THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

COURSE SCHEME

FOR

MCA

2019

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
MASTER OF COMPUTER APPLICATIONS (MCA)

SEMESTER-I

S. No.	Course No.	Title	L	T	P	Cr
1	PCA101	Discrete Mathematics	3	1	0	3.5
2	PCA102	Computer Programming	3	0	2	4.0
3	PCA103	Computer Organization and Architecture	3	0	2	4.0
4	PCA104	Web Designing	2	0	2	3.0
5	PCA105	Statistical and Numerical Methods	3	1	2	4.5
		Total	14	2	8	19.0

SEMESTER-II

S. No.	Course No.	Title	L	T	P	Cr
1	PCA201	Optimization Techniques	3	0	2	4.0
2	PCA202	Graph Theory and Applications	3	1	0	3.5
3	PCA203	Object Oriented Programming	3	0	2	4.0
4	PCA204	Programming languages	3	0	2	4.0
5	PCA205	Organizational Behaviour	3	0	0	3.0
		Total	15	1	6	18.5

SEMESTER-III

S. No.	Course No.	Title	L	T	P	Cr
1	PCA301	Computer Networks	2	0	2	3.0
2	PCA302	Operating Systems	2	0	2	3.0
3	PCA303	Software Engineering	3	0	2	4.0
4	PCA304	Data Structures	3	0	2	4.0
5	PCA305	Data Base Management System	3	0	2	4.0
		Total	13	0	10	18.0

SEMESTER-IV

S. No.	Course No.	Title	L	T	P	Cr
1	PCA401	Theory of Computation	3	1	0	3.5
2	PCA402	Computer Graphics	3	0	2	4.0
3	PCA403	Big Data Analytics and Business Intelligence	3	0	2	4.0
4	PCA404	Artificial Intelligence	3	0	2	4.0
5	PCA405	Design and Analysis of Algorithms	3	0	2	4.0
6		Elective-I	3	0	2	4.0
		Total	18	1	10	23.5

SEMESTER-V

S. No.	Course No.	Title	L	T	P	Cr
1	PCA501	Mobile Application Development	3	0	2	4.0
2	PCA502	Capstone Project	0	0	0	6.0
3	PCA503	IT Project Management	3	0	2	4.0
4		Elective-II	3	0	2	4.0
5		Elective-III	3	0	2	4.0
		Total	12	0	8	22.0

SEMESTER-VI

Alternate Semester						
S. No.	Course No.	Title	L	T	P	Cr
1	PCA601	Project	0	0	0	4.0
2	PCA602	Human Computer Interaction	3	0	2	4.0
3	PCA603	Agile Software Development	3	0	2	4.0
4	PCA604	Enterprise Resource Planning	2	0	2	3.0
		Total	8	0	6	15.0

Sr. No.	Course No.	Title	L	T	P	Cr
1	PCA605	Project Semester	-	-	-	15
		Total	-	-	-	15
Grand Total of All Semesters						**116**

Electives are divided into five baskets

A) **High Performance Computing**: Parallel and Distributed Computing, GPU Computing, Cloud Computing

B) **Data Science:** Machine Learning, Deep Learning, Data Analytics and Visualization

C) **Information and Cyber Security**: Cryptography and Network Security, Secure Coding, Cyber Forensics

D) **System Management and Administration**: Database Administration, Network Administration, Web Analytics and Intelligence

E) **Computer Vision**: Image Processing, 3D Modelling and Animation, Natural Language Processing

ELECTIVE-I

S. No.	Course No.	Title	L	T	P	Cr
1	PCA406	Parallel and Distributed Computing	3	0	2	4.0
2	PCA407	Machine Learning	3	0	2	4.0
3	PCA408	Cryptography and Network Security	3	0	2	4.0
4	PCA409	Database Administration	3	0	2	4.0
5	PCA410	Image Processing	3	0	2	4.0

ELECTIVE-II

S. No.	Course No.	Title	L	T	P	Cr
1	PCA504	GPU Computing	3	0	2	4.0
2	PCA505	Deep Learning	3	0	2	4.0
3	PCA506	Secure Coding	3	0	2	4.0
4	PCA507	Network Administration	3	0	2	4.0
5	PCA508	3D Modelling and Animation	3	0	2	4.0

ELECTIVE-III

S. No.	Course No.	Title	L	T	P	Cr
1	PCA509	Cloud Computing	3	0	2	4.0
2	PCA510	Data Analytics and Visualization	3	0	2	4.0
3	PCA511	Cyber Forensics	3	0	2	4.0
4	PCA512	Web Analytics and Intelligence	3	0	2	4.0
5	PCA513	Natural Language Processing	3	0	2	4.0

PCA101 DISCRETE MATHEMATICS

L	T	P	Cr
3	1	0	3.5

Course Objective: Detailed study of various discrete and algebraic structures, basic logic, basics of counting and proof techniques.

Sets, Relations, and Functions: Sets: Operations on set, Representation of Discrete Structures, Fuzzy set, Multi-set, bijective function, Inverse and Composition of functions, Floor and Ceiling functions, Hashing functions, Recursive function, Growth of functions: Big-O notation, Big-Omega and Big-Theta Notations, Determining complexity of a program, Functions applications.

Relations: Reflexivity, symmetry, transitivity, Equivalence and partial-ordered relations, Asymmetric, Anti-symmetric and Irreflexive relation, Inverse and complementary relations, Partition and Covering of a set, N-ary relations and database, Representation of relation using matrices and digraph, Closure of relations, Warshall's algorithm, Lexicographic ordering, Hasse diagram, Lattices, Boolean Algebra: Boolean algebras as lattices, Representation and Minimizations of Boolean functions, Application of transitive closure in medicine and engineering. Application: Embedding a partial order.

Basic Logic: Propositional logic, Logical connectives, Truth tables, Normal forms (conjunctive and disjunctive), Validity of well-formed formula, Propositional inference rules (concepts of modus ponens and modus tollens), Predicate logic, Universal and existential quantification, Limitations of propositional and predicate logic.

Proof Techniques: Notions of implication, equivalence, converse, inverse, contra positive, negation, and contradiction, The structure of mathematical proofs, Direct proofs, Disproving by counter example, Proof by contradiction, Induction over natural numbers, Structural induction, Weak and strong induction (i.e., First and Second Principle of Induction), Recursive mathematical definitions, Well orderings.

Basics of Counting: Counting arguments: Set cardinality and counting, Sum and product rule, Inclusion-exclusion principle, Applications of Inclusion-exclusion, Arithmetic and geometric progressions, The pigeonhole principle, Permutations and combinations, Binomial coefficients and identities, Generalized Permutations and combinations, Solving Linear recurrence relations, Divide-and-conquer algorithms and Recurrence relations, Generating functions, Basic modular arithmetic.

Algebraic Structures: Monoids, Semi group, Group, Subgroups, Normal subgroups, Homomorphism, Congruencies, Ring, Field, Integral domain, Homomorphism, Congruencies, cosets and Lagrange Theorem, Group Codes, coding theory, Polynomial rings and polynomial codes, Applications of algebra to control structure of a program, The application of Residue Arithmetic to Computers.

Recommended Books:

1. Rosen, K.H., Discrete Mathematics and its Applications, McGraw Hill (2011), 7th ed.
2. Tremblay, J.P. and Manohar R., "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill (2007), 1st ed.
3. Haggard G., Schlipf J. and Whitesides, Sue, Discrete Mathematics for Computer Science, Cengage Learning, (2008), 2nd ed.
4. Johnsonbaugh R., Discrete Mathematics, Pearson Education ,(2007), 7th ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Perform operations on various discrete structures such as set, function and relation.
CLO2	Apply basic concepts of asymptotic notation in analysis of algorithm
CLO3	Comprehend formal logical arguments and translate statements from a natural language into its symbolic structures in logic.
CLO4	Get familiar with the techniques for constructing mathematical proofs and basics of counting.
CLO5	Identify and prove various properties of different algebraic structures such as group, rings and fields.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/ Tutorials/ Quizzes)	35

PCA102 COMPUTER PROGRAMMING				
	L	T	P	Cr
	3	0	2	4.0

Course objective: This course is designed to explore computing and to show students the art of computer programming. Students will learn some of the design principles for writing good programs.

Computers Fundamentals: Classification of Computers, Application of Computers, Basic organization of computer, Input and Output Devices, Binary Number System, Computer memory, Computer Software.

Algorithms and Programming Languages: Algorithm, Flowcharts, Pseudocode, Generation of Programming Languages.

Introduction to 'C' programming: Fundamentals, Structure of a 'C' program, Compilation and linking processes.

Expressions and Console I/O : Basic Data types, Identifier Names, Variables, Scope, Type qualifiers, Storage class specifiers, Constants, Operators, Reading and writing characters, Reading and writing strings, Formatted and console I/O, printf(), scanf(), Suppressing input.

Statements: True and False in C, Selection statements, Iteration statements, Jump statements, Expression statements, Block statements.

Arrays and Strings: Single dimension array, Two dimension array, Strings, Array of strings, Multi-dimension array, Array 7nalyse7g7y7on, Variable length arrays.

Pointers: Pointer variables, Pointer operators, Pointer expressions, Pointers and arrays, Multiple indirection, Pointer initialization, Pointers to arrays, Dynamically allocated arrays, Problems with pointers.

Functions: General form of a function, Understanding scope of a function, Function arguments, Command line arguments, Return statement, Recursion, Function prototype, Pointers to functions.

Structures, Unions, Enumerations, and Typedef: Structures, Array of structures, Passing structures to functions, Structure pointers, Arrays and structures within structures, Unions, Bit-fields, Enumerations, typedef.

File I/O: Streams and files, File system basics, fread() and fwrite(), fseek() and random access I/O, fprintf() and fscanf(), Standard streams.

Pre-processor and Comments: Pre-processor, #define, #error, #include, Conditional compilation directives, #undef, Single line and multiple line comments.

Laboratory work: To implement Programs for various kinds of programming constructs in C Language.

Recommended Books:

1. Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language, Prentice Hall(1988), 2nd ed.
2. Schildt H., *C:* The Complete Reference, Tata Mcgraw Hill, (2000), 4th ed.
3. Stroustrup B., The C++ Programming Language, Addison Wesley (2000), 3rd ed.
4. Kanetkar Y., Let Us C, BPB Publications(2012),13th ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Learn the implementation of simple 'C' program, data types and operators and Console I/O function.
CLO2	Learn the Implementation of decision control statements, loop control statements and case control structures.
CLO3	Understand the declaration and implementation of arrays, pointers and functions.
CLO4	Learn the structures declaration, initialization and implementation.
CLO5	Understand the file operations, Character I/O, String I/O, File pointers and importance of pre-processor directives

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessional (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA103 COMPUTER ORGANIZATION AND ARCHITECTURE				
	L	T	P	Cr
	3	0	2	4

Course Objective: Focus is on the architecture and organization of the basic computer modules *viz* control unit, central processing unit, input-output organization and memory unit.

Basics of Computer Architecture: Codes, Number System, Logic gates, Flip flops, Registers, Counters, Multiplexer, Demultiplexer, Decoder, Encoder etc.

Register Transfer and Micro operations: Register transfer Language, Register transfer, Bus & memory transfer, Logic micro operations, Shift micro operation.

Basic Computer Organization: Instruction codes, Computer instructions, Timing & control, Instruction Cycles, Memory reference instruction, Input/output & Interrupts, Complete computer description & design of basic computer.

ARM Processor Fundamentals: ARM core data flow model, Architecture, ARM General Purpose Register set, Exceptions, Interrupts, Vector Table, ARM processors family.

Central Processing Unit: General register organization, Stack organization, Instruction format, Data transfer & manipulation, Program control, RISC, CISC.

Computer Arithmetic: Addition & subtraction, Multiplication Algorithms, Division algorithms.

Input-Output Organization: Peripheral devices, I/O interface Data transfer schemes, Program control, Interrupt, DMA transfer, I/O processor.

Memory Unit: Memory hierarchy, Processor vs. memory speed, High-speed memories, Cache memory, Associative memory, Interleave, Virtual memory, Memory management.

Introduction to Parallel Processing: Pipelining, Characteristics of multiprocessors, Interconnection structures, Interprocessor arbitration, Interprocessor communication & synchronization.

Laboratory Work: Installing software development toolkit for ARM processor-based microcontrollers, Assembly language programming for ARM processors.

Recommended Books:

1. Mano, Morris M., Computer System Architecture, Prentice Hall (1992).
2. Hayes, J.P., Computer Architecture and Organization, McGraw Hill (1998).
3. Hennessy, J.L., Patterson, D.A, and Goldberg, D., Computer Architecture: A Quantitative Approach, Pearson Education Asia (2006).
4. Leigh, W.E. and Ali, D.L., System Architecture: software and hardware concepts, South Wester Publishing Co. (2000).

COURSE LEARNING OUTCOMES (CLOs):

On completion of this course, the students will be able to

1. Illustrate various elementary concepts of computer architecture including syntax of register transfer language, micro operations, instruction cycle, and control unit.
2. Describe the design of basic computer with instruction formats & addressing modes.
3. Explore various memory management techniques and algorithms for performing addition, subtraction and division etc.
4. Interpret the concepts of pipelining, multiprocessors, and inter processor communication.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	30
2.	EST	45
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	25

PCA104 WEB DESIGNING

L	T	P	Cr
2	0	2	3.0

Course objective: To give a brief introduction to the open source technology. Through interactive sessions of this course is to make student able to understand the basic directory, file structure of Linux, basic database structure and design concepts. This course describes the fundamental principles and terms of web application development using Linux, Apache, MySQL and PHP (LAMP). This course explores how to use open source technologies in the LAMP stack to design and develop dynamic interactive data-driven web applications.

Introduction: Open Source Definition, Free Software vs. Open Source Software, Public Domain Software, Open Source History, Initiatives, Principle and Methodologies. Open Standards.

Linux Administration and Setting Environment: Configuring the bash shell, Finding and processing files, Managing users, groups and permissions, Investigating and managing processes, Essential system administration tools, Installing and configuring apache web server (Linux), Installing PHP (Linux), Editing httpd.conf, Setting up access privileges.

Open Source Web Technologies: Two Tier and Three Tier Web based Application Architecture. LAMP Terminologies, Advantages. Apache, Web server conceptual working, Web browser, HTTP, Installation and Configuration, httpd.conf file, Logging, Security, Running a website, MySQL, Database management system, ER diagram, Relational database, Installation, Configuration, Administration, Common SQL queries. PHP**,** Dynamic content, Server side scripting, Installation, Configuration, Administration, Language syntax, Built-in functions, PHP and MySQL connectivity.

Programming on PHP and JavaScript:
JavaScript: JavaScript variables, control structures, functions, arrays and objects. Cascading Style Sheets, Client Side Scripting – Java Script, PHP: Form processing and business logic, stream processing and regular expressions, viewing client/server environment variables, connecting to database and handling of cookies. SQL, Accessing databases with PHP.

Laboratory work: Installation and setting up of LAMP environment, Editing httpd.conf and setting up access privileges, Creating programs using PHP, Create database connectivity with SQL database, Creating a form for various operation SQL queries using PHP, Implementing cookies and session.

Recommended Books:

1. Rosebrock E.,FilsonE.,Setting Up LAMP: Getting Linux, Apache, MySQL, and PHP Working Together, SYBEX Inc.
2. Jason Gerner, Elizabeth Naramore, Morgan L. Owens, Matt Warden, Professional LAMP Linux, Apache, MySql and PHP5 Web development , Wiley (2006).
3. James Lee, Brent Ware, Open Source Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP, Pearson Education (2003).
4. Eric Filson, Erick Rosebrock, Setting up LAMP: Getting Linux, Apache, MySQL, and PHP Working Together, SyBex (2004).

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Learn about the Open Source Operating system and its distributions like Fedora, Google chrome OS, Ubuntu.
CLO2	Develop static and dynamic web pages using different functionalities of HTML, DHTML
CLO3	Develop dynamic web pages using PHP concepts (Linux Apache MySql and PHP)
CLO4	Write client side validation using Java script.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	20
2.	EST	40
3.	Sessional (include Projects / Quizzes and Lab Evaluations)	40

PCA105 STATISTICAL AND NUMERICAL METHODS				
	L	T	P	Cr
	3	1	2	4.5

Course Objectives: The main objective of this study is to understand about the generation of random numbers, probability distribution and its properties. This course discusses about the concept of various discrete and continuous probability distribution for solving various day-to-day life problems. Students will also learn about the floating – point representation of number, the error and its occurrence in numerical computation which can lead to catastrophic cancellation. They will also able to understand interpolating and extrapolation.

Analysis of Statistical Data: Frequency distribution; Measure of central tendency and dispersion.

Random Variables and probability distributions: Basic concepts of probability and its properties; Additive and multiplicative theorem of probability; Conditional probability and independent events; Random variable, Notion of sample space; distribution functions; Mathematical expectation, Binomial, Poisson, Rectangular, Exponential and Normal distributions.

Random Number Generation: Basic concepts in random number generation; Method for generating random numbers and their efficiency test; Methods for generating random numbers for probability distributions

Sampling distributions: Notion of random sample and sampling distributions; Parameter and statistics; Standard error; Chi-square, t, F distributions; Basic ideas of testing of hypothesis; Testing of significance based on normal, Chi-square, t and F distributions; Analysis of variance, One way ANOVA and two way ANOVA with fixed effect; interval estimation.

Floating-Point Numbers: Floating-point representation; Rounding, Chopping; Error analysis; Condition and Instability.

Non-Linear Equations: Bisection, Secant, Fixed-point iteration and Newton – Raphson methods; Order of convergence.

Linear Systems of equations: Gauss Elimination and LU- decomposition methods; Jacobi and Gauss-Seidel methods.

Interpolation: Newton form of polynomials; Finite differences, Newton's Forward, Lagrange and Newton's divided difference interpolation formula with error analysis; Introduction to Spline.

Principle of least Square: Curve fitting; correlation and regression coefficients (two variables only); Rank correlation.

Laboratory Work: Implementation of statistical and numerical techniques using C/C++/R including Program to obtain frequency charts for large data set and fitting a distribution; Generation of Random Numbers for some distributions; Hypothesis of Testing; Bisection and Newton-Raphson methods; Solve system of linear equations using Gass elimination and Gauss-Seidel methods; Interpolation by Lagrange and Newton's divided difference methods; Regression analysis using least square approximation; Correlation analysis.

Recommended Books:

1. V. K. Rohtagi, A.K. Md. Saleh, An Introduction to Probability and Statistics, Wiley India Pvt. Ltd. (2014).
2. Meyer P. L., Introductory Probability and Statistical Applications, Oxford and IBH (2008).
3. Conte, S. D. and Boor, C. D., Elementary Numerical Analysis: An Algorithmic approach, 3rd Ed., Tata McGraw Hill, New York (2006).

4. Jain, M. K., Iyengar, S. R. K. and Jain, R. K., Numerical Methods for Scientific and Engineering Computation, New Age International Publishers (2008).

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand the concept of random variable and various discrete and continuous distribution.
CLO2	Analyze the different samples of data at different level of significance using various hypothesis testing.
CLO3	Understand error, source of error and its effect on any numerical computation and also 14nalyse14g the efficiency of any numerical algorithm.
CLO4	Learn how to obtain numerical solution of nonlinear equations and linear simultaneous equations.
CLO5	Understand the methods to construct interpolating polynomials with practical exposure

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessional (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA201 OPTIMIZATION TECHNIQUES				
	L	T	P	Cr
	3	0	2	4.0

Course objective: This course is designed to enable the students to understand the basics and application of optimization techniques. Students will learn some of the design principles for writing good programs.

Scope of Operations Research: Introduction to linear and non-linear programming formulation of different models.

Linear Programming: Geometry of linear programming, Graphical method, Linear programming (LP) in standard form, Solution of LP by simplex method, Exceptional cases in LP, Duality theory, Dual simplex method, Sensitivity analysis.

Integer Programming: Branch and bound technique.

Transportation and Assignment Problem: Initial basic feasible solutions of balanced and unbalanced transportation/assignment problems, Optimal solutions.

Project Management: Construction of networks, Network computations, Floats (free floats and total floats), Critical path method (CPM), Crashing.

Game Theory: Two person zero-sum game, Game with mixed strategies, Graphical method and solution by linear programming.

Recommended Books:

1. Chandra S., Jayadeva, Mehra A., Numerical Optimization and Applications, Narosa Publishing House, (2013).
2. Taha A. H., Operations Research-An Introduction, Pearson (2013) 9th Edition.
3. Pant C. J., Introduction to optimization: Operations Research, Jain Brothers (2004) 7th Edition.
4. Bazaarra S. M., Jarvis J. J. and Shirali D. H., Linear Programming and Network flows, John Wiley and Sons (2010) 4th Edition. 3. Taha A. H., Operations Research-An Introduction, Pearson

COURSE LEARNING OUTCOMES (CLOs): On completion of the course, the student will be able to

CLO1	Formulate and solve linear programming problems.
CLO2	Solve the transportation and assignment problems
CLO3	Solve the Project Management problems using CPM
CLO4	Solve two person zero-sum games

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	25
2	EST	40
3	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA202 GRAPH THEORY AND APPLICATIONS				
	L	T	P	Cr
	3	1	0	3.5

Course objective: This course is designed to explore computing and to show students the art of computer programming. Students will learn some of the design principles for writing good programs.

Introduction: Graphs, Sub-graphs, Regular graph, Adjacency and incidence matrices, Finite and infinite graph, Incidence and degree, Isolated vertex, Pendent vertex and null graph, Turan's theorem.

Paths and Circuits: Isomorphism, Walk, Cycle, Paths and circuits, Simple and proper circuit, Connected and disconnected graph, Euler graphs, Operations on graphs, Hamiltonian paths and circuits, Bipartite graph, Berge theorem, Hall's theorem, Edge connectivity, Blocks, Menger's theorem.

Trees and Fundamental Circuits: Trees, Properties of tree, Pendant vertices in a tree, Distance and centers in a tree, Spanning tree, Cayley's Formula, Minimal spanning tree, Prim and Kruskal's algorithm, Matrix Tree theorem, Dijkstra's Shortest Path Algorithm, Floyd-Warshall algorithm, Huffman's Coding Algorithm, Depth-first and breath first algorithm.

Cuts sets and cut-vertices: Cut sets, Properties of cutset, all cut sets in a graph, 1-isomorphism, 2-isomorphism.

Planar graph and dual graphs: Planar graphs, Homoeomorphic graph, Kuratowski's Two Graphs, Different representation of a planar graph, Tutte's f-factor theorem, Detection of planarity, Geometric dual, Combinatorial dual.

Coloring, Covering and Partitioning: Chromatic number, Chromatic Partitioning, Chromatic Polynomial, Covering, Four colour conjecture, Five-colour theorem, Dirac Theorem, Brooks theorem, Vizing theorem.

Directed Graphs: Directed graph, Diagraph and binary relations, Directed Paths, Euler diagraphs, Acyclic digraphs, Topological sorting, Warshall's algorithm, Bellman-Ford algorithm, Ramsey theorems.

Application of Graphs: Study of Konigsberg bridge problem, Travelling-salesman problem, Utilities problem, Electrical network problem, Seating problem, Use of graph in sequential switching networks, Graphs in coding theory, Graphs in computer programming, Flow graph notation, Test case generation using graphs, Job sequencing problem, Graph coloring in scheduling of examinations.

Recommended Books:

1. Deo Narsingh, Graph Theory with applications to Engineering and Computer Science, Prentice-Hall of India (2007).
2. Parmenter Michael M., Goodaire Edgar G., Discrete Mathematics with Graph Theory, Prentice-Hall of India (2007).
3. Rosen, Kenneth H. Discrete Mathematics and its Applications, Tata Mcgraw-Hill (2003).
4. Bondy, J.A. Murty U.S.R., Graph theory and Applications, North Holland Publications (1995).

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand different types of graphs and their relevance to computer science problems
CLO2	Solve different problems related to trees, circuits and paths
CLO3	Analyse problems related to partitioning, coloring and covering
CLO4	Understand issues related to directed, undirected Graphs
CLO5	Discuss and correlate various Graph applications with computer science and applications

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	25
2	EST	40
3	Sessionals (May include Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA203 OBJECT ORIENTED PROGRAMMING				
	L	T	P	Cr
	3	0	2	4.0

Course objective: Understand fundamentals of object oriented programming in Java. To help students understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.

Introduction to Java: History and Evolution of Java, Java vs. other popular languages, Java programming environment, Applications of Java, Fundamental Programming structures: Data types-Primitive Types, Floating-Point Types, Literals, Variables, Type Conversion and Casting, Operators: Arithmetic, Bitwise, Relational, Conditional and Unary, Dot, New and instanceof operator; Precedence and Associativity of Operators Expressions, Statements and Blocks, Control flow statements- Selection, Iteration and Jump Statements, Arrays and Strings.

Object Oriented Programming Concepts in Java: Objects and Classes, declaring Objects, Constructors, This keyword, Method Overloading, Constructor Overloading, Static methods, Nested classes, Garbage collection and finalize() method.

Inheritance: Defining, Applying and Implementing Interfaces; Method Overriding, Super and Final keywords, Polymorphism, Generics, Defining, finding and importing Packages, Exceptions Handling with try, catch, throw, throws and finally keywords, Wrapper classes.

I/O and Threads: Binary I/O, File Handling: File class, Reading/Writing Bytes and objects. Thread: Model, Creation, Lifecycle, Determining when a thread ends: isAlive() and Join() Methods, Synchronization, Priority, Deadlock, Communication using wait() and notifyAll().

Building GUI in Java: Java.awt Package, Layout Managers, Swing component Hierarchy, and adding Swing Components, Painting, using Images, Performing animations. Borders, Icons, Event Handling Process, Steps required to Setup Event Handling for a GUI Component, Types of Events, and Implementation of Listeners for event handling.

Laboratory work: Main focus is on implementing basic concepts of object oriented programming and to enhance programming skills to solve specific problems.

Recommended books:

1. Deitel H. and Deitel P., JAVA – How to Program, Pearson Education, (2003).
2. Internet & World Wide Programming, Deitel, Deitel& Nieto, Pearson Education,(2000).
3. Naughton P., Schildt H., JAVA2 – The Complete Reference, Tata McGraw Hill, (2002),5th ed.
4. Greenlaw R. and Hepp G.- Fundamentals of the Internet and the World Wide Web, TMH,(2001).

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand the basics of Java Environment and program structure.
CLO2	Learn the implementation of data types, decision and looping statements.
CLO3	Understand concepts of inheritance and multi-threading in Java.
CLO4	Understand Input and output handling from console, and files in Java.
CLO5	Learn creation of frames, windows, containers, GUI components in Java and event handling for building GUI.

Evaluation Scheme:

Sr.No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA204 PROGRAMMING LANGUAGES				
	L	T	P	Cr
	3	0	2	4.0

Course Objective: Understanding the concepts of object oriented programming, functional programming, event-driven and reactive programming, formal semantics and language pragmatics.

Object-Oriented Programming: Object-oriented design, Definition of classes: fields, methods, and constructors, Subclasses, inheritance, and method overriding, Abstract Classes, Dynamic dispatch: definition of method-call, Object-oriented idioms for encapsulation, Using collection classes, iterators, and other common library components

Functional Programming: Processing structured data (e.g., trees) via functions with cases for each data variant, First-class functions (taking, returning, and storing functions), defining higher-order operations on aggregates, especially map, reduce/fold, and filter

Event-Driven and Reactive Programming: Events and event handlers, Canonical uses such as GUIs, mobile devices, robots, servers, Externally-generated events and program generated events, Separation of model, view, and controller

Formal Semantics: Syntax vs. Semantics, Lambda Calculus and Approaches to semantics: Operational, Denotational, Axiomatic, Proofs by induction over language semantics Formal definitions and proofs for type systems, Parametricity, Using formal semantics for systems modelling.

Language Pragmatics: Principles of language design, Evaluation order, precedence, and associativity, Eager vs. delayed evaluation, Defining control and iteration constructs, External calls and system libraries

Laboratory work: Implementing object oriented concepts, Functional Programming concepts, Event Driven Concepts, Reactive Programming concepts and formal semantic concepts.

Recommended Books:

1. Sethi, Ravi, Programming Languages-Concepts and Constructs, Addison-Wesley, 1996, 2nd ed.
2. Tucker A.B. and Noonan R., Programming Languages-Principles and Paradigms, McGraw Hill, 2007, 2nd ed.
3. Sebesta R.W., Concepts of Programming Languages, Addison Wesley, 2012, 10th ed.
4. Gabbrielli M, Martini S. Programming languages: principles and paradigms. Springer Science & Business Media; 2010

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Compare and contrast different language paradigms and understand object-oriented mechanisms such as abstraction and encapsulation.
CLO2	Understand the basic algorithms that avoid assigning to mutable state or considering reference equality.
CLO3	Understand the event handlers for use in reactive systems, such as GUIs and describe an interactive system in terms of a model, a view, and a controller.
CLO4	Use the formal semantics to build a formal model of a software system other than a programming language.

CLO5	Implement the program whose result can differ under different rules for evaluation order.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA205 ORGANIZATIONAL BEHAVIOUR				
	L	T	P	Cr
	3	0	0	3.0

Course Objective: Understanding psychology theories and research at individual, group and organisational levels; organisational behaviour and management practices by examining psychological principles and to facilitate a critical evaluation of organisational practices and their impact on work behaviours, attitudes and performance.

Introduction to Organization Behaviour: Introduction to organization, organization and managers, manager' roles and skills, behaviour at work, introduction to organization behaviour, major behavioural science disciplines contributing to OB, challenges and opportunities managers have in applying OB concepts, OB model (including motivation models) and levels of OB model

Individual behaviour: Introduction to individual behaviour, values, attitudes, job satisfaction, personality, perception and individual decision making, learning, motivation at work, managing emotions and stress (Meaning-Definition Stress and job performance relationship Approaches to stress management (Coping with stress)

Interpersonal behaviour: Interpersonal Behaviour, Johari Window, Transactional Analysis – ego states, types of transactions, life positions, applications of T.A., managerial interpersonal styles.

Group behaviour: Introduction to group behaviour, foundations of group behaviour, concept of group and group dynamics, types of groups, formal and informal groups, theories of group formation, group norms, group cohesiveness, group decision making, inter group behaviour, concept of team vs. group, types of teams, building and managing effective teams, leadership theories and styles, power and politics, conflict and negotiation.

Organisational behaviour: Foundations of organization structure, organization design, organization culture, organization change, managing across cultures, human resource management policies and practices, diversity at work.

Recommended Books:

1. Robbins, S. P/ Judge, T. A/ Sanghi, S., Organizational Behavior, Pearson Publication
2. Aswathappa, K., Organisational Behaviour– Text and Problem, Himalaya Publication
3. Pardeshi, P. C., Organizational Behaviour & Principles & Practice Of Management, Nirali publication

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	To understand the main theories of Organisational Behaviour
CLO2	To be able to analyse how these theories and empirical evidence can help to understand contemporary organisational issues
CLO3	To apply theories to practical problems in organisations in a critical manner

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA301 COMPUTER NETWORKS				
	L	T	P	Cr
	2	0	2	3.0

Course Objective: The subject will introduce the fundamentals of computer networks, related concepts and theories.

Introduction: Organization of the Internet (Internet Service Providers, Content Providers, etc.), Network criteria, Categories of networks, Network performance and Transmission Impairments. Physical pieces of a network, including hosts, routers, switches, bridges, hubs, ISPs, wireless, LAN, access point, and firewalls. OSI Model and TCP/IP Protocol Suite. Layering principles (encapsulation, multiplexing), Switching techniques. (e.g., circuit, packet).

Networked Applications : Naming and address schemes (DNS, IP addresses, Uniform Resource Identifiers, etc.), Distributed applications (client/server, peer-to-peer, cloud, etc.), HTTP as an application layer protocol, Electronic mail, File transfer, Remote login-introduction to protocol specification.

Reliable Data Delivery: Error control (retransmission techniques, timers), Flow control (Acknowledgements, sliding window), Performance issues (pipelining), Process-to-Process Delivery: UDP, TCP and SCTP. Multiplexing with TCP and UDP.

Routing and Forwarding: Routing versus forwarding, Static and dynamic routing, Unicast and Multicast Routing. Distance-Vector, Link-State, Shortest path computation, Dijkstra's algorithm, Network Layer Protocols (IP, ICMP), IP addressing, Address binding with ARP, Scalability issues (hierarchical addressing)

Local Area Networks: LAN topologies: Bus topology, Ring topology, Token passing rings, FDDI, Star topologies, Asynchronous transfer mode. Ethernet, IEEE standards 802.3, 802.5. Multiple Access Problem, Common approaches to multiple accesses (exponential-back off, TDM, etc). Virtual circuit switching including frame relay, X.25, and ATM.

Resource Allocation: Need for resource allocation, Fixed allocation (TDM, FDM, WDM) versus dynamic allocation, End-to-end versus network assisted approaches , Principles of congestion control, Approaches to Congestion, Quality of service, Flow characteristics, Techniques to improve QoS.

Mobility: Principles of cellular networks, Wireless LANs: IEEE 802.11 and Bluetooth, Issues in supporting mobile nodes (home agents).

Laboratory Work: Designing conceptual networks using E-Draw, Visual Studio etc. and implementing topologies such as Bus, Ring, Star, Mesh and configuring Routers using Packet tracer or GNS3 platform.

Recommended Books

1. Forouzan, B.A., Data communication and Networking,McGraw Hill , (2006) , 4th ed.
2. Tanenbaum , A.S.,Computer Networks,Prentice Hall,(2010) , 5th ed.
3. Kurose and Ross, Computer Networking: A Top Down Approach, Addison-Wesley, (2012), 6th ed.

4. Stallings, W.,Computer Networking with Internet Protocols and Tech,Prentice Hall of India (2010), 9th ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Conceptualize and explain the functionality of the different layers within a network architecture
CLO2	Analyze the requirements for a given organizational structure and select the most appropriate networking architecture and technologies, subnetting and routing mechanism.
CLO3	Demonstrate the operation of various routing protocols and their performance analysis.
CLO4	Illustrate design and implementation of datalink, transport and network layer protocols within a simulated/real networking environment.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA302 OPERATING SYSTEMS				
	L	T	P	Cr
	2	0	2	3.0

Course Objective: Role and purpose of the operating system, Functionality of a typical operating system, managing atomic access to OS objects.

Operating System Principles: Structuring methods (monolithic, layered, modular, microkernel models), processes, and resources, Application bases, Concepts of APIs, Device organization, interrupts: methods and implementations, Concept of user/system state and protection, transition to kernel mode.

Concurrency: Implementing synchronization primitives, Multiprocessor issues (spin locks, 27nalyse27g27y).

Scheduling and Dispatch: Dispatching and context switching, Preemptive and non-preemptive scheduling, Schedulers and policies, Processes and threads

Memory Management: Review of physical memory and memory management hardware, Working sets and thrashing, Caching, Paging and virtual memory, Virtual file systems.

File Systems: Files: data, metadata, operations, organization, buffering, sequential, nonsequential, Directories: contents and structure, Naming, searching, Journaling and log-structured file systems.

Deadlock: Introduction, Analysis of conditions, Prevention & avoidance, Detection & recovery.

Security and Protection: Overview of system security, Security methods and devices, Protection, access control, and authentication.

Virtual Machines: Types of virtualization (including Hardware/Software, OS, Server, Service, Network).

Device Management: Characteristics of serial and parallel devices, Buffering strategies, Direct memory access, Disk structure, Disk scheduling algorithms.

Laboratory Work: To explore different operating systems like Linux, Windows etc. To implement main algorithms related to key concepts in the operating systems using a high level language.

Recommended Books

1. Silberschatz, A., Galvin, P.B. and Gagne, G., Operating System Concepts, John Wiley (2013), 9th Ed.
2. Stallings, Willam, Operating Systems Internals and Design Principles, Prentice Hall, (2014), 7th Ed.
3. Dhamdhere, D.M., Operating Systems: A Concept Based Approach, McGraw Hill , (2008), 2nd Ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Comprehend the concepts related to basics of operating System.
CLO2	Exemplify protection and security in operating system.
CLO3	Simulate the various process management algorithms and specifications for simple as well as for concurrent processes.
CLO4	Grasp the concepts related to memory management.
CLO5	Implement the concepts related to file organization and handling in system.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA303 SOFTWARE ENGINEERING				
	L	T	P	Cr
	3	0	2	4.0

Course Objectives: To apply principles of software development and evolution. To specify, abstract, verify, validate, plan, develop and manage large software and learn emerging trends in software engineering.

Introduction: Introduction to Software Engineering, importance of Software, The Software Evolution, Software Characteristics, Software Applications, Software Crisis: Problem and Causes.

Software Processes: Software process models (Waterfall, Incremental, and Evolutionary process models and Agile), software quality concepts, process improvement, software process capability maturity models, Personal Software process and Team Software Process, Overview of Agile Process.

Software configuration management: CASE and its Scope, CASE support in Software Life Cycle, Documentation Support, Architecture of CASE Environment.

Requirements Engineering: Problem Analysis, Requirement elicitation and Validation, Requirements modeling: Scenarios, Information and analysis classes, flow and behavioral modeling, documenting Software Requirement Specification (SRS).

Software Design: System design principles: levels of abstraction (architectural and detailed design), separation of concerns, information hiding, coupling and cohesion, Structured design (top-down functional decomposition), object-oriented design, event driven design, component-level design, data-structured centered, aspect oriented design , function oriented, service oriented, Design patterns.

Software Construction: Coding Practices: Techniques, mechanisms for building quality programs including Secure Coding Practices, Integration Strategies, Internal Documentation, Verification.

Software Verification and Validation: Levels of Testing, Functional Testing, Structural Testing, Test Plan, Test Case Specification, Software Testing Strategies, Verification & Validation, Unit, Integration Testing, Top Down and Bottom Up Integration Testing, Alpha & Beta Testing, White box and black box testing techniques, System Testing and Debugging.

Software Quality Assurance: Software Quality Control and Quality Assurance, ISO 9000 Certification for Software Industry, SEI CMM and Comparison of ISO & SEICMM.

Software Evolution: Software development in the context of large, pre-existing code bases, Software evolution, Characteristics of maintainable software, Software Reengineering, Software reuse.

Formal Methods: Role of formal specification and analysis techniques in the software development cycle. Formal approaches to Software Modeling and Analysis.

Technical Metrics for Software: A Framework for Technical Software Metrics, Metrics for the Analysis Model, Metrics for Design Model, Metrics for Source Code, Testing and Maintenance.

Laboratory Work: Implementation of Software Engineering concepts using tools like Rational Suite etc. Exposure to CASE tools like Rational Software suit, Turbo Analyst, Silk Suite, Z formal specification language.

Recommended Books:

1. S. Pressman R., Software Engineering, McGraw Hill International Ed, 2014 8th ed.
2. Sommerville I., Software Engineering, Addison-Wesley Publishing, (2010) 9th ed.
3. Jalote P., An integrated Approach to Software Engineering, Narosa (2005).
4. James F. Peter, Software Engineering – An Engineering Approach, John Wiley (2004).

COURSE LEARNING OUTCOMES (CLOs):

On completion of this course, students will be able to

CLO1	Comprehend the basics of software project management in order to manage and deliver competent product.
CLO2	Develop software projects based on current technology, by managing resources economically and keeping ethical values.
CLO3	Formulate appropriate testing strategy for the given software system.
CLO4	Comprehend the principles, processes and knowledge regions for software project management.
CLO5	Comprehend and apply formal methods for software 30nalyse30g and analysis.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA304 DATA STRUCTURES

L	T	P	Cr
3	0	2	4.0

Course Objectives: To become familiar with different types of Data structures and their applications. To become familiar with different types of algorithmic techniques and strategies.

Linear Data Structures: Arrays, Linked lists, Strategies for choosing the appropriate data structure, Abstract data types and their implementation: Stacks, Queues, Priority queues, Sets, Maps.

Basic Analysis: Differences among best, expected, and worst case behaviours of an algorithm, Asymptotic analysis of upper and expected complexity bounds, Big O notation: formal definition and use, Little o, big omega and big theta notation, Complexity classes, such as constant, logarithmic, linear, quadratic, and exponential, Time and space trade-offs in algorithms, Recurrence relations, Analysis of iterative and recursive algorithms.

Searching and Sorting: Linear Search, Binary Search, Bubble Sort, Selection Sort, Insertion Sort, Shell Sort, Quick Sort, Heap Sort, Merge Sort, Counting Sort.

Algorithmic Strategies with examples and problem solving: Brute-force algorithms with examples, Greedy algorithms with examples, Divide-and-conquer algorithms with examples, Recursive backtracking, Dynamic Programming with examples, Branch-and-bound with examples, Heuristics, Reduction: transform-and-conquer with examples.

Non-Linear Data Structures: Hash tables, including strategies for avoiding and resolving collisions, Binary search trees, Common operations on binary search trees such as select min, max, predecessor, successor, insert, delete, iterate over tree, Graphs and graph algorithms, Representations of graphs, Depth- and breadth-first traversals, Heaps, Graphs and graph algorithms, Shortest-path algorithms (Dijkstra and Floyd), Minimum spanning tree (Prim and Kruskal).

Laboratory Work: Implementation of Arrays, Recursion, Stacks, Queues, Linked-Lists, Binary trees, Sorting techniques, Searching techniques. Implementation of all the algorithmic techniques.

Recommended Books:

1. Cormen, Leiserson, Rivest, & Stein, Introduction to Algorithms, The MIT Press (2009) 3rd Edition.
2. Langsam, Augenstein, & Tenenbaum, Data Structures Using C and C++, Pearson Education India (2015), 2nd edition.
3. Tremblay & Sorenson, An Introduction to Data Structures with Application, McGraw Hill Education (2017), 2nd edition.
4. Sedgewick & Wayne, Algorithms, Addison-Wesley Professional (2011), 4th Edition.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Implement basic data structures and solve problems using fundamental algorithms.
CLO2	Implement various searching and sorting techniques.
CLO3	Analyse the complexity of algorithms, to provide justification for that selection, and to implement the algorithm in a particular context.
CLO4	Analyse, evaluate and choose appropriate data structure and algorithmic technique to solve real-world problems.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA305 DATABASE MANAGEMENT SYSTEM

L	T	P	Cr
3	0	2	4.0

Course objective: Emphasis is on the need of information systems. Main focus is on E-R diagrams, relational database, concepts of normalization and de-normalization and SQL commands.

Introduction: Data, data processing requirement, desirable characteristics of an ideal data processing system, traditional file based system, its drawback, concept of data dependency, Definition of database, database management system, 3-schema architecture, database terminology, benefits of DBMS, Database development process - conceptual data modelling, logical database design, physical database design, database implementation, database maintenance.

Database Analysis: Conceptual data modelling using E-R data model -entities, attributes, relationships, generalization, specialization, specifying constraints.

Relational Data Model, Relational Model Constraints Relational Algebra: Set theoretic operations: Union, Intersection, difference, Cartesian product, Selection, Projection, Types of Join: Theta, Inner, Left Outer, Right Outer, Natural & Division, Relational calculus: Tuple and Domain relational calculus.

SQL: A Relational Database Language, SQL data types, Constraints: Primary key, Foreign key, check, unique, not null. Table level and column level specifications, DDL, DML and DCL Queries in SQL, Data retrieval statements and sub-queries, various Data type functions: Numeric, character and date data type functions. Views, Sequences, Synonyms, Indexing.

PL/SQL: Introduction to PL/SQL, data types, Cursors, Triggers, Functions, Procedures, Packages and Exception Handling.

Relational Data Base Design: Function Dependencies & Normalization for Relational Databases. Closure, Loss less join & Dependency preserving decomposition. Normal forms: INF, 2NF, 3NF, BCNF, 4NF and 5NF.

Concurrency Control & Recovery Techniques: Transactions, Concurrency Control Techniques: Serialiazability, Locking Techniques, Time stamping Protocols, Multiple Granularity of Data items, Recovery Techniques, Recovery concepts, Database backup and recovery from catastrophic failures

Laboratory work: Lab work will be based on implementation database concepts using SQL and PL/SQL.

Recommended Books:

1. H. F. Korth & Silverschatz, A., Database System Concepts, Tata McGraw Hill, 2010, 6th ed.
2. Elmasri & Navathe, Fundamentals of Database Systems, Addison-Wesley, 2011, 6th ed.
3. Hoffer, Prescott, Mcfadden, Modern Database Management, Paperback International, 2012, 11th ed.
4. Martin Gruber, Understanding SQL, BPB Publication, 1994, Revised ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand Information Systems as socio-technical systems, Its need and advantages as compared to traditional file based systems.
CLO2	Understand the architecture of DBMS, conceptual data, logical database design and physical database design.
CLO3	Perform Database Analysis using E-R data model by identifying entities, attributes, relationships, generalization and specialization.
CLO4	Perform Relational Database Design process with Normalization and De-normalization of data and understand the need of Indexing of Data and types of indexes and its implementation.
CLO5	Learn Database Implementation with SQL and PL/SQL.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	25
2	EST	40
3	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA401 THEORY OF COMPUTATION				
	L	T	P	Cr
	3	1	0	3.5

Course Objective: This course introduces basic theory of computer science and formal methods of computation. The course exposes students to the computability theory, as well as to the complexity theory.

Regular Languages: Alphabets, Language, Regular Expression, Definitions of Finite State Machine, Transition Graphs, Deterministic & Non-deterministic Finite State Machines, Regular Grammar, Thompson's Construction to Convert Regular Expression to NDFA & Subset Algorithm to convert NDFA to DFA, Various recent development in the Conversion of Regular Expression to NFA, Minimization of DFA, Finite State Machine with output-Moore machine and Melay Machine, Conversion of Moore machine to Melay Machine & Vice-Versa.

Properties of Regular languages: Conversion of DFA to Regular Expression, Pumping Lemma, Properties and Limitations of Finite state machine, Decision properties of Regular Languages, Application of Finite Automata.

Context Free Grammar and Push Down Automata : Context Free Grammar, Derivation tree and Ambiguity, Application of Context free Grammars, Chomsky and Greibach Normal form, Properties of context free grammar, CKY Algorithm, Decidable properties of Context free Grammar, Pumping Lemma for Context free grammar, Push down Stack Machine, Design of Deterministic and Non-deterministic Push-down stack.

Turing Machine: Turing machine definition and design of Turing Machine, Church-Turing Thesis, Variations of Turing Machines, combining Turing machine, Universal Turing Machine, Post Machine, Chomsky Hierarchy, Post correspondence problem.

Uncomputability: Halting Problem, Turing enumerability, Turing Acceptability and Turing decidabilities, unsolvable problems about Turing machines, Rice's theorem.

Recommended Books:

1. Hopcroft J. E., Ullman J. D. and Motwani R., Introduction to Automata Theory, Languages and Computation, Pearson Education, 3rd Edition, (2006).
2. John C. Martin, Introduction to Languages and the Theory of Computation, McGraw-Hill Higher Education (2011).
3. Daniel A. Cohen, Introduction to Computer Theory, John Wiley and Sons, 2nd Ed, 1996
4. Michael Sipser, Introduction to the Theory of Computation, Thomson, 2nd Ed, 2007

COURSE LEARNING OUTCOMES (Cos)

CO1	Comprehend regular languages and finite automata and develop ability to provide the equivalence between regular expressions, NFAs, and DFAs.
CO2	Disambiguate context-free grammars by understanding the concepts of context-free languages and push-down automata.
CO3	Apply the concepts of recursive and recursively enumerable languages and design efficient Turing Machines.
CO4	Solve analytical problems in related areas of theory in computer science

Evaluation Scheme:

S.No.	Evaluation Elements	Weightage (%)
1.	MST	30
2.	EST	40
3.	Sessional (May include Assignments/Projects/Tutorials/Quizzes)	25

PCA402 COMPUTER GRAPHICS

L	T	P	Cr
3	0	2	4.0

Course Objective: Detailed study of computer graphics, 2 D and 3 D transformations, representations and visualization.

Fundamentals of Computer Graphics: Overview of Computer Graphics, Computer Graphics Applications and Software, , Video Display Devices, Random scan displays, Raster scan displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random-Scan Display Processor, LCD displays.

Graphics Primitives: Scan converting line, circle, ellipse, arcs & sectors, Boundary Fill & Flood Fill algorithm, Color Tables.

Transformations: Homogeneous Coordinates, 2D and 3D Scaling, Translation, rotation, shearing & reflection, Composite transformation, Window to View port transformation.
Clipping: Cohen Sutherland, Liang Barsky, Nicholl – Lee – Nicholl Line clipping algorithms, Sutherland Hodgeman, Weiler Atherton Polygon clipping algorithm.

Three Dimensional Object Representations: Parallel & Perspective projection, Vanishing Points,. Curved lines & Surfaces, Spline representations, Spline specifications, Bezier Curves & surfaces, B-spline curves & surfaces,

Basic Rendering: Rendering in nature, Polygonal representation, Affine and coordinate system transformations, Visibility and occlusion, depth buffering, Painter's algorithm, ray tracing, forward and backward rendering equations.

Visualization: Visualization of 2D/3D scalar fields: color mapping, isosurfaces. Direct volume data rendering: ray-casting, transfer functions, segmentation. Visualization of: Vector fields and flow data, Time-varying data, High-dimensional data: dimension reduction, parallel coordinates, Non-spatial data: multi-variate, tree/graph structured, text Perceptual and cognitive foundations, Evaluation of visualization methods, Applications of visualization.

Laboratory work: Lab work should be done in OpenGL. Covers all the basic drawing, filling, transformation and clipping algorithms.

Recommended Books:

1. Donald D Hearn, M. Pauline Baker, Computer Graphics C version, Pearson Education, 2nd ed.
2. James D. Foley, Andries van Dam, Steven K. Feiner and John F. Hughes, Computer Graphics: Principles & Practice in C, Addison Wesley Longman, 2nd ed.
3. Zhigang Xiang, Roy A Plastock, Computer Graphics, Schaums Outline, TMH, 2nd ed.
4. Donald D Hearn ,Computer Graphics with OpenGL, Pearson Education,,4th edition,2013
5. OpenGL Programming Guide: The Official Guide to Learning OpenGL, Dave Shreiner, Mason Woo, Jackie Neider, Tom Davis, 5th Edition, 2013
6. Alexandru C. Telea, Data Visualization: Principles and Practice, A. K. Peters Ltd., 2007.
7. Charles D. Hansen and Chris R. Johnson, Visualization Handbook, Elsevier, 2004.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Able to comprehend the concepts related to basics of computer graphics and visualization.
CLO2	Skilled to know and simulate the various graphics transformations and clipping techniques.
CLO3	Able to grasp the concepts related three dimensional object representations.
CLO4	To learn and program in OpenGL.
CLO5	Able to understand the methods, applications of 2D and 3D visualization.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA403 BIG DATA ANALYTICS AND BUSINESS INTELLIGENCE				
	L	T	P	Cr
	3	0	2	4.0

Course Objective: To have a basic understanding of most recent advancements in Big Data and using insights, statistical models, visualization techniques for its effective application in Business intelligence.

Introduction to Data Analytics: Data and Relations, Data Visualization, Correlation, Regression, Forecasting, Classification, Clustering.

Big Data Technology Landscape: Fundamentals of Big Data Types, Big data Technology Components, Big Data Architecture, Big Data Warehouses, Functional vs. Procedural Programming Models for Big Data.

Introduction to Business Intelligence: Business View of IT Applications, Digital Data, OLTP vs. OLAP, Why, What and How BI? , BI Framework and components, BI Project Life Cycle, Business Intelligence vs. Business Analytics.

Big Data Analytics: Big Data Analytics, Framework for Big Data Analysis, Approaches for Analysis of Big Data, ETL in Big Data, Introduction to Hadoop Ecosystem, HDFS, Map-Reduce Programming, Understanding Text Analytics and Big Data, Predictive analysis on Big Data, Role of Data analyst.

Business implementation of Big Data: Big Data Implementation, Big Data workflow, Operational Databases, Graph Databases in a Big Data Environment, Real-Time Data Streams and Complex Event Processing, Applying Big Data in a business scenario, Security and Governance for Big Data, Big Data on Cloud, Best practices in Big Data implementation, Latest trends in Big Data, Latest trends in Big Data, Big Data Computation, More on Big Data Storage, Big Data Computational Limitations.

Laboratory Work: Introduction to most recent advancements in Big Data technology along with their usage and implementation with relevant tools and technologies.

Recommended books:

1. Michael Minelli, Michele Chambers, Ambiga Dhiraj, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley CIO Series(2013),1st ed.
2. T. white, Hadoop: The Definitive Guide, O' Reilly Media (2012), 3rd ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Translate a business challenge into an analytics challenge and deploy a structured lifecycle approach to data science and big data analytics projects.
CLO2	Basic understanding of the usage of Big Data in present World.
CLO3	Analyse big data, create statistical models, and identify insights that can lead to actionable results.
CLO4	Select visualization techniques, communicate analytic insights to business sponsors, and others and explain how advanced analytics can be leveraged to create competitive advantage.
CLO5	Define and distinguish a data scientist from a traditional business intelligence analyst.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA404 ARTIFICIAL INTELLIGENCE

L	T	P	Cr
3	0	2	4.0

Course Objective: To be familiar with the applicability, strengths, and weaknesses of the basic knowledge representation, problem solving, and learning methods in solving particular engineering problems.

Fundamental Issues: Overview of AI problems, Examples of successful recent AI applications, Intelligent behaviour, The Turing test, approaches of AI.

Basic Search Strategies: Problem spaces (states, goals and operators), Problem solving by search, Factored representation (factoring state into variables), Uninformed search (breadth-first, depth-first, depth-first with iterative deepening), Heuristics and informed search (hill-climbing, generic best-first, A*), Space and time efficiency of search, Constraint satisfaction (backtracking and local search methods).

Advanced Search: Constructing search trees, Dynamic search space, Combinatorial explosion of search space, Stochastic search: Simulated annealing, Genetic algorithms, Monte-Carlo tree search, Implementation of A* search, Beam search, Minimax Search, Alpha-beta pruning.

Knowledge Representation: Propositional and predicate logic, Resolution in predicate logic, Question answering, Theorem proving, Semantic networks, Frames and scripts, conceptual graphs, conceptual dependencies.

Reasoning under Uncertainty: Review of basic probability, Random variables and probability distributions: Axioms of probability, Probabilistic inference, Bayes' Rule, Conditional Independence, Knowledge representations using Bayesian Networks, Exact inference and its complexity, Randomized sampling (Monte Carlo) methods (e.g. Gibbs sampling), Markov Networks, Relational probability models, Hidden Markov Models, Decision Theory Preferences and utility functions, Maximizing expected utility.

Agents: Reactive, Deliberative, Goal-driven, Utility-driven and learning agents.

Expert Systems: Architecture of an expert system, existing expert systems: MYCIN, RI. Expert system shells.

Laboratory work: Programming in C/C++/java: programs for Search algorithms- Depth first, breadth first, best first, hill climbing, Implementation of games: 8-puzzle, Tic-tac-toe using heuristic search, Designing expert system using logic in Prolog, implementing an intelligent agent.

Recommended Books

1. Rich E., Artificial Intelligence, Tata McGraw Hills (2009) 3rd ed.
2. George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education Asia (2009) 6th ed.
3. Patterson D.W, Introduction to AI and Expert Systems, Mc GrawHill 1998, 1st ed.
4. Russel S., Norvig P., Artificial intelligence : A Modern approach, prentice Hall 2014, 3rd edition.

COURSE LEARNING OUTCOMES (Cos)

CO1	Learn and understand applications of artificial intelligence and categorize various problem domains and search methods
CO2	Analyze basic and advanced search techniques including game playing, evolutionary search algorithms, and constraint satisfaction.
CO3	Learn knowledge representation using various forms in artificial intelligence and understand the importance of probability in knowledge representation for reasoning under uncertainty.
CO4	Learn to represent knowledge using Bayesian networks and drawing Hidden Markov Models.
CO5	Acquire knowledge about the architecture of an expert system.

Evaluation Scheme:

S.No.	Evaluation Elements	Weightage (%)
1.	MST	20
2.	EST	40
3.	Sessional (May include Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	40

PCA405 DESIGN AND ANALYSIS OF ALGORITHMS

L	T	P	Cr
3	0	2	4.0

Course Objectives: To understand and apply the algorithm analysis techniques. To critically analyse the efficiency of alternative algorithmic solutions for the same problem. To understand different algorithm design techniques. To understand the limitations of Algorithmic power.

Introduction: Fundamentals of Algorithmic Problem Solving – Important Problem Types – Fundamentals of the Analysis of Algorithm Efficiency – Analysis Framework – Asymptotic Notations and its properties – Mathematical analysis for Recursive and Non-recursive algorithms.

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

Divide and Conquer Methodology – Binary Search, Merge sort, Quick sort, Heap Sort, Multiplication of Large Integers, Closest-Pair and Convex, Hull Problems.

Dynamic programming– Principle of optimality, Coin changing problem, Computing a Binomial Coefficient, Floyd‘s algorithm, Multi stage graph, Optimal Binary Search Trees, Knapsack Problem and Memory functions. Greedy Technique, Container loading problem, Prim‘s algorithm and Kruskal’s Algorithm, 0/1 Knapsack problem, Optimal Merge pattern, Huffman Trees.

The Simplex Method, The Maximum-Flow Problem, Maximum Matching in Bipartite Graphs, Stable marriage Problem.

Problem Clauses: Lower Bound Arguments, P, NP, NP Complete and NP Hard Problems. Backtracking, n-Queen problem – Hamiltonian Circuit Problem – Subset Sum Problem. Branch and Bound – LIFO Search and FIFO search – Assignment problem – Knapsack Problem – Travelling Salesman Problem – Approximation Algorithms for NP-Hard Problems – Travelling Salesman problem – Knapsack problem.

Laboratory Work: It will be based on designing of the algorithms and their analysis.

Recommended Books:

1. Anany Levitin, ―Introduction to the Design and Analysis of Algorithms‖, Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Computer Algorithms/ C++, Second Edition, Universities Press, 2007.
3. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, ―Data Structures and Algorithms‖, Pearson Education, Reprint 2006.
4. Cormen, Leiserson, Rivest, & Stein, Introduction to Algorithms, The MIT Press (2009) 3rd Edition.
5. Sedgewick & Wayne, Algorithms, Addison-Wesley Professional (2011), 4th Edition.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Design algorithms for various computing problems
CLO2	Implement various searching and sorting techniques.
CLO3	Analyse the complexity of algorithms, to provide justification for that selection, and to implement the algorithm in a particular context.
CLO4	Analyse, evaluate and choose appropriate data structure and algorithmic technique to solve real-world problems.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA501: MOBILE APPLICATION DEVELOPMENT				
	L	T	P	Cr
	3	0	2	4.0

Course objective: This course is concerned with the development of applications on mobile and wireless computing platforms.

Introduction: Cost of Mobile Application Development, Importance of Mobile Strategies, Challenges, Myths, Third-Party Frameworks, Mobile Web Presence, Applications

Introduction to Mobility: Mobility Landscape, Mobile Platforms, Mobile apps development, Overview of Android Platform, Setting up the mobile apps development environment with emulator.

Building blocks: App user Interface Designing, Layout, User Interface elements, Draw-able, Menu, Activity states and lifecycle, Interaction among activities.

App functionality based user interface:Threads, Asynchronous task, Services-states and lifecycle, Notifications, Broadcast receivers, Telephony and SMS API.

Naïve Data Handling: On Device File I/O, Shared preferences, Mobile Databases such as SQLite and enterprise data access.

Sprucing up Mobile Apps: Graphics and animation-custom views, canvas, animation API multimedia-audio/video playback and record, location aware.

Testing Mobile apps: Debugging Apps, White and Black Box Testing and test automation of apps.

Creating Consumable Web Services for Mobile Devices: What is a Web Service, Web Services Languages (Formats), Creating an Example Web Service, Debugging Web Services

Mobile User Interface Design: Effective Use of Screen Real Estate, Understanding Mobile Information Design, Understanding Mobile Application Users, Understanding Mobile Platforms, Using the Tools of Mobile Interface Design.

Mobile Websites: Choosing a Mobile Web Option, Adaptive Mobile Websites, Dedicated Mobile Websites Mobile Web Apps with HTML5

Android: Android as Competition to itself, Connecting to the Google Play, Android Development Practices, Building an App in Android

iOS: IOS Project, Debugging iOS Apps, Objective-C Basics, Building the Derby App in IOS

Windows Phone 7: Windows Phone 7 Project, Building an App in Windows Phone 7, Distribution.

Laboratory work: To develop robust mobile applications and work on related tools and technologies. exploring the application development for different mobile platforms like Android, iPhone, Symbian

Recommended Books

1. Professional Mobile Application Development, Jeff Mcwherter, Scott Gowell, Wrox Publisher, 1st Ed. 2012
2. Sams Teach Yourself Android Application Development in 24 Hrs, Lauren Darcy and Shane Conder, 1sted.
3. Mobile Application Security, HimanshuDwivedi, Chris Clark, David Thiel, Tata McGraw Hill, 1st Edition, 2010

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO 1	Understand the concept of mobility landscape, mobile apps development and mobile app development environment along with emulator
CLO 2	Design an app user interface with the use of provided GUI resources
CLO 3	Learn about the life cycle, states and interaction among the activities.
CLO 4	Understand the working of Threads, Services, Notifications and Broadcast Receivers, on device file IO and Shared preferences.
CLO 5	Design the animated and multimedia oriented mobile apps by the use of animation API.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA503: IT PROJECT MANAGEMENT				
	L	T	P	Cr
	3	0	2	4.0

Course objective: Learn and Explore SPM activities through knowledge of IT project management and planning.

Introduction to IT Project Management: The characteristics of IT projects, Objectives of project management: time, cost and quality, Basics of Project Management, Stakeholders, Stages of Project, The Feasibility Study, Cost-benefit Analysis, Planning, Project Execution, Project and Product Life Cycles, Project Management Knowledge areas, Project Management Tools & Techniques, Project success factors, role of project manager.

Project Management and Planning: System view of project management, Understanding organizations, stakeholder's management, project phases and project's life cycles. Introduction to Agile software, Why planning is necessary, Iterative steps for planning, Project Plan documentation methods, Software Requirement Specification.

Measurement and Control: Measurements for project monitoring, what and when to measure, Plan versus Control, managing the plan, The Deadline Effect. Reviews, feedback and reporting mechanisms, revisiting the plan.

Project Scope Management: Scope Planning & Scope management plans, Function point calculation, Scope definitions & project scope statement, Work Breakdown Structure (WBS), WBS dictionary, scope verification, scope control.

Time Management: Project time management, activities sequencing, network diagrams, activity recourse estimation, activity duration estimation, schedule development, Gantt Charts, Critical path method, Programme evaluation & review technique (PERT) and CPM, concept of slack time, schedule control.

Project Cost management: Basis principles of cost management, Cost estimating, type of cost estimate, cost estimate tools & techniques, COCOMO, Putnam/ SLIM model Estimating by Analogy, cost budgeting, cost control, earned value management, project portfolio management

Project Quality Management: Quality Planning, quality Assurance, Quality control, Tool &techniques for quality control, Pareto Analysis, Six Sigma, CMM, ISO Standards, Juran Methodology

Project Human Resource Management: Human resource planning, project organizational charts, responsibility assignment metrics, acquiring project team, resource assignment, resource loading, resource levelling, Different team structures developing project teams.

Project Communication Management: Communication Planning, Performance reporting, managing stakeholders, improving project communication

Project risk management: Risk Management planning, common sources of risk, risk identification, risk register, qualitative risk analysis, using probability impact matrixes, expert judgement, qualitative risk analysis, decision trees & expected monetary value, simulation, sensitivity analysis, risk response planning, risk monitoring & control.

Project procurement management: Procurement management plans, contract statement of work, planning contracts, requesting seller responses, selecting sellers, administrating the contract, closing the contract

Software Configuration Management: Why versions exist, why retain versions, SCI, Releases vs. version. Change Control and Management.

Laboratory work: Using Function Point calculation tools for estimation, comparing with COCOMO estimates, Implementation of various exercises using PERT, CPM methods, Preparing schedule, resource allocation etc. using MS Project or Fissure. sim or VENSIM can also be used, Preparing an RMMM Plan for a case study, Preparing Project Plan for a Software Project for Lab Project or case study. Exploring about PMBOK (Project

Management Body of Knowledge) and SWEBOK (Software Engineering Body of Knowledge) from related website, Implementation of software project management concepts using related tools and technologies.

Recommended Books

1. Hughes B. and Cotterell M. and Mall R., Software Project Management, Tata McGraw Hill (2011) 5th Ed.
2. Pressman R., A practitioner's Guide to Software Engineering, Tata McGraw Hill (2014) 7th Ed.
3. Stellman A., Greene J., Applied Software Project Management, O'Reilly Media, Inc. (2008).
4. Futrell T. R., Shafer F. D. and Shafer I. L., Quality Software Project Management, Prentice Hall (2002).

COURSE LEARNING OUTCOMES (CLOs)

CLO 1	Describe and apply basic concepts related to IT project planning, scope and feasibility.
CLO 2	Analyze IT project estimation techniques.
CLO 3	Comprehend the concept of team structure and project communication management.
CLO 4	Acquire knowledge about quality assurance, quality control, and risk management.
CLO 5	Describe various project management activities such as tracking, project procurement, configuration management, monitoring.

PCA602: HUMAN COMPUTER INTERACTION

L	T	P	Cr
3	0	4	4.0

Course Objectives: To understand the interaction between human and computer components and design interactive user interface.

Introduction to graphical user interface: Importance of user Interface – definition, importance of good design, benefits of good design, a brief history of Screen design, Popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – Interface popularity, characteristics- Principles of user interface.

Design Process: Interaction design basics, Human interaction with computers, design rules, Implementation support, Evaluation techniques, Importance of human characteristics, Human consideration, Human interaction speeds, and understanding business junctions.

Models and Theories: Cognitive Models- Goal and task hierarchies, Linguistic Models, Task analysis- task decomposition, knowledge based analysis, Entity-relationship-based techniques, Uses of task analysis.

Screen Designing: Design goals, Screen planning and purpose, organizing screen elements, ordering of screen data and content , screen navigation and flow , Visually pleasing composition , amount of information, focus and emphasis , presenting information simply and meaningfully , information retrieval on web , statistical graphics , Technological consideration in interface design.

Menus, Navigation and Windows: Structure of menus, Functions of menus, Navigating menus, Types of graphical menus, Navigation schemes for selection of window, window characteristics, components of a window, window presentation styles, types of windows, window management, organizing window functions, window operations.

Components and Interaction Devices: Text and messages, Icons and increases – Multimedia, colors, uses problems, choosing colors.Keyboard and function keys, pointing devices, speech recognition digitization and generation, image and video displays, drivers.

User-Centered Design and Testing: Functionality and usability requirements, Techniques for gathering requirements, Techniques and tools for the analysis and presentation of requirements, Prototyping techniques and tools, Evaluation without users, using both qualitative and quantitative techniques.

Laboratory Work: Main focus is on designing and implementing visually appealing graphical user interface. To implement all the related programs on relevant tools and technologies.

Recommended Books

1.Wilbert O. Galitz, The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques, John Wiley & Sons (2007).
2. Shneidermann B., Designing the user interface, Pearson Education Asia (2004), 3rded.
3. Dix A., Fincay J., Goryd G., Abowd, Bealg R., Human Computer Interaction, Pearson Education (2004), 3rd ed.
4. Lauesen S., User Interface Design: A Software Engineering Perspective, Pearson Education (2004).

Course Learning Outcomes (CLOs): On completion on the course students would be able to:

CLO1	Understand the interaction between human and computer components, importance of good interface design
CLO2	Implement the principles of GUI and design a visually pleasing UI.
CLO3	Understand and implement interaction design basics and associated design rules, cognitive architecture, linguistic models and task analysis techniques including task decomposition
CLO4	Understand the structure of menus, navigational scheme and characteristics of windows, concepts of Multimedia, pointing devices, speech recognition digitization and generation
CLO5	Learn gathering requirements and usage of tools for analysis and user requirements.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	25
2	EST	40
3	Sessional (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA603 AGILE SOFTWARE DEVELOPMENT				
	L	T	P	Cr
	3	0	2	4.0

Course Objectives: To learn the fundamental principles and practices associated with each of the agile development methods. To apply the principles and practices of agile software development on a project of interest and relevance to the student.

Agile Software Development: Basics and Fundamentals of Agile Process Methods, Values of Agile, Principles of Agile, stakeholders, Challenges

Lean Approach: Waste Management, Kaizen and Kanban, add process and products add value. roles related to the lifecycle, differences between Agile and traditional plans, differences between Agile plans at different lifecycle phases. Testing plan links between testing, roles and key techniques, principles, understand as a means of assessing the initial status of a project/ How Agile helps to build quality

Agile and Scrum Principles: Agile Manifesto, Twelve Practices of XP, Scrum Practices, Applying Scrum. Need of scrum, working of scrum, advanced Scrum Applications, Scrum and the Organization , scrum values

Agile Product Management: Communication, Planning, Estimation Managing the Agile approach Monitoring progress, Targeting and motivating the team, Managing business involvement, Escalating issue. Quality, Risk, Metrics and Measurements, Managing the Agile approach Monitoring progress, Targeting and motivating the team, Managing business involvement and Escalating issue

Agile Requirements: User Stories, Backlog Management. Agile Architecture**:** Feature-Driven Development. Agile Risk Management**:** Risk and Quality Assurance, Agile Tools

Agile Testing: Agile Testing Techniques, Test-Driven Development, User Acceptance Test

Agile Review: Agile Metrics and Measurements, The Agile approach to estimating and project variables, Agile Measurement, Agile Control: the 7 control parameters. Agile approach to Risk, The Agile approach to Configuration Management, The Atern Principles , Atern Philosophy ,The rationale for using Atern, Refactoring, Continuous integeration, Automated Build Tools

Scaling Agile for large projects: Scrum of Scrums, Team collaborations, Scrum, Estimate a Scrum Project, Track Scrum Projects, Communication in Scrum Projects, Best Practices to Manage Scrum.

Laboratory Work: exploring the tools related to Agile Development and approached and develop small projects using this technology.

Recommended Books

1. Agile Software Development, Principles, Patterns, and Practices (Alan Apt Series) Robert C. Martin (Author) ,2011
2. Succeeding with Agile : Software Development Using Scrum, Pearson 2010

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Analyze existing problems with the team, development process and wider organization
CLO2	Apply a thorough understanding of Agile principles and specific practices
CLO3	Select the most appropriate way to improve results for a specific circumstance or need
CLO4	Judge and craft appropriate adaptations to existing practices or processes depending upon analysis of typical problems
CLO5	Evaluate likely successes and formulate plans to manage likely risks or problems

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA604: ENTERPRISE RESOURCE PLANNING				
	L	T	P	Cr
	2	0	2	3.0

Course Objectives: To make student able to build an understanding of the fundamental concepts of ERP systems, their architecture, and working of different modules in ERP. Students will also able to develop and design the modules used in ERP systems, and can customize the existing modules of ERP systems.

Introduction to ERP: ERP Overview, Benefits, Business process reengineering, ERP implementation life cycle, Options of various paradigms, Supply chain Management, Critical factors guiding selection and evaluation, Strategies for successful implementation, impediments and initiatives to achieve success, Critical success and failure factors, Integrating ERP into organizational culture.

SAP and ABAP: Architecture of SAP, Data types in ABAP, ABAP programming Language, ABAP User Dialogs, Function groups and function modules, Accessing Database Access, open SQL, Native SQL, ABAP Object Orientation, Classes and objects in ABAP, Inheritance, Interfaces, Triggering and Handling Events, ABAP data dictionary, Declarations, selection screens, Formatting and Displaying Data, Program Events, , Dynpros, BSP applications.

SD: Basic functions and master data in SD, Sales orders, Deliveries, Pricing, Billing, Transportation, Credit Management.

MM: Basic functions and master data, Consumption based planning, Purchasing, Inventory management, Evaluation of materials, Invoice verification, Balance sheet evaluation, Material ledger.

Laboratory Work: Implementation of ERP concepts using any ERP package.

Recommended Books

1. Mary Sumner, Enterprise Resource Planning, Pearson Education (2010).
2. Kogent Learning Solutions Inc., SAP ABAP/4 Covers SAP ECC 6.0 Black Book, Dreamtech Press (2009).
3. Bradford M., Modern ERP Systems: Select Implement and Use Today's Advanced Business Systems (2010).
4. Dreamtech Software Team, SAP Architecture, Administration, Basis, ABAP Programming with MM and SD modules, Dreamtech Press (2005).

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand the architecture of ERP and working of different modules in ERP.
CLO2	Identify the risks and challenges in implementation of ERP system
CLO3	Develop and design the ERP modules and Customize the existing modules of an ERP systems.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA406 PARALLEL AND DISTRIBUTED COMPUTING				
	L	T	P	Cr
	3	0	2	4.0

Course Objective: To learn the concepts of Parallel and Distributed Computing and its implementation for assessment of understanding the course by the students

Parallelism Fundamentals : Scope and issues of parallel and distributed computing, Parallelism, Goals of parallelism, Parallelism and concurrency, Multiple simultaneous computations, Programming Constructs for creating Parallelism, communication, and coordination. Programming errors not found in sequential programming like data races, higher level races, lack of liveness

Parallel Architecture : Architecture of Parallel Computer, Communication Costs, parallel computer structure, architectural classification schemes, Multicore processors, Memory Issues : Shared vs. distributed, Symmetric multiprocessing (SMP), SIMD, vector processing, GPU, co-processing, Flynn's Taxonomy, Instruction Level support for parallel programming, Multiprocessor caches and Cache Coherence, Non-Uniform Memory Access (NUMA)

Parallel Decomposition and Parallel Performance: Need for communication and coordination/synchronization, Scheduling and contention, Independence and partitioning, Task-Based Decomposition, Data Parallel Decomposition, Actors and Reactive Processes, Load balancing, Data Management, Impact of composing multiple concurrent components, Power usage and management. Sources of Overhead in Parallel Programs, Performance metrics for parallel algorithm implementations, Performance measurement, The Effect of Granularity on Performance Power Use and Management, Cost-Performance trade-off;

Communication and Coordination: Shared Memory, Consistency, Atomicity, Message-Passing, Consensus, Conditional Actions, Critical Paths, Scalability, cache coherence in multiprocessor systems, synchronization mechanism.

CUDA programming model: Overview of CUDA, Isolating data to be used by parallelized code, API function to allocate memory on the parallel computing device, API function to transfer data to parallel computing device, Concepts of Threads, Blocks, Grids, Developing kernel function that will be executed by threads in the parallelized part, Launching the execution of kernel function by parallel threads, transferring data back to host processor with API function call.

Parallel Algorithms design, Analysis, and Programming: Parallel Algorithms, Parallel Graph Algorithms, Parallel Matrix Computations, Critical paths, work and span and relation to Amdahl's law, Speed-up and scalability, Parallel algorithmic patterns like divide and conquer, map and reduce, Specific algorithms like parallel Merge Sort, Parallel graph algorithms, parallel shortest path, parallel spanning tree.

Laboratory Work: To implement parallel programming using CUDA with emphasis on developing applications for processors with many computation cores, mapping computations to parallel hardware, efficient data structures, paradigms for efficient parallel algorithms

Recommended Books

1. C Lin, L Snyder. Principles of Parallel Programming. USA: Addison-Wesley 2008.
2. A Grama, A Gupra, G Karypis, V Kumar. Introduction to Parallel Computing (2nd ed.). Addison Wesley, 2003.
3. B Gaster, L Howes, D Kaeli, P Mistry, and D Schaa. Heterogeneous Computing With Opencl. Morgan Kaufmann and Elsevier, 2011.
4. T Mattson, B Sanders, B Massingill Patterns for Parallel Programming. Addison-Wesley 2004.
5. Quinn, M. J. (2004). Parallel Programming in C with MPI and OpenMP. McGraw-Hill.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand the tasks in parallel and distributed computing, different parallel architectures, programming models, and algorithms for common operations.
CLO2	Develop efficient parallel algorithms to solve various problems.
CLO3	Analyze time complexity as a function of the problem size and number of processors.
CLO4	Analyze performance, determine computational bottlenecks, and optimize the performance of the code.
CLO5	Use CUDA programming to solve different parallel algorithms.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA407 MACHINE LEARNING

L	T	P	Cr
3	0	2	4.0

Course Objectives: This course provides an advanced level of understanding to machine learning and statistical pattern recognition. It offers some of the most cost-effective approaches to automated knowledge acquisition in emerging data-rich disciplines and focuses on the theoretical understanding of these methods, as well as their computational implications.

Introduction: Well-Posed learning problems, Basic concepts, Designing a learning system, Issues in machine learning. Types of machine learning: Learning associations, Supervised learning (Classification and Regression Trees, Support vector machines), Unsupervised learning (Clustering), Instance-based learning (K-nearest Neighbor, Locally weighted regression, Radial Basis Function), Reinforcement learning (Learning Task, Q-learning, Value function approximation, Temporal difference learning).

Decision Tree Learning: Decision tree representation, appropriate problems for decision tree learning, Univariate Trees (Classification and Regression), Multivariate Trees, Basic Decision Tree Learning algorithms, Hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Issues in decision tree learning.

Bayesian Learning: Bayes theorem and concept learning, Bayes optimal classifier, Gibbs algorithms, Naive Bayes Classifier, Bayesian belief networks, The EM algorithm.

Artificial Neural Network: Neural network representation, Neural Networks as a paradigm for parallel processing, Linear discrimination, Pairwise separation, Gradient Descent, Logistic discrimination, Perceptron, Training a perceptron, Multilayer perceptron, Back propagation Algorithm. Recurrent Networks, Dynamically modifying network structure.

Genetic Algorithms: Basic concepts, Hypothesis space search, Genetic programming, Models of evolution and learning, Parallelizing Genetic Algorithms.

Data Mining Techniques for Analysis: Classification: Decision tree induction, Bayes classification, Rule-based classification, Support Vector Machines, Classification Using Frequent Patterns, k-Nearest-Neighbor, Fuzzy-set approach Classifier, Clustering:K-Means, k-Medoids, Agglomerative versus Divisive Hierarchical Clustering Distance Measures in Algorithmic Methods, Mean-shift Clustering

Laboratory Work: It is concerned with the design, analysis, implementation, and applications of programs that learn from experience. Learning algorithms can also be used to model aspects of human and animal learning.

Recommended Books

1. Mitchell T.M., Machine Learning, McGraw Hill (1997) 2nd ed.
2. Alpaydin E., Introduction to Machine Learning, MIT Press (2010) 2nd ed.
3. Bishop C., Pattern Recognition and Machine Learning, Springer-Verlag (2006) 2nd ed.
4. Michie D., Spiegelhalter D. J., Taylor C. C., Machine Learning, Neural and Statistical Classification. Overseas Press (2009) 1st ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of the course, students will able to:

CLO1	Demonstrate in-depth knowledge of methods and theories in the field of machine learning. To provide an introduction to the basic principles, techniques, and applications of Machine Learning, Classification Tasks, Decision tree learning.
CLO2	Understand and use Bayesian perspective on machine learning, Artificial neural networks, back propagation algorithm
CLO3	Assess learning algorithms modelled after biological evolution, including genetic algorithms and genetic programming.
CLO4	Assess explanation-based learning that uses prior knowledge to explain observed training examples, then generalizes based on these explanations and discuss approaches to combining approximate prior knowledge with available training data in order to improve the accuracy of learned hypotheses.
CLO5	Demonstrate knowledge of the disciplinary foundation and of proven experience in the design and analysis of learning algorithms and systems. To demonstrate the ability to critically evaluate and compare different learning models and learning algorithms and be able to adapt or combine some of the key elements of existing machine learning algorithms to design new algorithms as needed.

Evaluation Scheme:

S No	Evaluation Elements	Weightage (%)
1.	MST	20
2.	EST	40
3.	Sessionals (May include Assignments/ Projects/ Tutorials/ Quizes/ Lab Evaluations)	40

PCA408: CRYPTOGRAPHY AND NETWORK SECURITY				
	L	T	P	Cr
	3	0	2	4.0

Course Objectives: This course is designed to impart a critical theoretical and detailed practical knowledge of a range of computer network security technologies as well as network security tools.

Introduction to Cryptography: Concepts and Techniques. Security Trends, Possible Types of Attacks, Model for Network Security, Plain Text and Cipher Text, Encryption and Decryption , Substitution Techniques, Transposition Techniques DES, AES, Principals of public key crypto systems, RSA algorithm, Security of RSA, Symmetric and Asymmetric Key Cryptography.

Introduction: Essential Terminology, Elements of Security, Difference between Hacking and Ethical Hacking.

Reconnaissance: Information Gathering Methodology, Active and Passive reconnaissance.

Scanning: Scanning, Elaboration phase, active scanning, scanning tools NMAP, hping2. Enumeration, DNS Zone transfer. Detecting live systems on the target network, Discovering services running /listening on target systems, Understanding port scanning techniques, Identifying TCP and UDP services running on the target network, Understanding active and passive fingerprinting

Sniffers: Definition of sniffing, Sniffer working, Passive Sniffing, Man-in-the-Middle Attacks, Spoofing and Sniffing Attacks, ARP Poisoning and countermeasures.

Denial of Service: Goal of DoS (Denial of Service), Impact and Modes of Attack.

Session Hijacking: Understanding Session Hijacking, Spoofing vs Hijacking, Steps in Session Hijacking, Types of Session Hijacking

Ethical Hacking- System Hacking and Hacking Wireless Networks: Understanding Sniffers ,Comprehending Active and Passive Sniffing, ARP Spoofing and Redirection, DNS and IP Sniffing, HTTPS Sniffing.

Introduction to 802.11, Role of WEP, Cracking WEP Keys, Sniffing Traffic, Wireless DOS attacks, WLAN Scanners, WLAN Sniffers, Hacking Tools, Securing Wireless Networks.

Laboratory work: To learn and experiment different types of scenerios related to Security of an organization. To implement core algorithms in high level language and to use related tools and technologies for various security concepts above.

Recommended Books

1. Hackers Beware, Eric Core, EC-Council Press, 2003
2. Network Security Essentials, William Stallings , Prentice Hall, 5th Edition, 2013
3. Firewalls and Internet Security, William R. Cheswick and Steven M. Bellovin, Addison-Wesley Professional, 2nd Edition, 2003.
4. Cryptography and Network Security, W. Stallings , Prentice Hall, 5th Edition, 2010

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Identifying the possible types of attacks on the networks and studying different types of the algorithms for securing the data.
CLO2	Familiarization with the scanning tools for the information gathering.
CLO3	Recognization of the phases in the Session Hijacking and the use of tools for counter-measuring the various Sniffing attacks.
CLO4	Analyzing the attacks on Wireless Networks and use of tools for securing them.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA409 DATABASE ADMINISTRATION

L	T	P	Cr
3	0	2	4.0

Course Objective: Detailed study of database in all aspects from database administration, management, creation, security, configuring the DBMS network to data backup, recovery and its transportation.

Database administration: Database, Data, and System Administration. DBA tasks, Types of DBA, New Technology Impacts on DBA. DBA roles and responsibility.

Database Architectural Components: Database architectures, DBMS Server, DBMS Instance, System Global Area, Program Global Area, Logical Structure, Installing and managing DBMS servers.

Creating the Database and its Management: Planning the Database, Managing the database Instance, Starting and Stopping Database Control, Database Startup and Shutdown database Change Management: Managing Database Storage Structures- How Table Data is Stored, Creating a New Table space, Altering a Table space, Actions with Table spaces, Dropping Table spaces, Managing Schema Objects-Managing data files, Managing tables, indexes, and constraints, Managing Data and Concurrency, Managing Undo Data.

Data Availability, Administering User and Database Security: Define availability, Availability Problems, Impact of Change on Database Structures, Performance Management: System, Database, and Application performance. Database User Accounts, Predefined Accounts: SYS and SYSTEM, Creating a User, Authenticating Users, Privileges, Benefits of Roles, Creating a Role, Profiles and Users, Assigning Quota to Users, Implementing Database Security.

Configuring the DBMS Network Environment: Database net server side configuration, basic net client-side configuration.

Data Backup, Recovery and Its transportation: Backup and Recovery, Transporting data between databases – export and import utility.

Laboratory work: To understand and use database environment, physical structure, granting/revoking privileges and Recovery Manager (RMAN). To use tools and techniques related to data base administration.

Recommended Books:

1. Oracle Database 10*g*:Administration Workshop I, Oracle Press
2. Loney K., Baryla B. Oracle database 10g: DBA handbook, Tata McgrawHills.
3. Thomas B., Baryla B., Oracle DBA Fundamental-II, BPB publications.
4. Carig S. Mullins, Database Administration: The Complete Guide to DBA Practices and Procedures Second Edition, Addison-Wesley.8288014010

COURSE LEARNING OUTCOMES (CLOs):

CLO1	To understand the database architectural components and role of a DBA.
CLO2	To learn about creation and management of a database.
CLO3	To learn for administrating the user and security aspects of database.
CLO4	To perform hands-on for configuring database network environment and data backup, recovery.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Internal Evaluations: Quizzes Lab Evaluations	 20 15

PCA410 IMAGE PROCESSING				
	L	T	P	Cr
	3	0	2	4.0

Course Objective: To learn the advanced concepts Image Processing and Video Processing and its implementation for assessment of understanding the course by the students.

Introduction and Digital Image Fundamentals: Digital Image Representation, Fundamental steps in Image Processing, Elements of Digital image processing systems, Sampling and quantization, neighbors of a pixel, adjacency, connectivity, Regions and Boundaries, Distance measures, Image operations on a pixels basis, Linear and Nonlinear operations.

Image Enhancement and Restoration: Image Enhancement in the Spatial domain: Gray level transforms, Histogram Processing, Enhancement using Arithmetic/Logic Operations, smoothing and sharpening filters, Image Enhancement in the Frequency domain: 1-D and 2-D Fourier Transform and their Inverse, Filtering, Smoothing and sharpening domain filters, Homomorphic Filtering, Degradation Model, Noise models, Restoration in the presence of Noise only spatial filtering, Periodic Noise reduction by frequency domain filtering, estimating the degradation function.

Color Image Processing: Color models, Pseudocolor Image Processing, Color Transforms, Smoothing and sharpening, Color Segmentation, Noise in color images, Color Image compression.

Image Compression: Fundamentals, Compression Models, Error free Comparison, Lossy Compression, wavelets in Image compression, Image compression standards.

Morphological Image Processing: Dilation and Erosion, Basic Morphological algorithms, Extension to gray scale images.

Image Segmentation, Representation and Description: Detection of discontinuities, Edge linking and boundary detection, Thresholding, Region oriented Segmentation, Motion based Segmentation; Representation schemes, Boundary description, Regional descriptors, Morphology.

Object Recognition: Patterns and Pattern classes, Decision Theoretic Methods, Structural methods.

Laboratory Work: Demonstrate the use of Image Processing Toolbox on MATLAB/Python/Octave/SciLab to create interactive image processing applications like image enhancement, image compression, image segmentation, feature extraction etc.

Recommended Books:

1. Gonzalez R. C. and Woods R. E., "Digital Image Processing", Edition 2nd, Pearson Education, 2007.
2. Jain A. K., "Fundamentals of Digital Image Processing", Edition 4th, PHI, 2002.
3. Umbaugh S E, "Computer Imaging, Digital Image Analysis and Processing", Edition 3rd, CRC Press Book, 2005.
4. Gonzalez R C, Woods R.E. and Eddins S.L., "Digital Image Processing using MATLAB", Pearson Education, 2004.
5. Wang Y, Ostermann J, Zhang Y Q, "Video Processing and Communications", Edition 1st, Prentice Hall, 2001.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand basic fundamentals of digital images
CLO2	Enhance the visual quality of given grey/color image using well known transformations and filters
CLO3	Distinguish between lossy and lossless image compression
CLO4	Implement and analyse various image segmentation techniques
CLO5	Recognize objects from the image using object recognition.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA504 GPU COMPUTING				
	L	T	P	Cr
	3	0	2	4.0

Course Objective: To study architecture and capabilities of modern GPUs and learn programming techniques for the GPU such as CUDA programming model.

Introduction: Heterogeneous Parallel Computing, Architecture of a Modern GPU, Speeding Up Real Applications, Parallel Programming Languages and Models.

History of GPU Computing: Evolution of Graphics Pipelines, The Era of Fixed-Function Graphics Pipelines, Evolution of Programmable Real-Time Graphics, Unified Graphics and Computing Processors, GPGPU, Scalable GPUs, Recent Developments, Future Trends.

Introduction to Data Parallelism and CUDA C: Data Parallelism, CUDA Program Structure, A Vector Addition Kernel, Device Global Memory and Data Transfer, Kernel Functions and Threading.

Data-Parallel Execution Model: CUDA Thread Organization, Mapping Threads to Multidimensional Data, Matrix-Matrix Multiplication—A More Complex Kernel, Synchronization and Transparent Scalability, Assigning Resources to Blocks, Thread Scheduling and Latency Tolerance.

CUDA Memories: Importance of Memory Access Efficiency, CUDA Device Memory Types, A Tiled Matrix – À Matrix Multiplication Kernel, Memory as a Limiting Factor to Parallelism.

An Introduction to OpenCL: Data Parallelism Model, Device Architecture, Kernel Functions, Device Management and Kernel Launch, Electrostatic Potential Map in OpenCL.

Parallel Programming with OpenACC: OpenACC Versus CUDA C, Execution Model, Memory Model, Basic OpenACC Programs, Parallel Construct, Loop Construct, Kernels Construct, Data Management, Asynchronous Computation and Data Transfer.

Laboratory work: Practice programs using CUDA, OpenCL and OpenACC.

Recommended Books:

1. Sanders, J. and Kandrot, E., CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison-Wesley Professional (2012) 4th Edition.
2. Kirk, D. and Hwu, M., W., Programming Massively Parallel Processors: A Hands-on Approach. Morgan Kaufmann (2016) 3rd Edition.
3. Hwu, M., W., A GPU Computing Gems Emerald Edition (Applications of GPU Computing Series), Morgan Kaufmann (2011) 1st Edition

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Comprehend commonly used terms in parallel computing.
CLO2	Understand common GPU architectures and Programming Models.
CLO3	Implement algorithms efficiently for common application kernels.
CLO4	Develop efficient parallel algorithms to solve given problems.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA505 DEEP LEARNING				
	L	**T**	**P**	**Cr**
	3	**0**	**2**	**4.0**

Course Objective: There have been many recent advances in the field of deep learning. The objective of the course is to provide exposure to these advances and facilitate in depth discussion on chosen topics.

Machine Learning Basics: Learning, Underfitting, Overfitting, Estimators, Bias, Variance, Maximum Likelihood Estimation, Bayesian Statistics, Supervised Learning, Unsupervised Learning and Stochastic Gradient Decent.

Deep Feedforward Network: Feed-forward Networks, Gradient-based Learning, Hidden Units, Architecture Design, Computational Graphs, Back-Propagation, Regularization, Parameter Penalties, Data Augmentation, Multi-task Learning, Bagging, Dropout and Adversarial Training and Optimization.

Convolution Networks: Convolution Operation, Pooling, Basic Convolution Function, Convolution Algorithm, Unsupervised Features and Neuroscientific for convolution Network.

Sequence Modelling: Recurrent Neural Networks (RNNs), Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Network, Recursive Neural Networks and Echo State networks.

Deep Generative Models: Boltzmann Machines, Restricted Boltzmann Machines, Deep Belief Networks, Deep Boltzmann Machines, Sigmoid Belief Networks, Directed Generative Net, Drawing Samples from Auto –encoders.

Laboratory Work: To implement models using python and google open source library Tensorflow.

Recommended Books:

1. Goodfellow L., Bengio Y. and Courville A., Deep Learning, MIT Press (2016).
2. Patterson J. and Gibson A., Deep Learning: A Practitioner's Approach, O'Reilly (2017), 1st ed.
3. Haykin S., Neural Network and Machine Learning, Prentice Hall Pearson (2009), 3rd ed.
4. Geron A., Hands-on Machine Learning with Sci-kit and TensorFlow, O'Reilly Media (2017).

Course Learning Outcomes (CLOs)

CLO1	Comprehend the advancements in learning techniques.
CLO2	Compare and explain various deep learning architectures and algorithms.
CLO3	Demonstrate the applications of deep learning in various fields.
CLO4	Apply deep learning specific open source libraries for solving real life problems.

Evaluation Scheme:

S.No.	Evaluation Elements	Weightage (%)
1.	MST	20
2.	EST	40
3.	Sessionals (May include Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	40

PCA506 SECURE CODING				
	L	**T**	**P**	**Cr**
	3	**0**	**2**	**4.0**

Course Objective: This course aims to provide an understanding of the various security attacks and knowledge to recognize and remove common coding errors that lead to vulnerabilities. It gives an outline of the techniques for developing a secure application.

Introduction: Security, CIA Triad, Viruses, Trojans, and Worms, Security Concepts- exploit, threat, vulnerability, risk, attack.

Decipher journey starting from FQDN to html page getting served to browser, Authoritative reply, revisit layer 2 and layer 3 of TCP/IP, DNS poisoning, ARP poisoning, C language obfuscation. ARP poisoning and its countermeasures. Buffer Overrun- Stack overrun, Heap Overrun, Array Indexing Errors, Format String Bugs, PE Code injection.

Malware Terminology: Rootkits, Trapdoors, Botnets, Key loggers, Honeypots. Active and Passive Security Attacks. IP Spoofing, Tear drop, DoS, DDoS, XSS, SQL injection, Smurf, Man in middle, Format String attack.

Types of Security Vulnerabilities: buffer overflows, Invalidated input, race conditions, access-control problems, weaknesses in authentication, authorization, or cryptographic practices. Access Control Problems.

Need for secure systems: Proactive Security development process, Secure Software Development Cycle (SSDLC) , Security issues while writing SRS, Design phase security, Development Phase, Test Phase, Maintenance Phase, Writing Secure Code – Best Practices SD3 (Secure by design, default and deployment), Security principles and Secure Product Development Timeline.

Threat modelling process and its benefits: Identifying the Threats by Using Attack Trees and rating threats using DREAD, Risk Mitigation Techniques and Security Best Practices. Security techniques, authentication, authorization. Defence in Depth and Principle of Least Privilege.

Secure Coding Techniques: Protection against DoS attacks, Application Failure Attacks, CPU Starvation Attack.

Database and Web-specific issues: SQL Injection Techniques and Remedies, Race conditions, Time of Check Versus Time of Use and its protection mechanisms. Validating Input and Interprocess Communication, Securing Signal Handlers and File Operations. XSS scripting attack and its types – Persistent and Non-persistent attack XSS Countermeasures and Bypassing the XSS Filters.

Laboratory Work: In this Lab, the student will be able to practically understand how all the security attacks does has happened, as well as learn to recognize and remove common coding errors that lead to vulnerabilities. This lab also gives an outline of the techniques for developing a secure application code that consists of using network monitoring tools, implementing different types of attacks and some protection schemes. Evaluation will be mainly based on projects and assignments.

Recommended Books:

1. Howard, M. and LeBlanc, D., Writing Secure Code, Howard, Microsoft Press (2002) 2nd Edition.
2. Deckard, J., Buffer Overflow Attacks: Detect, Exploit, Syngress (2005) 1st Edition.
3. Swiderski, F. and Snyder, W., Threat Modeling, Microsoft Professional, (2004) 1st Edition.

4. Salt, C., J., SQL Injection Attacks and Defence, Elsevier (2012), 2nd Edition

Course Learning Outcomes (CLOs)

CLO1	Implement ARP poisoning attack and demonstrate countermeasure against these for different operating environments.
CLO2	Implement DNS poisoning attack and demonstrate authoritative reply in this context.
CLO3	Implement PE Code injection and demonstrate control hijacking via EIP manipulation
CLO4	Demonstrate skills needed to deal with common programming errors and develop secure applications.
CLO5	Demonstrate client-side attacks and identify nature of threats to software and incorporate secure coding practices throughout the planning and development of software product.
CLO6	Demonstrate SQL, XSS attack and suggest countermeasures for the same.

Evaluation Scheme:

S.No.	Evaluation Elements	Weightage (%)
1.	MST	20
2.	EST	40
3.	Sessionals (May include Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	40

PCA507 NETWORK ADMINISTRATION				
	L	**T**	**P**	**Cr**
	3	**0**	**2**	**4.0**

Course Objective: To learn concepts of networking as well as related equipment and terminologies.

Networking Overview: History, Protocol Standards**,** Reference Model (OSI, TCP/IP), Windows and Linux Networking Basics**,** Switching and Routing basics.

Server Administration Basics: Server and Client Installation, Boot Process and Startup Services:Xinetd/Inetd, Managing accounts: users, groups and other privileges, File Systems and Quota Management, Job Scheduling with cron, crontab, anacron and system log analysis, Process controlling and management, Online Server upgrade/update process, Administering Database Server (MySQL).

Network Configuration Basics: IPv4 and IPv6 addressing, Network Interface Configuration, Diagnosing Network startup issues, Linux and Windows Firewall configuration, Network troubleshooting commands.
Dynamic Host Configuration Protocol (DHCP): DHCP Principle**,** DHCP Server Configuration**,** DHCP Options, Scope, Reservation and Relaying**,** DHCP Troubleshooting.

Name Server and Configuration: DNS principles and Operations, Basic Name Server and Client Configuration, Caching Only name server, Primary and Slave Name Server, DNS Zone Transfers, DNS Dynamic Updates, DNS Delegation, DNS Server Security, Troubleshooting.

Web and Proxy Server Configuration: HTTP Server Configuration Basics**,** Virtual Hosting**,** HTTP Caching**,** Proxy Caching Server Configuration**,** Proxy ACL**,** Proxy-Authentication Mechanisms**,** Troubleshooting.

FTP, File and Print Server: General Samba Configuration, SAMBA SWAT, NFS and NFS Client Configuration, CUPS configuration basics, FTP Principles, Anonymous FTP Server, Troubleshooting.

Mail Server: SMTP, POP and IMAP principles, SMTP Relaying Principles, Mail Domain Administration, Basic Mail Server Configuration (Sendmail, postfix, qmail, exim..), SPAM control and Filtering, Troubleshooting.

Remote Administration and Management: Router Configuration**,** Webmin/usermin**,** Team Viewer**,** Telnet**,** SSH**,** SCP, Rsync

Recommended Books:

1. The Practice of System and Network Administration, Second Edition Thomas A. Limoncelli, Christina J. Hogan , Strata R. Chalup
2. Advanced Linux Networking, Roderick W. Smith, Addison-Wesley Professional (Pearson Education), 2002.
3. Linux Network Administrator's Guide, Tony Bautts, Terry Dawson, Gregor N. Purdy, O'Reilly, Third Edition, 2005

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Install, configure and manage enterprise networks, including hardware/software.

CLO2	Implement and administer desktop and server operating systems (client/server), switching and routing devices.
CLO3	Create user/group accounts and configure server roles.
CLO4	Administer permissions for users, files and network resources.
CLO5	Maintain and troubleshoot enterprise networks

Evaluation Scheme:

Sr.No.	Evaluation Elements	Weightage (%)
1	MST	25
2	EST	40
3	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA508 3D MODELLING AND ANIMATION				
	L	**T**	**P**	**Cr**
	3	**0**	**2**	**4.0**

Course Objective: To develop the skill & knowledge in 3D Modeling and Animation. Students will understand the know how and can function either as an entrepreneur or can take up jobs in the multimedia and animation industry, video studios, edit set-up and other special effects sectors.

Introduction: Definition of Computer-based Animation, Basic Types of Animation: Real Time,Non-real-time, Definition of Modelling, Creation of 3D objects. Exploring the Max Interface, Controlling & Configuring the Viewports, Customizing the Max Interface & Setting Preferences, Working with Files, Importing & Exporting, Selecting Objects & Setting Object Properties, Duplicating Objects, Creating & Editing Standard Primitive & extended Primitives objects, Transforming objects, Pivoting, aligning etc.

2D Splines & Shapes& compound object: Understanding 2D Splines & shape, Extrude & Bevel 2D object to 3D, Understanding Loft & terrain, Modeling simple objects with splines, Understanding morph, scatter, conform, connect compound objects, blobmesh, Boolean,Proboolean&procutter compound object.

3D Modelling: Modeling with Polygons, using the graphite, working with XRefs, Building simple scenes, Building complex scenes with XRefs, using assets tracking, deforming surfaces & using the mesh modifiers, modeling with patches & NURBS.

Keyframe Animation: Creating Keyframes, Auto Keyframes, Move & Scale Keyframe on the timeline, Animating with constraints &simple controllers, animation Modifiers & complex controllers, function curves in the track view, motion mixer etc..

Simulation & Effects: Bind to Space Warp object, Gravity, wind, displace force object, deflectors, FFD space warp, wave, ripple, bomb, Creating particle system through PArray, understanding particle flow user interface, how to particle flow works, hair& fur modifier, cloth & garment maker modifiers etc.

Lighting & Camera: Configuring & Aiming Cameras, camera motion blur, camera depth of field, camera tracking, using basic lights & lighting Techniques, working with advanced lighting, Light Tracing, Radiosity, video post, mental ray lighting etc.

Texturing with Max: Using the material editor & the material explorer, creating& applying standard materials, adding material details with maps, creating compound materials &material modifiers, unwrapping UVs & mapping texture, using atmospheric & render effects etc.

Rendering with V-Ray: V-ray light setup, V-ray rendering settings, HDRI Illumination, Fine-tuning shadows, Final render setting etc.

Laboratory Work: This course covers beginner to intermediate 3D Modeling and Animation. In this Lab the students will be able to model the 3D character and objects, its UV Mapping, Texture Painting, Rigging, and Animation. Evaluation will be mainly via projects and assignments taking a creative approach to expressive 3D modelling and Animation.

Recommended Books:

1. House, H., D. and Keyser, C., J., Foundations of Physically Based Modeling and Animation, CRCPress (2017) 1st Edition.
2. Chopine, A., 3D Art Essentials: The Fundamentals of 3D Modeling, Texturing, and Animation, Focal Press (2011) 1stEdition.

3. Zeman, B., N., Essential Skills for 3D Modeling, Rendering, and Animation, A K Peters/CRC Press (2017) 1st Edition.

4. Villar, O., Learning Blender: A Hands-On Guide to Creating 3D Animated Characters, Addison Wesley (2017) 2nd Edition.
5. Kerlow, I., The Art of 3D Computer Animation and Effects, Wiley, (2009) 4th Edition.
6. Flavell, L., Beginning Blender: Open Source 3D Modelling, Animation, and Game Design, Apress, (2010) 1st Edition.
7. Boardman, T., 3dsmax 7 Fundamentals, New Riders, (2005) 1st Edition.

Course Learning Outcomes (CLOs)

CLO1	Describe Computer-based animation using 3D modeling tool (Blender/ Max).
CLO2	Develop the practical skills in 2D Splines, Shapes & compound objects.
CLO3	Illustrate the theoretical and practical aspects of 3D Modeling, Keyframe Animation, Simulation and Effects.
CLO4	Demonstrate different types of animation and its effects in the real world.
CLO5	Analyse the different processes, post processes involved in computer animation field.

Evaluation Scheme:

S.No.	Evaluation Elements	Weightage (%)
1	MST	20
2	EST	40
3	Sessionals (May include Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	40

PCA509 CLOUD COMPUTING				
	L	**T**	**P**	**Cr**
	3	**0**	**2**	**4.0**

Course Objective: To learn the advanced concepts of cloud infrastructure and services and its implementation for assessment of understanding the course by the students.

Introduction and Evolution of Computing Paradigms: Overview of Existing Hosting Platforms, Cluster Computing, Grid Computing, Utility Computing, Autonomic Computing, Fog Computing, Introduction to Cloud Computing, Cloud Computing history and evolution, practical applications of cloud computing for various industries, HealthCare and education, ~~economics~~ and benefits of cloud computing.

Cloud Issues and Challenges: Cloud computing issues and challenges like Security, Elasticity, Resource management and scheduling, QoS (Quality of Service) and Resource Allocation, Cost Management, Big Data, Energy Efficency, Load Balancing.

Cloud Computing Architecture: Cloud Architecture model, Types of Clouds: Public Private & Hybrid Clouds, Cloud based services: Iaas, PaaS and SaaS

Classification of Cloud Implementations: Amazon Web Services, The Elastic Compute Cloud (EC2), The Simple Storage Service (S3), DyanmoDB, The Simple Queuing Services (SQS), Google AppEngine – PaaS, Windows Azure, Aneka, A Comparison of Cloud Computing Platforms .

Virtualization: Virtualization, Advantages and disadvantages of Virtualization, Types of Virtualization: Resource Virtualization-Server, Storage and Network virtualization, Migration of processes, Classic Data Center, Virtualized Data Center (Compute, Storage, Networking and Application), Business Continuity in VDC.

Cloud based Data Storage: Introduction to Map Reduce ~~for Simplified data processing on Large clusters~~, Design of data base applications based on Map Reduce in Apache Hadoop, Task Partitioning,Data partitioning, Hadoop Schedulers, Data Synchronization, Distributed File system, Data Replication, Shared access to weakly consistent to data stores, introduction to Python.

Laboratory Work: To implement Cloud, Apache and Hadoop framework and related services on AWS. To understand various concepts practically about virtualization, data storage. To implement few algorithms with the help of MapReduce and some high level language.

Recommended Books:

1. Raj Kumar Buyya, James Broberg, Andrezei M.Goscinski, Cloud Computing: Principles and paradigms (2011)
2. Michael Miller, Cloud Computing, Que Publishing (2008).
3. Cloud Computing: A practical Approach Anthony Velte, Toby Velte and RobertElsenpeter by Tata McGrawHill
4. Judith Hurwitz, Robin Bllor, Marcia Kaufman, Fern Halper, Cloud Computing for dummies (2009).

Course Learning Outcomes (CLOs)

CLO1	To assess an existing hosting Cloud platforms and computing paradigms currently being used in industry and academia.
CLO2	To comprehend the need of data centre, its virtualization techniques and types of clouds.
CLO3	To demonstrate the implementation of cloud by using Amazon Web Services, Azure, Google App Engine. And its virtualization.
CLO4	To Know the cloud based data storage by considering issues of task partitioning, Resource Management, Energy Efficiency Security, data replication etc.
CLO5	To demonstrate the use of Hadoop framework, data storage and MapReduce using real world applications.

Evaluation Scheme:

S.No.	**Evaluation Elements**	**Weightage (%)**
4.	MST	20
5.	EST	40
6.	Sessionals (May include Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	40

PCA510 DATA ANALYTICS AND VISUALIZATON				
	L	T	P	Cr
	3	0	2	4.0

Course Objective: To learn the analysis of various types of data and its visualization using visualization tools.

Data Representation: Data Objects and Attribute Types: Nominal, Binary, Ordinal, Numeric, Discrete and Continuous, Types of data: Record, Temporal, Spatial Temporal, Graph, Unstructured and Semi structured data, Basic Statistical Descriptions of Data.

Introduction to Data Analysis: Probability and Random Variables, Correlation, Regression.

Data Analysis Pipeline: Data pre-processing- Attribute values, Attribute transformation, Sampling, Dimensionality reduction: PCA, Eigen faces, Multidimensional Scaling, Non-linear Methods, Graph-based Semi-supervised Learning, Representation Learning Feature subset selection, Distance and Similarity calculation.

Data Mining Techniques for Analysis: Classification: Decision tree induction, Bayes classification, Rule-based classification, Support Vector Machines, Classification Using Frequent Patterns, k-Nearest-Neighbor, Fuzzy-set approach Classifier, Clustering: K-Means, k-Medoids, Agglomerative versus Divisive Hierarchical Clustering Distance Measures in Algorithmic Methods, Mean-shift Clustering

Visualization: Traditional Visualization, Multivariate Data Visualization, Principles of Perception, Color, Design, and Evaluation, Text Data Visualization, Network Data Visualization, Temporal Data Visualization and visualization Case Studies.

Laboratory work: Implementation of various data analytics techniques such as classification clustering on real world problems using R.

Recommended books:

1. Han, J., Kamber, M. and Pei, J., Data Mining Concepts and Techniques, Morgan Kaufmann (2011) 3rd Edition
2. Peng, D., R., R Programming for Data Science, Lulu.com (2012).
3. Hastie, T., Tibshirani, R. and Friedman, J., The Elements of Statistical Learning, Springer (2009) 2nd Edition.
4. Simon, P., The Visual Organization: Data Visualization, Big Data, and the Quest for Better Decisions, John Wiley & Sons (2014).

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Analyze and extract features of complex datasets.
CLO2	Evaluate and visualize inter-dependencies among variables in dataset.
CLO3	Apply techniques for classification and clustering in datasets.
CLO4	Develop and validate models for real life datasets.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1.	MST	25
2.	EST	40
3.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA511 CYBER FORENSICS

L	T	P	Cr
3	0	2	4.0

Course Objective: To maintain an appropriate level of awareness, knowledge and skill required to understand and recreate the criminal terminology and Cyber Forensics investigation process.

Introduction to Cybercrime: Defining Cybercrime, Understanding the Importance of Jurisdictional Issues, Quantifying Cybercrime, Differentiating Crimes That Use the Net from Crimes That Depend on the Net, working toward a Standard Definition of Cybercrime, Categorizing Cybercrime, Developing Categories of Cybercrimes, Prioritizing Cybercrime Enforcement, Reasons for Cybercrimes

Understanding the People on the Scene: Understanding Cybercriminals, Profiling Cybercriminals, Categorizing Cybercriminals, Understanding Cyber victims, Categorizing Victims of Cybercrime, Making the Victim Part of the Crime-Fighting Team, Understanding Cyber investigators, Recognizing the Characteristics of a Good Cyber investigator, Categorizing Cyber investigators by Skill Set

Computer Investigation Process: Demystifying Computer/Cybercrime, Investigating Computer Crime, How an Investigation Starts, Investigation Methodology, Securing Evidence, Before the Investigation, Professional Conduct , Investigating Company Policy Violations, Policy and Procedure Development , Policy Violations, Warning Banners, Conducting a Computer Forensic Investigation, The Investigation Process, Assessing Evidence, Acquiring Evidence, Examining Evidence, Documenting and Reporting Evidence, Closing the Case

Acquiring, Duplicating and Recovering Deleted Files: Recovering Deleted Files and Deleted Partitions, recovering "Deleted" and "Erased" Data, Data Recovery in Linux, Recovering Deleted Files, Deleted File Recovery Tools, Recovering Deleted Partitions, Deleted Partition Recovery Tools, Data Acquisition and Duplication, Data Acquisition Tools, Recovering Data from Backups, Finding Hidden Data, Locating Forgotten Evidence, Defeating Data Recovery Techniques

Collecting and Preserving Evidence: Understanding the Role of Evidence in a Criminal Case, Defining Evidence, Admissibility of Evidence, Forensic Examination Standards, Collecting Digital Evidence, Evidence Collection, Preserving Digital Evidence, Preserving Volatile Data, Special Considerations, Recovering Digital Evidence, Deleted Files, Data Recovery Software and Documentation, Decrypting Encrypted Data, Documenting Evidence, Evidence Tagging and Marking, Evidence Logs, Documenting the Chain of Custody, Computer Forensic Resources, Computer Forensic Training and Certification, Computer Forensic Equipment and Software, Computer Forensic Services, Computer Forensic Information, Understanding Legal Issues, Searching and Seizing Digital Evidence

Building the Cybercrime Case: Major Factors Complicating Prosecution, Difficulty of Defining the Crime, Jurisdictional Issues, The Nature of the Evidence, Human Factors, Overcoming Obstacles to Effective Prosecution, The Investigative Process, Investigative Tools, Steps in an Investigation, Defining Areas of Responsibility

Laboratory Work: Hands with open source tools for forensic investigation process models (from Item confiscated to submitting evidence for lawful action), such as FTK, Sleuth Toolkit (TSK), Autopsy, etc.

Recommended books:

1. Shinder L. D., Cross M., Scene of the Cybercrime, Syngress (2008) 2nd ed.
2. Marcella J. A. and GuillossouF.,Cyber Forensics: From Data to Digital Evidence, Wiley (2012).

3. Marcella J. A. and Menendez D., Cyber Forensics: A Field Manual for Collection, Examining and preserving Evidence of computer crimes, Auerbach Publication (2010), 2nd ed.

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Familiarize with cybercrime and forensics ontology.
CLO2	Analyse and demonstrate the crime scene and criminology.
CLO3	Redesign the crime scene using digital investigation process
CLO4	Recovery of evidence and creating document for judicial proceedings.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
4.	MST	25
5.	EST	40
6.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA512 WEB ANALYTICS AND INTELLIGENCE

L	T	P	Cr
3	0	2	4.0

Course Objective: This subject enables to examine how website visitors view and interact with a site's pages and features, and business intelligence, which would allow using data on customer purchasing patterns, demographics, and demanding trends to make effective strategic decisions.

Introduction: Definition, Process, Key terms: Site references, Keywords and Key phrases; building block terms: Visit characterization terms, Content characterization terms, Conversion metrics; Categories: Offsite web, On site web; Web analytics platform, Web analytics evolution, Need for web analytics, Advantages, Limitations.

Data Collection: Clickstream Data: Web logs, Web Beacons, JavaScript tags, Packet Sniffing; Outcomes Data: E-commerce, Lead generation, Brand/Advocacy and Support; Research data: Mindset, Organizational structure, Timing; Competitive Data: Panel-Based measurement, ISP-based measurement, Search Engine data.

Qualitative Analysis: Heuristic evaluations: Conducting a heuristic evaluation, Benefits of heuristic evaluations; Site Visits: Conducting a site visit, Benefits of site visits; Surveys: Website surveys, Post-visit surveys, Creating and running a survey, Benefits of surveys.

Web Analytic fundamentals: Capturing data: Web logs or JavaScripts tags, Separate data serving and data capture, Type and size of data, Innovation, Integration, Selecting optimal web analytic tool, Understanding clickstream data quality, Identifying unique page definition, Using cookies, Link coding issues.

Web Metrics: Common metrics**:** Hits, Page views, Visits, Unique visitors, Unique page views, Bounce, Bounce rate, Page/visit, Average time on site, New visits; Optimization (e-commerce, non e-commerce sites): Improving bounce rates, Optimizing adwords campaigns; Real time report, Audience report, Traffic source report, Custom campaigns, Content report, Google analytics, Introduction to KPI, characteristics, Need for KPI, Perspective of KPI, Uses of KPI.

Web analytics 2.0: Web analytics 1.0, Limitations of web analytics 1.0, Introduction to analytic 2.0, Competitive intelligence analysis : CI data sources, Toolbar data, Panel data ,ISP data, Search engine data, Hybrid data, Website traffic analysis: Comparing long term traffic trends, Analyzing competitive site overlap and opportunities.

Google Analytics: Brief introduction and working, Adwords, Benchmarking, Categories of traffic: Organic traffic, Paid traffic; Google website optimizer, Implementation technology, Limitations, Performance concerns, Privacy issues.

Relevant technologies: Internet & TCP/IP, Client / Server Computing, HTTP (HyperText Transfer Protocol), Server Log Files & Cookies, Web Bugs.

Laboratory Work: Lab assignments cover the basic fundamentals of analyzing the websites, understanding their basic features. It also involves use of related tools and technologies in this domain.

Recommended books:

1. Clifton B., Advanced Web Metrics with Google Analytics, Wiley Publishing, Inc. (2010), 2nd ed.

2. Kaushik A., Web Analytics 2.0 The Art of Online Accountability and Science of Customer Centricity, Wiley Publishing, Inc. (2010),1st ed.
3. Sterne J., Web Metrics:Proven methods for measuring web site success, John Wiley and Sons (2002),1sted

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Understand the basic concepts of web analytics, its goals, key terminologies, advantages, limitations, various data collection methods for the purpose of analysis.
CLO2	Perform qualitative analysis using heuristic evaluation and surveys. To explore and understand fundamentals of web analytics using different tools.
CLO3	Explore various web metrics and understand what metrics to optimize, understanding key performance indicators, their role, and how to implement them for success.
CLO4	Understanding the limitations of web 1.0 and to explore web analytic 2.0 using search engine data, Analysis of traffic website.
CLO5	Understanding Google analytics, Implementation technology, and it's working mechanism.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
7.	MST	25
8.	EST	40
9.	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35

PCA513 NATURAL LANGUAGE PROCESSING				
	L	T	P	Cr
	3	0	2	4.0

Course Objective: To understand the basic concepts of Natural Language Processing (NLP). The student must be able to apply the various concepts of NLP in other application areas.

Introduction: Origin of Natural Language Processing (NLP), Challenges of NLP, NLP Applications, Processing Indian Languages.

Words and Word Forms: Morphology fundamentals; Morphological Diversity of Indian Languages; Morphology Paradigms; Finite State Machine Based Morphology; Automatic Morphology Learning; Named Entities.

Phrase structure and constituency models: phrase structure grammar; dependency grammar; formal language theory.

Parsing: Definite clause grammars; shift-reduce parsing; chart parsing' Shallow Parsing, Statistical Parsing, Maximum Entropy Models; Random Fields, Scope Ambiguity and Attachment Ambiguity resolution, Approaches to discourse, generation.

Language Modeling and Part of Speech Tagging: Markov models, N-grams, estimating the probability of a word, and smoothing, Parts-of-speech, examples and its usage.

Machine Translation: Need of MT, Problems of Machine Translation, MT Approaches, Direct Machine Translations, Rule-Based Machine Translation, Knowledge Based MT System, Statistical Machine Translation.

Meaning: Lexical Knowledge Networks, WorldNet Theory; Indian Language Word Nets and Multilingual Dictionaries; Semantic Roles; Word Sense Disambiguation; WSD and Multilinguality; Metaphors.

Other Applications: Sentiment Analysis; Text Entailment; Question Answering in Multilingual Setting; NLP in Information Retrieval, Cross-Lingual IR. Text-classification.

Laboratory Work: To implement Natural language concepts and computational linguistics concepts using popular tools and technologies. To implement key algorithms used in Natural Language Processing. To implement various machine translations techniques for Indian languages.

Reference Books:

1. Jurafsky, D. and Martin J. H., Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. Upper Saddle River, NJ: Prentice-Hall, 2000.
2. Manning, Christopher D., and HinrichSchütze. Foundations of Statistical Natural Language Processing, Cambridge, MA: MIT Press, 1999.
3. Dale R., Moisl H. and Somers H., Handbook of Natural Language Processing, MARCEL DEKKER, Inc., 2009
4. Bird, S.K. Ewan and Loper E., Natural Language Processing withPhthon, Oreilly Publication, 2009

*

COURSE LEARNING OUTCOMES (CLOs): On completion of this course, students will be able to

CLO1	Comprehend the concept of natural language processing, its challenges and applications.
CLO2	Comprehend the concepts word forms of the language by considering the concept of morphology analysis.
CLO3	Perform syntax and semantics in natural language processing.
CLO4	Design and 83nalyse the NLP algorithms.
CLO5	Acquire knowledge of machine learning techniques used in NLP, including hidden Markov models, N-Grams and probabilistic context-free grammar.

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	25
2	EST	40
3	Sessionals (Assignments/Projects/Tutorials/Quizzes/Lab Evaluations)	35