO4	3	3	3	3	3	1	0	1	1	1	2	3	3	2
CO5	3	3	3	2	3	1	0	1	1	1	2	3	3	3
CO6	3	3	2	3	2	2	1	1	1	1	3	3	2	2

Cyber Physical System

Code: PEC- IT-701B

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1.	To understand the fundamentals, architecture, and design principles of Cyber Physical Systems (CPS).
2.	To analyze integration of computation, communication, and control in real-time embedded systems.
3.	To design and evaluate secure, reliable, and intelligent CPS applications in domains such as smart grids, healthcare, and autonomous systems.

Pre-Requisite:	
1.	Basic knowledge of Embedded Systems and Microcontrollers.
2.	Fundamental understanding of Computer Networks and Control Systems.

Unit	Content	Hrs.
1.	Introduction: Cyber-Physical Systems (CPS) in the real world, Basic principles of design and validation of CPS, Industry 4.0, IIOT implications, Smart Manufacturing, Smart transport, Smart City.	3
2.	CPS Platform components: CPS HW platforms, Processors, Sensors and Actuators, CPS Network - Wireless, CAN, Automotive Ethernet, Scheduling Real Time CPS tasks, Synchronous Model and Asynchronous Model, 5C Architecture, Self X Capabilities.	5
3.	Automated Control Design: Dynamical Systems and Stability, Controller Design Techniques, Stability Analysis: CLFs, MLFs, stability under slow switching, Performance under Packet drop and Noise	6
4.	Communication Protocols: Networking protocols, Communication protocols, Communication standards, Comparative analysis, Semantic ontology, Interoperability	4
5.	Implementation: Embedded System vs IoT vs CPS, features to software components, mapping software components to ECUs, Performance Analysis - effect of scheduling, bus latency, sense and actuation faults on control performance, network congestion, Control, Bus and Network Scheduling using Truetime	5
6.	Modeling and Design: Design of CPS, OPC-UA/MT Connect, Digital twins of physical system, Computational system, Tools and techniques, Design & Fundamental requirements.	5
7.	Security of Cyber: Physical Systems: Introduction to CPS Securities, Basic Techniques in CPS Securities, Cyber Security Requirements, Attack Model and Countermeasures, Advanced Techniques in CPS Securities.	4
8.	CPS Application: Health care and Medical Cyber-Physical Systems, Smart grid and Energy Cyber-Physical Systems, WSN based Cyber-Physical Systems, Smart Cities, CPPS & Agent based system for implementation.	4

Text book and Reference books:

1. Rajeev Alur, Principles of Cyber-Physical Systems, MIT Press, 2015.
2. Lee, E. A. and Seshia, S. A., Introduction to Embedded Systems: A Cyber-Physical Systems Approach, MIT Press, 2017.
3. Song, H., Rawat, D., Jeschke, S., Brecher, C., Cyber-Physical Systems: Foundations, Principles and Applications, Academic Press, 2017.
4. Reference Books:
5. Kim, K.D. and Kumar, P.R., Cyber-Physical Systems: A Perspective at the Centennial, Proceedings of the IEEE, 2012.

Course Outcome:	
On completion of the course students will be able to	
PEC-IT701B.1	Explain the architecture, components, and applications of Cyber-Physical Systems.
PEC-IT701B.2	Analyze CPS platforms including processors, sensors, actuators, and real-time scheduling models.
PEC-IT701B.3	Design and evaluate control strategies for dynamical CPS systems under practical constraints.
PEC-IT701B.4	Compare communication protocols and standards used in CPS and analyze interoperability issues.
PEC-IT701B.5	Implement and evaluate CPS considering performance metrics such as latency, scheduling, and fault tolerance.
PEC-IT701B.6	Identify security threats in CPS and propose suitable countermeasures for secure CPS design.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	-	-	-	2	1	-
CO2	3	3	2	1	2	-	-	-	-	-	-	3	2	-
CO3	3	3	3	2	2	-	-	-	-	-	-	3	2	1
CO4	2	3	2	1	2	-	-	-	-	-	-	2	3	-
CO5	3	3	3	2	3	-	-	-	-	-	1	3	3	2
CO6	2	3	2	2	2	1	1	1	-	-	-	2	3	1

Generative AI

Code: PEC- IT-701C

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1.	Understand the fundamentals of Generative AI and Agentic AI systems.
2.	Learn advanced generative models and Large Language Models and concepts of prompt engineering.
3.	Become familiar with design of intelligent agent architectures with communication protocols.

Pre-Requisite:	
1	Probability and Statistics
2	Machine Learning Fundamentals

Unit	Content	Hrs.
1.	History and evolution of AI/ML, Deep learning revolution, Transfer learning, Basics of Natural Language Processing, Structure of Artificial Neural Networks, Training process, Parameters and Hyperparameters, Backpropagation.	6
2.	Generative AI models: GANs, VAEs, Transformers, Attention mechanism, Long Short-Term Memory (LSTMs), Large Language Models (LLMs), Prompt engineering basics, Applications of generative models in text, image and multimodal systems.	10

Unit	Content	Hrs.
3.	Introduction to Intelligent Agents, Agent–environment interaction, Perception–Reasoning–Action loop, Tool usage and structured function calling, Embeddings and vector databases, Retrieval-Augmented Generation (RAG), Document chunking, similarity search and hallucination mitigation.	10
4.	Single-agent and multi-agent systems, Workflow orchestration and task decomposition, Agent frameworks overview (LangChain, LangGraph, AutoGen, CrewAI), Agent communication mechanisms, Model Context Protocol (MCP), State management, Human-in-the-loop systems.	6
5.	Applications of Generative AI in content generation, image synthesis, coding assistants, automation, healthcare, finance, education, and research support. Applications of Agentic AI in task automation, intelligent assistants, workflow orchestration, multi-agent collaboration, decision support systems, and real-time problem solving. Common challenges in generative and agentic systems including hallucinations, bias, evaluation difficulties, safety risks, and computational constraints. Guardrails, basic safety mechanisms, and ethical considerations in AI system design.	4

Text book and Reference books:

- Deep Learning (Adaptive Computation and Machine Learning series): Ian Goodfellow , Yoshua Bengio, Aaron Courville: MIT Press, 2016
- Generative AI with Lang chain: Ben Auffarth: Packt, 2025
- Building Agentic AI Systems: Anjanava Biswas, Wrick Talukdar: Packt, 2025
- Building Multimodal Generative AI and Agentic Applications: Indrajit Kar(eBook): BPB publishers, 2025
- Generative AI on AWS: Building Context-Aware Multimodal Reasoning Application: Chris Fregly , Antje Barth , et al. O'Reilly, 2024
- Autonomous Agents and Multiagent Systems. Best and Visionary Papers: Springer, 2022

Course Outcome:	
On completion of the course students will be able to	
PEC-IT701C.1	Explain the theoretical foundations of Generative AI, including GANs, VAEs, Transformers, neural network training, and backpropagation.
PEC-IT701C.2	Analyze advanced neural network architectures and Large Language Models (LLMs), including attention mechanisms, transformer-based models, and their performance characteristics.
PEC-IT701C.3	Design Retrieval-Augmented Generation (RAG) systems using document ingestion, embeddings, vector search, and grounding techniques.
PEC-IT701C.4	Develop agent-based AI systems with task planning, tool integration, multi-agent collaboration, and communication protocols such as MCP.
PEC-IT701C.5	Discuss applications of Generative and Agentic AI systems and explain the explain basic safety mechanisms and ethical considerations.
PEC-IT701C.6	Identify technical challenges in Generative AI systems such as hallucinations, bias, and evaluation limitations, and suggest basic mitigation approaches.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	1	2	-	-	-	-	-	1	3	2	
CO2	2	3	-	2	3	-	-	-	-	-	2	2	3	
CO3	2	2	3	2	3	-	-	-	1	-	1	2	3	2
CO4	2	2	3	2	3	-	1	3	2	2	2	2	3	2
CO5	1		1	-	1	3	3	1	2	1	2	1	2	1
CO6	1	3	1	3	2	1	2	-	1	-	3	2	2	-

Human Computer Interaction

Code: PEC-IT 701D

Contact Hours:

3 Credit point:

3

Course Objective:	
1.	Understand the interdisciplinary foundations of Human-Computer Interaction.
2.	Apply cognitive and human factors principles in interface design.
3.	Design usable, accessible, and effective user interfaces for desktop and web systems.
4.	Evaluate interfaces using usability engineering methods.
5.	Explore emerging interaction paradigms including AI, AR/VR, and ubiquitous systems.

Pre-Requisite:	
1.	Basic Software Engineering
2.	Fundamentals of Computer Systems

Unit	Content	Hrs.
1.	Introduction to HCI: Definition, goals and importance of HCI, Importance of good design, Benefits of well-designed interfaces, History of user interfaces (CLI to GUI to Web), Direct and indirect manipulation, WIMP interfaces, GUI vs Web interface characteristics, Principles of user interface design (clarity, consistency, feedback, simplicity, control, flexibility).	6
2.	Human Factors in Design: Human perception (Gestalt principles), visual acuity, signals vs noise, Memory (sensory, short-term, long-term), recognition vs recall, Mental models and conceptual models, Human interaction speeds (reading, typing, pointing), Psychological responses to poor design, Individual differences and inclusive design considerations.	10
3.	User-Centered Design Process: Need finding methods (interviews, surveys, observation), Ethical considerations in human studies, Screen design principles (layout, navigation, visual hierarchy, minimizing clutter), Interaction styles (menu-based, form fill-in, command language, touch interfaces), Prototyping (low-fidelity, high-fidelity, wireframes), Wizard-of-Oz method.	10
4.	Usability Engineering and Evaluation: HCI in software lifecycle, Iterative design and usability engineering, Design rules and heuristics (Nielsen, Shneiderman), Evaluation methods (heuristic evaluation, cognitive walkthrough, usability testing, A/B testing), Universal design principles, Multi-modal interaction and accessibility standards.	6
5.	Advanced Interaction Paradigms: Cognitive models (Model Human Processor, GOMS, task hierarchies), Information visualization principles, Human-AI interaction (trust, transparency, explainability), Ubiquitous computing and wearable systems, Virtual and Augmented Reality interfaces, HCI challenges for next billion users and multilingual contexts.	4

Text book and Reference books:

1. Wilbert O. Galitz – *The Essential Guide to User Interface Design*, Wiley, 2002
2. Jennifer Preece, Yvonne Rogers, Helen Sharp – *Interaction Design*: John Wiley & Sons, 2002
3. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale – *Human-Computer Interaction*, Pearson/Prentice-Hall, 2009
4. Alan Cooper, Robert Reimann, David Cronin, Christopher Noessel : *About Face: The Essentials of Interaction Design*, Wiley, 2014

Course Outcome:	
On completion of the course students will be able to	
PEC-IT701D.1	Explain the theoretical foundations and evolution of Human-Computer Interaction and user interface design principles.

PEC-IT701D.2	Apply human cognitive and perceptual principles in the design of usable interfaces.
PEC-IT701D.3	Design and prototype user interfaces using user-centered design methodologies.
PEC-IT701D.4	Conduct usability evaluation using qualitative and quantitative methods.
PEC-IT701D.5	Analyze and design advanced interaction systems including AI-enabled, AR/VR, and ubiquitous interfaces.
PEC-IT706D.6	Design inclusive and accessible user interfaces considering diverse users, cultural contexts, and assistive technologies.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	1	-	2	3	2	-
CO2	2	3	2	1	1	-	1		-	-	1	2	2	-
CO3	2	2	3	2	3	1	1	3	2	3	1	2	3	3
CO4	1	3	2	3	2	1	1	2	2	-	2	1	2	2
CO5	2	2	3	2	2	2	1	1	1	1	2	2	2	2
CO6	1	2	3	2	2	2	3	2	2	-	1	1	2	2

Project Management and Entrepreneurship

Code: OEC-IT-701A

Contacts

Hours: 3

Credit point: 3

Course Objective:	
1.	To understand the fundamental principles, tools, and techniques of project management.
2.	To develop entrepreneurial skills including opportunity identification, business planning, and risk management.
3.	To apply project management methodologies and entrepreneurial strategies in real-world business and technology ventures.

Pre-Requisite:	
1.	Basic knowledge of Management Principles and Organizational Behavior.
2.	Fundamental understanding of Economics and Business Environment.

Unit	Content	Hrs.
1.	Entrepreneurship: Introduction: Meaning and Concept of Entrepreneurship, Innovation and entrepreneurship, Contributions of entrepreneurs to the society, risk-opportunities perspective and mitigation of risks	2
2.	Entrepreneurship – An Innovation: Challenges of Innovation, Steps of Innovation Management, Idea Management System, Divergent v/s Convergent Thinking, Qualities of a prospective Entrepreneur	2
3.	Idea Incubation: Factors determining competitive advantage, Market segment, blue ocean strategy, Industry and Competitor Analysis (market structure, market size, growth potential), Demand-supply analysis	4

Unit	Content	Hrs.
4.	Entrepreneurial Motivation: Design Thinking - Driven Innovation, TRIZ (Theory of Inventive Problem Solving), Achievement motivation theory of entrepreneurship – Theory of McClelland, Harvesting Strategies	2
5.	Information: Government incentives for entrepreneurship, Incubation, acceleration. Funding new ventures – bootstrapping, crowd sourcing, angel investors, Government of India's efforts at promoting entrepreneurship and innovation – SISI, KVIC, DGFT, SIDBI, Defense and Railways	4
6.	Closing the Window: Sustaining Competitiveness, Maintaining Competitive Advantage, the Changing Role of the Entrepreneur.	2
7.	Applications and Project Reports Preparation	4
8.	Project Management: Definitions of Project and Project Management, Issues and Problems in Project Management, Project Life Cycle - Initiation / Conceptualization Phase, Planning Phase, Implementation / Execution Phase, Closure / Termination Phase	4
9.	Project Feasibility Studies – Pre-Feasibility and Feasibility Studies, Preparation of Detailed Project Report, Technical Appraisal, Economic/Commercial/Financial Appraisal including Capital Budgeting Process, Social Cost Benefit Analysis	2
10.	Project Planning – Importance of Project Planning, Steps of Project Planning, Project Scope, Work Breakdown Structure (WBS) and Organization Breakdown Structure (OBS), Phased Project Planning	2
11.	Project Scheduling and Costing – Gantt chart, CPM and PERT Analysis, Identification of the Critical Path and its Significance, Calculation of Floats and Slacks, Crashing, Time Cost Trade-off Analysis, Project Cost Reduction Methods.	6
12.	Project Monitoring and Control – Role of Project Manager, MIS in Project Monitoring, Project Audit	2

Text book and Reference books:

1. Innovation and Entrepreneurship by Drucker, P.F.; Harper and Row
2. Business, Entrepreneurship and Management: Rao, V.S.P.; Vikas
3. Entrepreneurship: Roy Rajeev; OUP.
4. Text Book of Project Management: Gopal krishnan, P. and Ramamoorthy, V.E.; McMillan
5. Project Management for Engineering, Business and Technology: Nicholas, J.M., and Steyn, H.; PHI

Course Outcome:	
On completion of the course students will be able to	
OEC-IT701A.1	Understand the theories of entrepreneurship and Entrepreneurial Development Programs.
OEC-IT701A.2	Create innovative business ideas and market opportunities.
OEC-IT701A.3	Understand the importance of Project Management and Project's life cycle
OEC-IT701A.4	Analyze the learning and understand techniques for Project planning, scheduling and Execution Control
OEC-IT701A.5	Analyze Project Finance and project report and the role of stakeholders. Apply the risk management plan.
OEC-IT701A.6	Evaluate Social Sector Perspectives and Social Entrepreneurship

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2	3	3	1	2	2	2	2	3	3	3
CO2	3	2	2	2	3	1	-	1	2	1	2	3	2	2
CO3	2	2	-	-	-	-	-	2	3	3	2	2	2	-

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO4	1	2	2	2	1	-	-	2	3	2	2	1	2	2
CO5	1	2	-	-	-	2	-	1	2	2	2	1	2	-
CO6	1	-	-	-	-	3	-	3	1	2	2	1	-	-

Organizational Behavior

Code: OEC-IT-701B

Contacts Hours: 3

Credit point: 3

Course Objective:	
1.	To understand individual, group, and organizational factors that influence behavior in organizations.
2.	To develop skills in motivation, leadership, communication, teamwork, conflict management, and decision-making.
3.	To analyze organizational culture, change, and contemporary workplace issues in technology-oriented organizations.
Pre-Requisite:	
1.	Basic understanding of management principles and business organizations.
2.	General awareness of workplace communication, teamwork, and organizational processes.

Unit Content Hrs.

Unit	Content	Hrs.
1.	Introduction to Organizational Behavior: Meaning, nature, scope and importance; contributing disciplines; foundations of individual behavior; personality; values; attitudes; perception; learning; emotions and emotional intelligence.	8
2.	Motivation and Individual Decision-Making: Motivation concepts and theories; Maslow, Herzberg, McGregor, expectancy and equity theories; job satisfaction; employee engagement; perception and attribution; rational and bounded-rationality decision-making; individual differences.	7
3.	Group Behavior and Leadership: Groups and teams; stages of group development; group norms and roles; communication; team effectiveness; leadership theories; trait, behavioral and contingency approaches; power and politics; negotiation and conflict management.	7
4.	Organizational Structure and Culture: Organizational structure and design; centralization and decentralization; organizational culture; diversity and inclusion; organizational climate; organizational communication; work stress; employee well-being; workplace ethics.	8
5.	Organizational Change and Contemporary Issues: Forces of change; resistance to change; change management; organizational development; technology and digital transformation; virtual and hybrid teams; knowledge management; innovation; cross-cultural behavior; future of work and responsible organizational practices.	6

Textbook and Reference books:

1. Stephen P. Robbins and Timothy A. Judge, Organizational Behavior, Pearson.
2. Fred Luthans, Organizational Behavior, McGraw-Hill.
3. Udai Pareek, Understanding Organizational Behaviour, Oxford University Press.
4. Arun Kumar and Rachana Sharma, Organizational Behaviour, Atlantic Publishers.
5. John W. Newstrom, Organizational Behavior: Human Behavior at Work, McGraw-Hill.

Course Outcome:	
On completion of the course students will-be able to	
OEC-IT-701B.1	Explain the foundations, scope, and major concepts of Organizational Behavior.
OEC-IT-701B.2	Analyze how personality, perception, attitudes, learning, emotions, and motivation influence individual behavior.
OEC-IT-701B.3	Apply group, communication, leadership, power, conflict, and negotiation concepts to organizational situations.
OEC-IT-701B.4	Analyze organizational structure, culture, diversity, communication, and employee well-being.
OEC-IT-701B.5	Evaluate organizational change, digital transformation, virtual work, and contemporary workplace challenges.

OEC-IT-701B.6			Develop evidence-based approaches for improving teamwork, leadership, motivation, and organizational effectiveness.											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	1	2	-	-
CO2	2	3	2	2	3	-	-	-	2	2	2	3	2	-
CO3	2	3	2	2	-	2	1	1	2	2	2	2	2	-
CO4	2	3	2	3	-	2	1	1	2	2	2	2	2	-
CO5	2	2	3	2	2	1	-	1	3	3	2	2	3	2
CO6	1	2	2	2	2	2	2	2	2	3	3	2	2	3

Industrial Training-I/ Internship-I

Code: PCC-IT-781

Contacts

Hours: 0

Credit point: 3

Course Outcome:	
On completion of the course students will be able to	
PSC-IT781.1	To expose students to the 'real' working environment and get acquainted with the organization structure, business operations and administrative functions.
PSC-IT781.2	To have hands-on experience in the students' related field so that they can relate and reinforce what has been taught at the Institute.
PSC-IT781.3	To develop schedule in terms of resources for a project and Influence the time management of a project.
PSC-IT781.4	To promote cooperation and to develop synergetic collaboration between industry and the Institute in promoting a knowledgeable society.
PSC-IT781.5	An ability to write technical documents and give oral presentations related to the work completed.
PSC-IT781.6	To set the stage for future recruitment by potential employers.

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1			3	1	1	2		2	–	1	2
CO2	3	2	2		3						2	3	3	1
CO3	2			2				2		3	3	1	1	2
CO4		–	–	–	–	2	1	2	1				1	2
CO5	2	1		–	–	–	–	–	3			2	3	1
CO6	–	–	–	–	–	1	3	2	2	1	3	–	1	3

Project-I

Code: PROJ -IT-781

Contacts Hours: 12

Credit point: 6

Course Outcome:	
On completion of the course students will be able to	
PROJ-IT 881.1	Ability to identify, define & articulate a relevant engineering problem through a thorough literature review & analysis.
PROJ-IT 881.2	Demonstrate the ability to apply appropriate engineering principles, tools & techniques to solve complex engineering problems.
PROJ-IT 881.3	Ability to organize, executes, and manages project tasks, timelines, and resources effectively while adhering to project constraints.
PROJ-IT 881.4	Competence in designing, developing, testing, and interpreting an engineering solution or prototype that meets specified requirements.
PROJ-IT 881.5	Develop effective oral & written communication skills for defending project ideas, progress & results through technical reports & presentations.
PROJ-IT 881.6	Analyze complex engineering scenarios to evaluate ethical implications and team dynamics, and formulate strategies for promoting ethical conduct & effective collaboration.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	3	1	1	1	1	2	-	1	3	2	1
CO2	3	3	3	2	3	1	1	1	1	1	1	3	3	1
CO3	2	2	2	2	3	1	2	3	2	3	1	3	2	2
CO4	3	2	3	3	3	2	1	2	2	2	1	3	3	2
CO5	1	2	2	2	1	1	1	3	3	2	2	1	2	2
CO6	1	3	1	1	-	3	3	3	2	2	1	1	1	2

Eighth Semester (4thYear)

Augmented & Virtual Reality

Code: PEC- IT 801A

Contacts

Hours: 3L

Credit point: 3

Course Objective:	
1.	To understand fundamental concepts, technologies, and system components of Augmented and Virtual Reality.
2.	To develop skills for designing and implementing interactive AR/VR applications using modern tools and platforms.
3.	To analyze usability, performance, and ethical considerations for real-world immersive system deployment.
Pre-Requisite:	
1.	Basic knowledge of computer graphics, programming concepts, and mathematics fundamentals.
2.	Familiarity with object-oriented programming and basic understanding of human-computer interaction principles.

Unit	Content	Hrs.
1.	Introduction of Virtual Reality: Fundamental Concept and Components of Virtual Reality - Primary Features and Present Development on Virtual Reality - Multiple Models of Input and Output Interface in Virtual Reality: Input - Tracker - Sensor - Digital Glove - Movement Capture - Video-based Input - 3D Menus & 3DScanner – Output - Visual /Auditory / Haptic Devices.	10
2.	Visual Computation in Virtual Reality: Fundamentals of Computer Graphics - Software and Hardware Technology on Stereoscopic Display - Advanced Techniques in CG: Management of Large-Scale Environments & Real Time Rendering.	6
3.	Interactive Techniques in Virtual Reality: Body Track - Hand Gesture - 3D Manus - Object Grasp. Development Tools and Frameworks in Virtual Reality: Frameworks of Software Development Tools in VR. X3D Standard; Vega - MultiGen - Virtools.	8
4.	Application of VR in Digital Entertainment: VR Technology in Film and TV Production - VR Technology in Physical Exercises and Games - Demonstration of Digital Entertainment by VR.	6
5.	Augmented and Mixed Reality: Taxonomy - technology and features of augmented reality - difference between AR and VR - Challenges with AR - AR systems and functionality - Augmented reality methods - visualization techniques for augmented reality - wireless displays in educational augmented reality applications - mobile projection interfaces - marker-less tracking for augmented reality - enhancing interactivity in AR environments - evaluating AR systems.	8

Text book and Reference books:

1. An Introduction to Virtual and Augmented Reality: Designing Impactful Experiences – Marco Gillies & Xueni Pan (*Routledge*).
2. Augmented and Virtual Reality: Theory and Practice – Tony Thomas, Ravi Prakash, Abhishek Kaushik (*Wiley*).
3. Creating Augmented and Virtual Realities – Erin Pangilinan, Steve Lukas, Vasanth Mohan (*O'Reilly*).
4. Virtual & Augmented Reality for Dummies – Paul Mealy (*Wiley*).
5. Advances in Augmented Reality and Virtual Reality – Jitendra Kumar Verma & Sudip Paul.

Course Outcome:	
On completion of the course students will be able to	
PEC- IT 801A.1	Recall and explain the foundational concepts, history, and evolution of AR/VR systems.
PEC- IT 801A.2	Differentiate and analyze various AR/VR hardware and software components used in immersive systems.
PEC- IT 801A.3	Apply, model, and design basic AR/VR experiences using development tools and SDKs.
PEC- IT 801A.4	Evaluate the interaction techniques and usability challenges in AR/VR applications.
PEC- IT 801A.5	Create innovative AR/VR prototypes that solve real-world problems across domains.
PEC- IT 801A.6	Demonstrate ethical, safety, and future trend awareness in AR/VR implementation.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	0	0	0	0	0	0	0	0	2	3	0	0
CO2	3	3	1	0	2	0	0	0	0	0	0	3	2	0
CO3	2	3	3	1	3	0	0	0	1	0	0	2	3	2

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO4	2	3	2	2	2	2	1	0	2	0	0	2	2	2
CO5	2	3	3	2	3	1	1	2	2	2	2	2	3	3
CO6	1	2	1	0	1	3	3	1	2	0	3	1	2	2

Cyber Security and Ethical Hacking

Code: PEC- IT 801B

Contacts Hours: 3L

Credit point: 3

Course Objective:	
1.	To understand fundamental concepts, principles, and techniques of cyber security and ethical hacking.
2.	To develop skills for identifying vulnerabilities, performing penetration testing, and securing computer systems and networks.
3.	To analyze security threats, legal frameworks, and ethical responsibilities involved in protecting digital assets and information systems.

Pre-Requisite:	
1.	Basic knowledge of computer networks, operating systems, and fundamental programming concepts.
2.	Understanding of information security basics, internet protocols, and system administration fundamentals.

Unit	Content	Hrs.
1.	Introduction: Hacking Impacts, The Hacker Framework: Planning the test, Sound Operations, Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Final Analysis, Deliverable, Integration Information Security Models: Computer Security, Network Security, Service Security, Application Security, Security Architecture Information Security Program: The Process of Information Security, Component Parts of Information Security Program, Risk Analysis and Ethical Hacking.	9
2.	The Business Perspective: Business Objectives, Security Policy, Previous Test Results, Business Challenges Planning for a Controlled Attack: Inherent Limitations, Imposed Limitations, timing is Everything, Attack Type, Source Point, Required Knowledge, Multi-Phased Attacks, Teaming and Attack Structure, Engagement Planner, The Right Security Consultant, The Tester, Logistics, Intermediates, Law Enforcement.	10
3.	Preparing for a Hack: Technical Preparation, Managing the Engagement Reconnaissance: Social Engineering, Physical Security, Internet Reconnaissance.	4
4.	Enumeration Methods: Enumeration Techniques, Soft Objective, Looking Around or Attack, Elements of Enumeration, Preparing for the Next Phase Exploitation: Intuitive Testing, Evasion, Threads and Groups, Operating Systems, Password Crackers, RootKits, applications, Wardialing, Network, Services and Areas of Concern.	8
5.	Security Framework: The Deliverable, The Document, Overall Structure, Aligning Findings, Presentation Integration: Integrating the Results, Integration Summary, Mitigation, Defense Planning, Incident Management, Security Policy, Conclusion.	7

Text book and Reference books:

1. Gray Hat Hacking: The Ethical Hacker's Handbook – Michael Gregg, et al. (McGraw-Hill / TMH).
2. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws – Dafydd Stuttard & Marcus Pinto (Wiley).
3. Hands-On Ethical Hacking and Network Defense – Michael T. Simpson, Kent Backman, James E. Corley (Cengage Learning).
4. Social Engineering: The Science of Human Hacking – Christopher Hadnagy.
5. Black Hat Python: Python Programming for Hackers & Pentesters – Justin Seitz.

Course Outcome:

On completion of the course students will be able to	
PEC- IT 801B.1	Recall fundamental concepts of cyber security, ethical hacking principles, security models, and terminology.
PEC- IT 801B.2	Explain various types of cyber threats, attack methodologies, vulnerabilities, and security policies used in organizational environments.
PEC- IT 801B.3	Apply reconnaissance, scanning, enumeration, and vulnerability assessment techniques using standard security tools and methodologies.
PEC- IT 801B.4	Analyze system, network, and application security weaknesses by interpreting penetration testing results and identifying attack vectors.
PEC- IT 801B.5	Evaluate security risks, ethical implications, and legal considerations to determine appropriate mitigation and defense strategies.
PEC- IT 801B.6	Design a structured penetration testing plan and prepare professional security assessment reports with recommendations.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	2	-	-	-	-	-	3	-	-
CO2	3	3	2	-	-	2	-	-	-	-	-	3	3	-
CO3	-	3	2	2	2	2	-	-	-	-	-	-	3	-
CO4	-	3	-	3	2	-	-	-	2	-	-	-	3	-
CO5	-	3	-	3	3	-	-	2	2	-	-	-	3	-
CO6	-	-	3	-	3	3	-	-	-	3	2	-	-	3

Computational Geometry

PEC- IT-801C

Contacts Hours:

3

Credit point: 3

Course Objective:	
1.	To understand fundamental geometric structures and algorithms used in computational geometry.
2.	To design efficient algorithms for geometric problems such as convex hulls, intersections, and triangulations.
3.	To apply geometric algorithms in real-world applications like graphics, robotics, and spatial data processing.

Pre-Requisite:	
1.	Data Structures and Algorithms
2.	Discrete Mathematics / Linear Algebra basics

Unit	Content	Hrs.
1.	Introduction: Geometric problems and applications, collinearity, closest pair of points (divide & conquer), geometric primitives (orientation test, area of triangle), line segment intersection basics	6
2.	Convex Hulls: Naive method, Gift Wrapping (Jarvis March), Graham Scan, Quick Hull, Divide & Conquer, lower bound proof, convex hulls in 3D	6
3.	Intersection & Polygon Algorithms: Line segment intersection (Bentley–Ottmann sweep line algorithm), Art Gallery Problem (Fisk's proof), polygon triangulation (ear clipping, monotone polygon triangulation, trapezoidalization $O(n \log n)$)	8
4.	Geometric Data Structures & Searching: Range searching (k-d trees, range trees), interval trees, nearest neighbor search	10

Unit	Content	Hrs.
5.	Voronoi & Delaunay Structures: Voronoi diagrams, Delaunay triangulation, properties and applications	5
6.	Motion Planning: Configuration space, obstacles, visibility graph, shortest path, probabilistic roadmaps (PRM), rapidly exploring random trees (RRT)	5

Text book and Reference books:

1. Computational Geometry: Algorithms and Applications, Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars, Springer
2. Computational Geometry: An Introduction, Franco P. Preparata, Michael Ian Shamos, Springer
3. Computational Geometry in C, Joseph O'Rourke, Cambridge University Press

Course Outcome:	
On completion of the course students will be able to	
PEC-IT801C.1	Apply geometric primitives and basic algorithms to solve computational geometry problems.
PEC-IT801C.2	Analyze and implement convex hull algorithms in 2D and 3D.
PEC-IT801C.3	Design algorithms for geometric intersections and polygon triangulation.
PEC-IT801C.4	Apply geometric data structures for efficient searching and spatial queries.
PEC-IT801C.5	Analyze Voronoi diagrams and Delaunay triangulations for spatial partitioning.
PEC-IT801C.6	Design motion planning algorithms for robotics and real-world applications.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	-	-	-	-	1	-	3	2	1
CO2	3	3	2	2	2	-	-	-	-	1	-	3	3	2
CO3	3	3	3	2	2	-	-	-	-	1	-	3	3	2
CO4	2	3	2	2	3	-	-	-	-	1	-	2	3	2
CO5	2	3	2	2	2	-	-	-	-	1	-	2	3	2
CO6	3	3	3	2	3	1	-	-	1	2	1	3	3	3

Social Network Analysis

PEC- IT-801D

Contacts

Hours: 3

Credit point: 3

Course Objectives:	
1.	Analyze structural properties of social networks.
2.	Explore community detection and information diffusion.
3.	Apply network metrics and models to real-world datasets.

Pre-Requisite:

1.	Discrete Mathematics (Graph Theory, Sets, Relations, Probability Basics)
2.	Data Structures and Algorithms (Graphs, Trees, Searching, Basic Algorithm Analysis)

Unit	Content	Hrs.
1.	Introduction: What is Social Networks Analysis, Evolution of Social Media Networks, Types of Networks (Directed, Undirected, Weighted, and Signed), Graph Theory Fundamentals (Nodes, Edges, Paths, Walks, Components, and Connectivity), Network Representation (Adjacency Matrix, Adjacency List, and Edge List), and Applications of Social Network Analysis.	3
2.	Network Structure and Metrics: Degree Distribution, Power Law and Scale-Free Networks, Centrality Measures (Degree Centrality, Betweenness Centrality, Closeness Centrality, Eigenvector Centrality), Clustering Coefficient, Network Density, Reciprocity, Transitivity, Assortativity, Small-World Property, Network Robustness	4
3.	Network Models and Link Analysis: Properties of Real-World Networks, Random Graph Model (Erdős–Rényi Model), Small-World Model (Watts–Strogatz Model), Preferential Attachment Model (Barabási–Albert Model), Link Analysis Concepts, Page Rank Algorithm, HITS Algorithm, Strong and Weak Ties, Structural Balance in Signed Networks.	6
4.	Community Detection and Link Prediction: Community Structure in Networks, Modularity, Girvan–Newman Algorithm, Louvain Method, Overlapping and Disjoint Communities, Community Evaluation Metrics, Link Prediction Problem, Similarity-Based Methods (Common Neighbours, Jaccard Coefficient, Adamic-Adar Index, Preferential Attachment Score), Supervised Link Prediction Techniques.	8
5.	Information Diffusion and Social Influence: Information Cascades, Diffusion in Social Networks, Independent Cascade Model, Linear Threshold Model, Epidemic Models (SI, SIR, SIS), Influence Maximization Problem, Viral Marketing, Rumour and Fake News Propagation, Temporal Networks.	6
6.	Advanced Topics in Social Network Analysis: Graph Representation Learning, Network Embeddings (Deep Walk, Node2Vec), Introduction to Graph Neural Networks, Anomaly Detection in Social Networks, Bot Detection and Spam Detection, Recommender Systems in Social Media, Privacy, Security and Ethical Issues in Social Network Analysis.	5

Text book and Reference books:

1. Charu C. Aggarwal, Social Network Data Analytics, Springer; 2011.
2. S. Wasserman, K. Faust: Social Network Analysis: Methods and Applications, Cambridge Univ Press, 1994.
3. Scott, J. (2007). Social network analysis: A handbook (2nd Ed.). Newbury Park, CA: Sage.
4. Knoke (2008). Social Network Analysis, (2nd Ed). Sage.

Course Outcome:	
On completion of the course students will be able to	
PEC-IT801D.1	Model social media platforms as graphs and analyze their structural properties.
PEC-IT801D.2	Compute and interpret various network metrics.
PEC-IT801D.3	Apply link analysis and ranking algorithms.
PEC-IT801D.4	Detect communities and predict future links.
PEC-IT801D.5	Analyze diffusion processes and influence propagation.

PEC-IT801D.6	Apply machine learning techniques to solve advanced network problems.
--------------	---

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	-	-	-	-	-	-	3	2	-
CO2	3	3	2	2	2	-	-	-	-	-	-	3	2	-
CO3	3	3	3	2	3	-	-	-	-	-	-	3	3	-
CO4	3	3	3	3	3	-	-	-	-	-	-	3	3	-
CO5	3	3	2	3	2	1	-	1	-	-	-	3	2	-
CO6	3	3	3	3	3	-	-	1	2	-	-	3	3	2

Industrial Training-II/ Internship-II

Code: PSC-IT-881

Contacts Hours:

0

Credit point: 3

Course Outcome:	
On completion of the course students will be able to	
PSC-IT-881.1	To expose students to the 'real' working environment and get acquainted with the organization structure, business operations and administrative functions.
PSC-IT-881.2	To have hands-on experience in the students' related field so that they can relate and reinforce what has been taught at the Institute.
PSC-IT-881.3	To develop schedule in terms of resources for a project and influence the time management of a project.
PSC-IT-881.4	To promote cooperation and to develop synergetic collaboration between industry and the Institute in promoting a knowledgeable society.
PSC-IT-881.5	An ability to write technical documents and give oral presentations related to the work completed.
PSC-IT-881.6	To set the stage for future recruitment by potential employers.

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1			3	1	1	2		2	–	1	2
CO2	3	2	2		3						2	3	3	1
CO3	2			2				2		3	3	1	1	2
CO4		–	–	–	–	2	1	2	1				1	2
CO5	2	1		–	–	–	–	–	3			2	3	1
CO6	–	–	–	–	–	1	3	2	2	1	3	–	1	3

Project-II

Code: PROJ -IT-881

Contacts Hours: 12

Credit point: 6

Course Outcome:	
On completion of the course students will be able to	
PROJ-IT 881.1	Ability to identify, define & articulate a relevant engineering problem through a thorough literature review & analysis.
PROJ-IT 881.2	Demonstrate the ability to apply appropriate engineering principles, tools & techniques to solve complex engineering problems.
PROJ-IT 881.3	Ability to organize, executes, and manages project tasks, timelines, and resources effectively while adhering to project constraints.
PROJ-IT 881.4	Competence in designing, developing, testing, and interpreting an engineering solution or prototype that meets specified requirements.
PROJ-IT 881.5	Develop effective oral & written communication skills for defending project ideas, progress & results through technical reports & presentations.
PROJ-IT 881.6	Analyze complex engineering scenarios to evaluate ethical implications and team dynamics, and formulate strategies for promoting ethical conduct & effective collaboration.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	3	1	1	1	1	2	-	1	3	2	1
CO2	3	3	3	2	3	1	1	1	1	1	1	3	3	1
CO3	2	2	2	2	3	1	2	3	2	3	1	3	2	2
CO4	3	2	3	3	3	2	1	2	2	2	1	3	3	2
CO5	1	2	2	2	1	1	1	3	3	2	2	1	2	2
CO6	1	3	1	1	-	3	3	3	2	2	1	1	1	2