SYLLABUS STRUCTURE

FOR

TWO-YEAR MASTERS PROGRAMME

IN

MASTER OF COMPUTER SCIENCE AND APPLICATION

(NBA Accredited)

NAAC – A Grade

DEPARTMENT OF COMPUTER SCIENCE AND APPLICATION

COLLEGE OF ENGINEERING & TECHNOLOGY

(An Autonomous and Constituent College of BPUT, Odisha)

Techno Campus, MahalaxmiVihar, Ghatikia,

Bhubaneswar-751029, Odisha, INDIA

www.cet.edu.in

Ph. No.: 0674-2386075 (Off.), Fax: 0674-2386182

COURSE: MCA (CSA-Master of Computer Application)
Duration: 2 years (Four Semesters)

Abbreviations Used: **AR: Academic Regulation** **U = UG, I = Integrated, P = PG**
L = Lectures **P = Practical or Laboratory** **IA = Internal Assessment**
T = Tutorial **SE = Semester Examination** **EA = External Assessment**

Subject Code Format:

1	2	3	4	5	6	7	8
Prog (U/I/P)	Type (PC/PE/OE/LC/MC/AC)		Department (CE/EE/IE/ME/…)		Semester (1/2/…/0)	Serial No. (1/2/3/…/99)	

1st SEMESTER

Sl. No.	Subject Type	Subject Code	Subject Name	Teaching Hours			Credit	Maximum Marks			
				L	T	P		IA	SE	EA	Total
1	Core 1	PPCCA101	Introduction to Computation	3	0	0	3	30	70	-	100
2	Core 2	PPCCA102	Computer Organization and Architecture	3	0	0	3	30	70	-	100
3	Core 3	PPCCA103	Web Design and Development	3	0	0	3	30	70	-	100
4	Core 4		Computational Mathematics & Statistics	3	0	0	3	30	70	-	100
5	Mandatory 1	PMCMH105	Business Communication	3	0	0	3	30	70	-	100
6	Lab 1	PLCCA101	Introduction to Computation Lab	0	0	2	1	100	-	-	100
7	Lab 2	PLCCA103	Web Design and Development Lab	0	0	2	1	100	-	-	100
8	Lab 3	PLCMH105	Business Communication Lab	0	0	2	1	100	-	-	100
			Total	**15**	**0**	**6**	**18**	**450**	**350**	**-**	**800**

2nd SEMESTER

Sl. No.	Subject Type	Subject Code	Subject Name	Teaching Hours			Credit	Maximum Marks			
				L	T	P		IA	SE	EA	Total
1	Core 5	PPCCA201	Data Structures using C	3	0	0	3	30	70	-	100
2	Core 6	PPCCA202	Programming with Java	3	0	0	3	30	70	-	100
3	Core 7	PPCCA203	Database Management System	3	0	0	3	30	70	-	100
4	Core 8	PPCCA204	Operating Systems	3	0	0	3	30	70	-	100
5	Lab 4	PLCCA201	Data Structures Lab	0	0	2	1	100	-	-	100
6	Lab 5	PLCCA202	Programming with Java Lab	0	0	2	1	100	-	-	100
7	Lab 6	PLCCA203	Database Management System Lab	0	0	2	1	100	-	-	100
8	Lab 7	PLCCA204	Python Lab	0	0	4	2	100	-	-	100
9	Project-1	PPRCA201	Minor Project-I	0	0	2	1	100	-	-	100
			Total	**12**	**0**	**08**	**18**	**620**	**280**	**-**	**900**

Students must undertake a summer internship program at the end of the 2nd semester which will be evaluated in 3rd semester.

3rdSEMESTER

Sl. No.	Subject Type	Subject Code	Subject Name	Teaching Hours			Credit	Maximum Marks			
				L	T	P		IA	SE	EA	Total
1	Core 5	PPCCA301	Artificial Intelligence	3	0	0	3	30	70	-	100
2	Core 6	PPCCA302	Computer Networks	3	0	0	3	30	70	-	100
3	Core 7	PPCCA303	Software Engineering	3	0	0	3	30	70	-	100
4	Core 8	PPCCA304	Design and Analysis of Algorithms	3	0	0	3	30	70	-	100
5	Elective 1 (Any One)	PPECA301	Theory of computation	3	0	0	3	30	70	-	100
		PPECA302	Computer security								
		PPCCA303	Machine Learning								
		PPECA304	Computer Graphics								
6	Lab-08	PLCCA302	Enterprise Java Technologies	0	0	4	2	100	-	-	100
7	Project-2	PPRCA301	Minor Project-II	0	0	2	1	100	-	-	100
8	Project -3	PPRCA302	Summer Internship Evaluation	0	0	2	1	100	-	-	100
9	Mandatory 2	PMCMH301	Group Discussion and Seminar	0	0	2	1	100	-	-	100
			Total	**15**	**0**	**10**	**20**	**550**	**350**	**-**	**900**

4thSEMESTER

Sl. No.	Subject Type	Subject Code	Subject Name	Teaching Hours			Credit	Maximum Marks			
				L	T	P		IA	SE	EA	Total
1	Elective 2 (Any One) MOOCS	PPECA401	Introduction to Industry 4.0 and Industrial Internet of Things								
			Applied Natural Language Processing								
			Cryptography and Network security								
			Compiler Design								
2	Elective 3 (Any One) MOOCS	PPECA402	Deep Learning								
			Social Networks								
			Software Project Management								
			Software Testing								
3	Project 4	PPRCA401	Industrial Training cum Project	0	0	25	12		-	-	700
			Total	**0**	**0**	**25**	12		-	-	700

DETAILED SYLLABUS

1st SEMESTER

Introduction to Computing

Code: PPCCA101 **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To provide understanding of algorithmic approach to problem solving.
2. To provide knowledge on Procedural as well as Object Oriented Approaches to program design.
3. To provide elaborate knowledge on C language to write procedural programs.
4. To introduce relevant features of C++ language to write object oriented programs.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Develop skills to write computer programs to solve a variety of real-world problems.

2. Write programs using both procedural and object oriented approaches
3. Design programs using readable, reusable and cohesive modules.
4. Develop skills to use pointers and data files in programs.

Course Prerequisites:

1. This course does not require any prerequisite as such.

Detailed Syllabus:

Module I (10Hours)

Introduction to Computer: Basic Organization of a Computer, Hardware and Software, Programming Languages, Number System, Conversion.

Program Development: Programming as Problem-Solving, Steps in Program Development, Algorithm, Flowchart, Pseudo code, Top-down and Bottom-up approaches, Characteristics of a good program, Structure of a CProgram, Compiling, Linking and Executing Programs.

C Language Fundamentals: Language Elements, Data Types, Variables andConstants, Operators, Expressions, Type Conversions, Statements, Managing Console Input andOutput Operations, Function.

Control Structures: Decision Making and Branching - If and Switch, Loop Structures - While, Do While and For, Unconditional Jumps - Continue, Break and Go To.

Module II (10Hours)

Arrays and Strings: Concept, Declaration and Manipulation of Arrays, One Dimensional and Multidimensional Arrays, Sorting and Searching an Array, Concept of Strings, String Handling Functions, Array of Strings.

Pointers: Pointer Variable and its Importance, Dereferencing, Pointer Arithmetic and Scale Factor,Pointers and Arrays, Pointer and Strings, Array of Pointers, Pointers to Pointers.

Functions: Designing Structured Programs, User Defined and Standard Functions, Formal andActual Arguments, Function Prototype, Parameter Passing, Functions Returning Multiple Values,Functions Returning Pointers, Pointers to Functions, Nesting of Functions, Recursion, PassingArrays to Functions.

Scope and Extent: Scope Rules, Storage Classes - Auto, Extern, Register and Static.

Module III (10 Hours)

Structures, Unions and Enumerations: Declaration and Initialization of Structures, Structure asFunction Parameters, Structure Pointers, Unions, Enumerations.

File I/O: Defining, Opening a File and Closing a File, Input/output Operations in Files,Random Access to Files, Error Handling.Command Line Arguments, Dynamic Memory Management, Pre-Processor Directives.

Module IV (10 Hours)

Introduction: Need of Object Orientation, Basic Concepts of Object Oriented Approach, BasicProgram Construction in C++, Namespace, Data Types, Input and Output, Handling Exceptions.

Objects and Classes: Defining and Using Classes, Constructors andDestructors, Controlling Accessibility, Public and Private Class Members, Member Functions, *this* pointer, *static*class data and*const*Member Functions, Constructor and Function Overloading.

Inheritance: Base and Derived classes, Access Control Mechanisms, Types of Inheritance, VirtualFunctions, Abstract Class and Pure Virtual Function, Virtual Base Class.

Text Book:

1. Jeri R. Hanly and Elliot B. Koffman, Problem Solving and Program Design in C, 8th Edition, Pearson Education, 2016. (Module-I, II, III)
2. The C++ Programming Language, Bjarne Stroustrup, Addition Wesley. (Module-IV)

Reference Books:

1. R. G. Dromey, How to Solve it by Computer. Prentice-Hall India EEE Series.
2. E. Balagurusamy, Programming in ANSI C, 4th edition, McGraw-Hill Publication,2007.
3. PradipDey, ManasGhosh, Programming in C, Second Edition,OxfordUniversity Press, 2011.
4. Brian W. Kernighan, Dennis Ritchie, The C Programming Language, 2nd Edition, Prentice Hall, 1988.
5. Yashavant P. Kanetkar. Let Us C, BPB Publications, 2011.
6. Byron S Gottfried, Programming with C, Schaum's Outlines, Second Edition, Tata McGrawHill, 2006.
7. Bruce Eckel, Thinking in C++, Vol. 1: Introduction to Standard C++, 2nd Edition,

Computer Organizations andArchitecture

Code: PPCCA102 **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To study the basic *organization* and architecture of digital computers
2. To study design aspects of different subsystems of a computer system
3. To understand the instructions and instruction execution life cycle
4. To understand various data transfer techniques in digital computer.
5. To understand processor performance improvement using instruction level parallelism
6. To understand microprocessor and assembly language program

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand basic structure of a computer.
2. Understand computer instruction and its execution
3. Understand ALU, design basic circuits and perform computer arithmetic operations.
4. Understand different memory and their performance issues
5. Understand cache mapping techniques.
6. Understand I/O organization and data transfer techniques.
7. Understand processor performance and instruction level parallelism.
8. Write basic assembly language programs

Course Prerequisites:

This course does not require any prerequisite as such.

Detailed Syllabus:

Module 1 (12 Hours)

Introduction: Basic architecture of computer, Functional units, Operational concepts, Bus structures, Von Neumann Concept.

Basic Processing: Instruction code, Instruction set, Instruction sequencing, Instruction Cycle & Execution Cycle, Instruction format, Addressing modes, Micro instruction, Data path and control path design, Micro programmed vs. Hardwired controlled unit, RISC vs. CISC.

Arithmetic: Design of ALU, Binary arithmetic, Addition and Subtraction of signednumber, Multiplication of Positive number, Signed operand multiplication, Division, Floating point number representation andarithmetic.

Digital Electronics: Boolean algebra, Digital Logic, Truth Tables, K map, Number system, Flip- Flop

Module 2 (10 Hours)

Memory: Memory Hierarchy, RAM, ROM, Cache memory organization, Mapping techniques, Virtual memory, Memory Interleaving, Secondary Storage, Flash drives.

Module 3(10 Hours)

Input/output: Accessing I/O devices, I/O mapped I/O, Programmed I/O, Memory Mapped I/O, Interrupt Driven I/O, Standard I/O interfaces, Synchronous and Asynchronous Data transfer, DMA data transfer.

Introduction to Parallel processing: Flynn's Classification, Pipelining, Super Scalar processors, Array processing, vector processing.

Module 4 (8 Hours)

8085 Microprocessor and Assembly level Programming using 8085 microprocessor

Text Books:

1. William Stalling, Computer Organization and Architecture, PearsonEducation (Module I, II, II)
2. M. Mano, "Computer System and Architecture", PHI. (Module IV)

Reference Books:

1. J. P. Hayes, "Computer Architecture andOrganization", MGH
2. A.S. Tananbaum, "Structured Computer Organization", PearsonEducation
3. Alan Clements, Computer Organization and Architecture, Cengage.
4. C. Hamacher, Z. Vranesic, S. Zaky, "Computer Organization", McGraw-Hill Education India

Web Design and Development

Code: PPCCA103 **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To provide basic understanding of the Internet and World Wide Web.
2. To provide elaborate knowledge on how to use HTML, CSS and JavaScript to develop webpages.
3. To provide understanding on Front-End Libraries such as jQuery and Bootstrap to develop webpages.
4. To introduce JavaFX for GUI Development
5. Develop skills to write Desktop Applications involving GUI .

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Develop web pages using core front-end coding languages such as HTML, CSS and JavaScript.
2. Create Responsive Websites compatible with different devices and screen sizes.
3. Display ability to develop professional websites using Bootstrap and jQuery.
4. Create and maintain websites on the Internet.

Course Prerequisites:

Basic understanding of computer and programming.

Detailed Syllabus:

Module-I (13hrs):

Introduction to Internet and World Wide Web: Introduction to Internet, client- server model, IP address, protocols, Basic Services, the Internet verses the World Wide Web, Domain Name, URL, Evolution of World Wide Web, Web 2.0.

Page Structuring using HTML: Structure of a webpage, Basic formatting markups, Adding links, images, Table markup, Lists, Forms, Div and Span, Semantic markups in HTML 5.

Basics of Web Graphics: Image formats, Size and Resolution, Transparency, Scalable Vector Graphics, Image Optimization.

Presentation using CSS: Overview of CSS, benefits of CSS, Basic syntax of writing style rules, Selectors, Types of Style sheet, Inheritance and Cascading styles, Text and Font properties, Color and Background properties, Box Model, Page Layout, Floating and Positioning, Styling forms and tables, Basic responsive web design.

Module-II (14hrs):

Page Interaction using JavaScript: Introduction to JavaScript, Adding JavaScript to a page, Basics of JavaScript Language, variable, data types, operators, array, control structures, Browser objects, Events, Document Object Model, Accessing page contents using JavaScript, Form validation using JavaScript,

Introduction to XML and AJAX: Basics of XML document, DTD, Schema, XMLHttpRequest object, Sending request and receiving server response using AJAX.

Introduction to jQuery: Basics of jQuery, Selecting elements, Handling events, Applying effects and animations, Manipulating DOM, jQuery and AJAX.

Module-III (13 hrs):

Introduction to Bootstrap: Overview of Bootstrap, Grid basics, Using Bootstrap Base CSS, Typographic elements, Colors, Images, Buttons, Navs, Navbar, Carousel, Forms.

GUI Development: AWT Classes, Window fundamentals, working with graphics, working with color & fonts. Event handling in Java, Delegation Event Model, Swing Package: JFrame, JPanel, swing GUI controls, layout managers, working with menus, Introduction to JavaFX

Main Texts:

1. The Web Warrior Guide to Web Design Technologies, Don Gosselin, et. al, Cengage Learning

Recommended Texts:

1. Learning Web Design, Jennifer N. Robbins, O'Reilly Media
2. HTML & CSS: Design and Build Websites, Jon Duckett, John Wiley
3. jQuery IN ACTION, Bear Bibeault et.al., Dreamtech
4. Learn Java GUI Applications - 11th Edition: A JFC Swing Tutorial 11th Edition by Philip Conrod (Author), Lou Tylee (Author)

Computational Mathematics and Statistics

Code: **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To understand various concepts in several areas of discrete mathematics and statistics.
2. To develop problem-solving techniques using those mathematical and statistical concepts.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Represent various real-world problems using mathematical and statistical concepts.
2. Use those mathematical and statistical methods in practical applications.

Course Prerequisites:

This course does not require any prerequisite as such.

Detailed Syllabus:

MODULE-I Matrix & Vector Space (12 HOURS)

Introduction to Vectors: Vectors and Linear combinations, Dot Products, Matrices

Solving Linear Equations: Vectors and Linear Equations, Elimination using Matrices, Rules for Matrix Operations, Inverse of Matrices.

Vector Spaces and Subspaces: Spaces of Vectors, The Nullspace of A: Solving Ax=0, Complete solution to Ax=b, Independence, Basis and Dimension.

MODULE-II Numerical Metods (12 HOURS)

Computing errors, Significant Digits, Root finding methods: Bisection, Newton Raphson, and Secant method, Interpolation: Newton's formula for Interpolation, Lagrange's Interpolation formula.

Differentiation and Integration: First and second order ordinary differentiation, Trapezoidal and Simpsons method.

MODULE-III Statistics (16 HOURS)

Understand the Type of Analytics, Probability: Probability concepts, Sampling Concepts, Generating Random Variables, Central Tendency, Variability, Probability Distribution, Hypothesis Testing and Statistical Significance, Regression: Least squares estimation, Analysis of variance, prediction.

TEXT BOOKS:

1. Gilbert Strang, "Introduction to Linear Algebra", WELLESLEY - CAMBRIDGE PRESS, 5th Edition,2016
2. S. S. SASTRY, " **INTRODUCTORY METHODS OF NUMERICAL ANALYSIS**", PHI Learning Pvt. Ltd. Fifth Edition, 2012
3. Wendy L. Martinez, Angel R. Martinez "Computational Statistics Handbook with MATLAB®", CHAPMAN & HALL/CRC, 2002

Business Communication

Code: PMCMH105 **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To introduce students to various building blocks of communication, both within and outside their formal articulations.
2. To train students in the basic science of writing and help them use the same in various sites such as report, paragraph etc.
3. To create conditions in the classroom that encourages students to engage in meaningful conversation.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Develop skills to communicate effectively in formal settings.
2. Develop skills to write CV, Report, Minutes, Business Letters etc.
3. Develop skills to present effectively on topical issues.

Course Prerequisites:

This course does not require any prerequisite as such.

Detail Syllabus

Module - I

Basics of Communication in Practice (12 hours)

1.1 Types of Communication in an organization: Formal (internal and external) and Informal (grapevine) (1 Hour)

1.2 Communication Channels: Upward, Downward, Diagonal and Horizontal (1 Hour)
1.3 Introduction to cross-cultural communication. (2 Hour)
1.4 Bias-free communication & use of politically correct language in communication (1 Hour)
1.5 Importance of reading and ethics of writing (1Hour)
1.6 Negotiation Skills, Argumentation & Consensus building.

Module-II
Business Writing (14 hours)
2.1 Skills of Writing: Coherence, Cohesion, Sentence Linkers, Clarity of Language and stylistic variation, process of writing.(3 Hour)
2.2 Paragraph writing: Topic Sentence, Supporting sentence &Concluding sentence, Logical structuring (Inductive approach and deductive approach) (2 Hour)
2.3 Letters, Applications (2 Hour)
2.4 Reports and Proposals (1 Hour)
2.5 Memos, Notices, Summaries, Abstracts& e-mails (1 Hour)
2.6 Writing a CV/Resume': Types of CV (2 Hour)
2.7 Writing a Cover letter (1 Hour)

Module -III
Speaking and Presentation (14 Hours)
3.1 Oral Presentation: 4 P's of presentation, PPT (2 Hour)
3.2 Group Discussion: Structured and Un-structured, Various types of topics (abstract, absurd, contemporary etc.) (3 Hour)
3.3 Types of Interview: Preparing an Interview and techniques (2 Hour)
3.4 Grooming and dress code, Personality development (2 Hour)

Introduction to computation Laboratory

Code: PLCCA101 **Credit: 1** **Hours: 2/week**

Note: This course shares the objectives and outcomes of its associated theory course PPCCA101. Suitable execution environment preferably Linux will be used to carry out laboratory exercises. Exercises will primarily follow algorithmic approach as provided in reference book serial number 1. The programs will follow proper modeling either function-oriented or object-oriented as the case may be. The exercises suggested below are illustrative in nature. Additional exercises suitably may be suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Write a program that will exchange the values of three variables a, b and c as follows: the variable *b* will hold value of *a*, *c* will hold the value of *b* and *a* will hold the value of the variable *c*.
2. Write a program which reads a set of marks in an examination, count the number of pass marks, number of fail marks, percentage of pass and fail.
3. Write a program to find the harmonic mean of a set of n numbers.

4. Write a program to count the number of digits in an integer.
5. Write a menu based program to find LCM and GCD of a set of numbers
6. Write a program to convert binary numbers to octal and binary numbers to decimal.
7. Write a program to find the maximum, the minimum and how many times they both occur in an array of numbers.
8. Write a program to find the k^{th} smallest element in an array of numbers.
9. Write a menu based program to implement sorting algorithms.
10. Write a program to perform operations such as multiplication and transpose on matrices.
11. Write programs involving string manipulation (to be decided by faculty).
12. Write programs involving pointer arithmetic (to be decided by the faculty)
13. Write a program involving recursive function (to be decided by the faculty)
14. Write programs involving structures and unions (to be decide by the faculty)
15. Write programs involving data files (to be decided by the faculty)
16. Write an Object Oriented Program to find the area and perimeter of a circle.
17. Write a menu based Object Oriented Program to perform operations in a bank account.
18. Write OO programs to implement overloading of function and constructor.
19. Write an OO program to process student results in an examination. Model students and marks scored in an examination as classes using has-a relation. Print the grade sheet of a student.
20. Write an OO program to find the area and perimeter of different shapes such as rectangles, triangles and squares. Use inheritance appropriately.

Web Design and Development Laboratory

Code: PLCCA103 **Credit: 1** **Hours: 2/week**

Note: This course shares the objectives and outcomes of its associated theory course PPCCA103. Suitable page authoring IDEs and development tools will be used to carry out laboratory exercises.

List of Exercises:

1. Develop a webpage to display your curriculum vitae. Use appropriate markups to structure the page.
2. Develop a user registration form.Use appropriate form controls for submitting user information.
3. Use CSS in the curriculum vitae developed earlier to enhance its presentation.
4. Use CSS in the registration form to enhance its appearance.
5. Use JavaScript to validate data of the registration form at the client-side.
6. Develop XML document for a list of Books in a library
7. Use AJAX to develop a webpage that fetches data from the server.
8. Use jQuery to display a list of students and their details in a table.
9. Use Bootstrap to develop a website for your department.
10. Write GUI programs using basic swing classes
11. Write GUI program involving Menus
12. Write a GUI program using JavaFX

Communication in Practice Lab

Code: PLCMH105 **Credit: 1** **Total Hours: 2/week**

Course Objectives:

- To enable the students engage in polite, negotiating and argumentative conversation.
- To train the learners in writing CV, Report, Minutes, Business Letters etc.
- To give students an opportunity of power point presentation relating to topical issues.

There will be 10 lab sessions of 2 hours each. Lab sessions will be used to give the students an in-hand experience of communication taking place in an organization. This will help the students to understand the requirement of communication in the workplace. Students will be encouraged to brush-up themselves in activities based on all the modules of theory taught in the class room. Special emphasis will be given to speaking and writing business correspondences.

Ist session:
Speaking: Greeting an acquaintance/ friend, introducing oneself, introducing a third person to a friend, breaking off a conversation politely, leave-taking, Describing people, objects, places, processes etc. (1 Hour), Writing an application (1 Hour)

IInd session:
Speaking: making and responding to inquiries; expressing an opinion; expressing agreement/ disagreement, contradicting/ refuting an argument; expressing pleasure, sorrow, regret, anger, surprise, wonder, admiration, disappointment etc (1 Hour), Writing an informal letter/Business Letter (1 Hour)

IIIrd session:
Speaking: Narrating or reporting an event (1 Hour), Writing a Report (1 Hour)

IVth session:
Speaking: Ordering / directing someone to do something, Making requests; accepting / refusing a request, Expressing gratitude; responding to expressions of gratitude, Asking for or offering help; responding to a request for help, Asking for directions (e.g. how to reach a place, how to operate a device etc.) and giving directions, Speaking: asking for and granting/ refusing permission, prohibiting someone from doing something, suggesting, advising, persuading, dissuading, making a proposal, praising, complimenting, felicitating, expressing sympathy (e.g. condolence etc.), Complaining, criticizing, reprimanding etc., (1 Hour), Writing a proposal (1 Hour)

Vth Session:
Speaking:Understanding and interpreting graphs, flowcharts, pictograms, pictures, curves etc., (1 Hour), Writing: Describing, explaining and interpreting graphs, flowcharts, pictograms, pictures, curves etc.

VIth session:
Speaking: Group discussion (1 Hour), Writing a memo, notice and circular (1 Hour)

VIIth session:

Speaking: Public speaking, in-house communication on work-related situations (1 Hour), Writing a CV (1 Hour)

VIIIth session: Presentation 1 (Students will make and present a topic in power point on a pre-assigned topic) (1 Hour),Writing an e-mail (1 Hour)

IXth session: Presentation 2 (Students will make and present a topic in power point on a pre-assigned topic) (1 Hour), Writing an abstract (1 Hour)

Xth session: Presentation 3 (Students will make and present a topic in power point on a pre-assigned topic) (1 Hour), Writing a summary (1 Hour)

Note: 70 marks will be devoted for sessions, 10 marks for record submission, 10 marks for viva-voce and 10 marks for project work.
End term assignment: Students are required to make a review report of at least 5 pages on a topic of their own choice (The topic should be pre-approved by teacher).

2nd SEMESTER

Data Structures

Code: PPCCA201 **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To provide knowledge and understanding of various basic and advanced data structures available in computing domain.
2. To provide skills to write programs to implement various data structures using procedural or object oriented programming languages.
3. To provide knowledge to analyze problems in application domains and design solution using data structures.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Be well-versed in various standard data structures available in computing domain.
2. Write programs to perform operations like searching, insertion, deletion, traversing mechanism etc. on various data structures.
3. Analyze problems; choose the appropriate data structures and write program solutions to problems in specified applications using such data structures.

Course Prerequisites:

This course requires understanding of computer programming in any language.

Detailed Syllabus:

Module 1 (10 hours)
Fundamentals: Introduction to Data Structures, Classification of Data Structures, Algorithms, Measuring Space and Time Complexities, Asymptotic Notations, Abstract Data Types.
Arrays: Storage Structures for Arrays, Sparse Matrixes, Strings, Pattern Matching.
Stacks and Queues: Representation, Operations on Stacks and Queues, Applications of Stack and Queues.
Linked Lists: Dynamic Memory Management, Single Linked Lists, Double Linked Lists,
Circular Linked Lists, Linked Stacks and Queues, Operations on Polynomials.
Module 2 (10 hours)
Trees: Terminology, Representation, Binary Trees, Binary Search Trees, Searching, Insertion and Deletions Operations in a Binary Search Tree, Height Balanced Trees, M-way Search Trees, B-Trees, B+ Trees, General Trees, Representation of General Trees and Binary Trees, Forests, Application of Trees.
Module 3 (16 hours)
Graphs: Terminology, Representation, Path Matrix, Graph Traversal, Shortest Path Problems, Topological Sort.
Searching and Sorting Techniques: Linear and Binary Search, Bubble Sort, Insertion Sort, Selection Sort, Quick Sort, Merge Sort, Heap and Heap Sort, Radix Sort, Comparison of Sorting Techniques.
Hashing: Hash Functions and Hashing Techniques, External sorting.

Text Books:

1. Tremblay, Jean-Paul, and Paul G. Sorenson, "An introduction to data structures with applications", McGraw-Hill.
2. Aaron M. Tenenbaum, Data Structures Using C

Reference Books:

1. Richard F. Gilberg & Behrouz A. Forouzan, Data Structures A Psedocode Approach with C, Second Edition, CENGAGE Learning.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, “Fundamentals of Data Structures in C”, Universities Press Pvt. Ltd.
3. Seymour, Lipchitz. “Data Structures with C.” TMH.

Programming with Java

Code: PPCCA202 **Credit: 3** **Total Hours: 36**

Course Objectives:

1. To provide elaborate knowledge on standard Java language.
2. To provide knowledge on Object Oriented Approach to program design.
3. To provide adequate knowledge on basic Server-side Java technology such as Servlet.
4. To understand Java framework such as Spring.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Develop skills to write Java programs to solve a variety of real-world problems.
2. Write programs using object oriented approach and standard Java
3. Develop client server applications using network sockets
4. Design programs using readable, reusable and cohesive modules.
5. Develop web applications using Servlet.
6. Develop web applications using Spring framework.

Course Prerequisites:

This course does not require any prerequisite as such.

Detailed Syllabus:

Module 1 (10 Hours)

Features of Java, Data types, operators & expressions, control structures, arrays, Classes, objects & methods, constructors, garbage collection, access qualifiers, Overloading, String Handling – string operations, character extraction, string comparison, searching and modifying strings, String Buffer, String Builder, Packages, Interfaces, Wrapper classes, Static variables and methods.

Module 2 (14 Hours)

Inheritance: single and multilevel inheritance, method overriding, abstract class, use of super and final keywords**.**

Exception Handling: Exception types, uncaught exceptions, multiple catch clauses, nested try statements, built-in exceptions, creating your own exceptions.

Multithreading: Java thread model, creating multiple threads, thread priorities, synchronization, inter-thread communication, suspending, resuming and stopping threads; Familiarity with Java Collection Framework.

I/O Streams: Console I/O, Files I/O – Byte Streams, Character Streams, Object Serialization; Socket Programming: TCP Socket, Datagram Socket.

Module 3 (12 Hours)

JDBC programming: JDBC Drivers, Creating connection, executing queries and stored procedures, handling database transactions.

Enterprise Java Technologies: Web Applications, Servlet Overview, Servlet API, Writing HelloWorld Program using Servlet, Servlet Life Cycle, Configuring Servlet in web.xml, Retrieving information from Request object, HTML form processing using Servlet, Servlet Initialization, Session tracking, Cookies, Database Access using Servlet, Error Handling, Servlet Collaboration, Forward verses Redirect.

Spring Framework: Spring Framework, Dependency Injection, Spring JdbcTemplate, Spring MVC, Spring MVC Validation, MVC Form Tag Library, Spring Remoting.

Text Books:

1. Liang Y. Daniel, Introduction to Java Programming, Pearson Education.
2. Herbert Schildt, The Complete Reference Java 2, Tata McGraw Hill
3. Java Server Programming (Java EE 7) Black Book, by DT Editorial Services, Dreamtech Press, 2015.

Reference Books:

1. E. Balaguruswami, Programming with Java, Tata McGraw Hill.
2. Mughal K.A., Rasmussen R.W., A Programmer's Guide to Java Certification, Addison-Wesley
3. Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans, Kim Haase, William Markito, "The Java EE 7 Tutorial", 5th Edition, Addison-Wesley Professional, Pearson India, 2014.

Database Management System

Code: PPCCA203 **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To provide understanding on fundamental concepts of Relational database systems
2. To provide knowledge on Modeling and Design of Relational Databases
3. To provide elaborate knowledge on how to query databases
4. To provide understanding on database transactions

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand basic concepts of Relational databases management systems
2. Model data requirements of real-world applications
3. Develop databases for a variety of applications
4. Write SQL queries to perform simple to complex data manipulation tasks.

Course Prerequisites:

This course requires basic understanding of computer and programming.

Detailed Syllabus:

Module-I (10 hrs):

Introductory concepts of DBMS:Introduction and applications of DBMS, Purpose of data base, Data, Independence,Database System architecture- levels, Mappings, Database, users and DBA

Relational Model:Structure of relational databases, Domains, Relations, Relational algebra – fundamentaloperators and syntax, relational algebra queries, tuple relational calculus

Entity-Relationship Model:Basic concepts, Design process, constraints, Keys, Design issues, E-R diagrams, weakentity sets, extended E-R features – generalization, specialization, aggregation,reduction to E-R database schema.

Module-II (14 hrs):

Relational Database design:Functional Dependency – definition, trivial and non-trivial FD, closure of FD set, closureof attributes, irreducible set of FD, Normalization – 1NF, 2NF, 3NF, Decompositionusing FD-dependency preservation, BCNF, Multi- valued dependency, 4NF, Joindependency and 5NF.

SQL Concepts:Basics of SQL, DDL,DML,DCL, structure – creation, alteration, defining constraints – Primary key, foreign key, unique, not null, check, IN operator,Functions – aggregatefunctions, Built-in functions –numeric, date, string functions, set operations, sub-queries,correlated sub-queries, Use of group by, having, order by, join and its types, Exist, Any,All , view and its types; transaction control commands – Commit, Rollback, Savepoint; Introduction to PL/SQL Concepts:Cursors, Stored Procedures, Stored Function, Database Triggers

Module-III (12 hrs):

Query Processing & Query Optimization:Overview, measures of query cost, selection operation, sorting, join, evaluation ofexpressions, transformation of relational expressions, estimating statistics of expressionresults, evaluation plans, materialized views

Transaction Management:Transaction concepts, properties of transactions, serializability of transactions, testingfor serializability, System recovery, Two- Phase Commit protocol, Recovery andAtomicity, Log-based recovery, concurrent executions of transactions and relatedproblems, Locking mechanism, solution to concurrency related problems, deadlock, two-phase locking protocol, Isolation, Intent locking

Security:Introduction, Discretionary access control, Mandatory Access Control, Data Encryption

Text Books:

1. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, "Database SystemsConcepts", McGraw-Hill Education , New Delhi
2. RamezElmasri and Shamkant B. Navathe, "Fundamentals of DatabaseSystems", Pearson Education Inc., New Delhi.

Reference Books:

1. Hector Garcia-Molina, Jeffret D. Ullman, JennifferWidom, "Database Systems: AComplete Book", Pearson Education Inc., New Delhi.
2. C. J. Date "An introduction to Database System", Pearson Education Inc., NewDelhi.
3. Bipin Desai, "An introduction to Database System", Galgotia Publications.
4. Peter Rob & Carlos Coronel, "Database Systems: Design, Implementation, andManagement", CENGAGE Learning India Pvt. Ltd., New Delhi.
5. Mark L. Gillenson, "Fundamentals of Database Management Systems", WileyIndia Pvt. Ltd., New delhi.
6. Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems",McGraw-Hill Education (India), New Delhi.

Operating Systems

Code: PPCCA204 **Credit: 3** **Total Hours: 40**

Course Objectives:

1. To study the main components of an OS and their functions.
2. To study the concept of process, process management and CPU scheduling algorithms.
3. To study the concepts and implementation Memory management policies and virtual memory
4. To study the file system, its implementation and disk management.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand the managerial roles of OS for resource, file system, process, memory and I/O.
2. Understand the process management policies and scheduling of processes by CPU
3. Understand process synchronization and coordination handled by operating system
4. Understand and analyze the memory management and its allocation policies.

5. Conceptualize the components involved in designing a contemporary OS.

Course Prerequisites:

This course requires understanding of computer organization and programming in any language.

Detailed Syllabus

Module 1 (10 Hours)

Operating System Introduction- Functions, Characteristics, Structures - Simple Batch, Multi programmed, timeshared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating-System services, System Calls, Virtual Machines. Process and CPU Scheduling - Process concepts and scheduling, Operation on processes, Cooperating Processes, Threads, and Interposes Communication Scheduling Criteria, Scheduling Algorithm, Multiple -Processor Scheduling, Real-Time Scheduling.

Module 2 (12 Hours)

Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors.

Deadlocks - System Model, Dead locks Characterization, Methods for Handling Deadlocks Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

Module 3 (14 Hours)

Memory Management and Virtual Memory- Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging.Demand Paging, Performance of Demanding Paging, Page Replacement, Page Replacement Algorithm, Allocation of Frames, Thrashing.

File System Interface and Implementation-Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management, Directory Management, Directory Implementation, Efficiency and Performance.

I/O Management – I/O software and its types, Disk Scheduling.

Text Books:

1. Operating System Concepts, by Abraham Silberschatz, Greg Gagne, and Peter Baer Galvin, Ninth Edition, John Wiley & Sons.

Reference Books

2. Operating Systems: Internals and Design Principles, by William Stallings, 8th edition Pearson Education Limited, 2014
3. Operating systems - A concept based Approach, by D.M Dhamdhere, 3rd Edition, Tata McGraw- Hill, 2012.
4. Operating systems, by Harvey M Deital, 3rd Edition, Pearson Education, 2011.

Data Structures Lab

Code: PLCCA201 **Credit: 1** **Hours: 2/week**

Note: This course shares the objectives and outcomes of its associated theory course PPCCA201. Suitable programming language preferably object-oriented will be used to carry out laboratory exercises. The exercises suggested below are illustrative in nature. Additional exercises may be

suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Implement Stack and use it for evaluation of post-fix expression.
2. Implement conversion of prefix expression into post-fix form using recursion.
3. Implement circular queue (using array) with menu options like insert, delete, display and exit.
4. Implement a priority queue (using pointers) and use it to organize student records prioritized by marks.
5. Implement doubly linked circular list to hold strings and use it for organizing a sequence of cities constituting at our program.
6. Implement of a binary search tree with menu options: Construct a tree, insert a node, delete a node, traverse and display preorder, in order and post order sequence of its nodes.
7. Implement of di-graphs using adjacency matrix and find the transitive closure using Warshall's algorithm.
8. Implement a weighted graph and find minimal cost spanning tree using PRIM's Algorithm.
9. Generate 70 random integers in a given range and sort them using quick sort. Apply both binary search and Interpolation search to locate a given integer and compare the search algorithms based on the number of comparisons / probes required or a successful as well as unsuccessful search..
10. Implement Heap Sort, Merge Sort and other sorting algorithms on the above random numbers.
11. Implement a small Real World Application illustrating DS usage.

Programming with Java Lab

Code: PLCCA202 **Credit: 1** **Hours: 2 hrs/week**

Note: This course shares the objectives and outcomes of its associated theory course PPCCA202. Suitable IDE will be used to carry out laboratory exercises. The programs will follow proper object-oriented modeling. The exercises suggested below are illustrative in nature. Additionally, suitable exercises may be suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Develop an Object Oriented Program to find the area and perimeter of a circle.
2. Develop an interest calculator program to find simple interest payable monthly, compound interest payable annually compounded quarterly. Use keyboard inputs for interest rate and principal amount.
3. Define a class to calculate professional tax on a salary amount based on the following tax rate. Use if and switch control structures.

Salary Slab	Tax Rate
Up to Rs. 10000.00	Nil
Between Rs. 10001.00 – Rs. 25000.00	Rs. 100.00
Between Rs. 25001.00 – Rs. 50000.00	Rs. 200.00
Between Rs. 50001.00 – Rs. 75000.00	Rs. 300.00
Between Rs. 75001.00 – Rs. 100000.00	Rs. 450.00
Above Rs. 100000.00	Rs.650.00

4. Develop a program to find the sum of even numbers and sum of odd numbers in a set of numbers. Define a class with suitable methods to carry out the operations. Use array to store numbers.
5. Modify the class defined in sl-6 to find largest and smallest numbers in a set of numbers.
6. A student scores marks in subjects in a semester. A semester has 5 or 6 subjects depending of MCA course. Define a class called Score that contains subject code, name and marks in that subject. Define a class called Student having an array of objects of Score class in it following object composition. Process result of students in different semesters.
7. Define a class called SimpleMath with overloaded methods to carryout arithmetic operations using it. Use static methods appropriately.
8. Redefine the Circle class in exercise 1 to use value of Pi as a constant and a variable to count number of instances created as you go on creating objects.
9. Develop a class to perform the following tasks on a line of text
   a. Count the number of words in the text
   b. Searches a particular string in the text
   c. Checks if the text is a palindrome
10. A library is a collection of books. Generally, a book is authored by one or more authors. Develop a program to add books and display a list of books when searched by author name. Consider title, ISBN number, publisher name, publication year for book; designation, organization, and country for author.
11. In CET, two types of people are there: students and employees. As per Govt. of India, everyone must have his or her AADHAR number for unique identification. Model these objects appropriately using inheritance and create an array of people with several students and employees in it. Write a program to search a student or an employee based on AADHAR number and print its details.
12. A bank account maintains a minimum balance. If the account balance comes down below this level due to some withdrawal then it raises warning and disallows the operation. Define a custom exception class called “InsufficientFundException” which will be raised when such event occurs. Also use the built-in exception class “IllegalArgumentException” which is to be raised when you try to either withdraw or deposit an amount less than or equal to zero.
13. Write a multithreaded program to perform following parallel operations on a set of numbers.
    a) Find the largest number b) Find the sum of the number c) Sort the numbers
14. Write program using console I/O.
15. Write Programs using File I/O.
16. Write program using serialization.

17. Write client server programs using Java sockets.
18. Write JDBC programs to perform CRUD operations.
19. Write JDBC program to execute stored procedures.
20. Write a page counter Servlet to display how many times the webpage is accessed.
21. 21.Write a Servlet program to display the date and time the webpage was last accessed by any user.
22. 22.Write a Servlet program that displays the details of a student like Name, Roll No, Sex, Semester, Hobbies, Programming Languages known when user submits a form using suitable controls such as text field, combo box, list box, radio button, checkbox etc.
23. Write a Servlet program that validates user login. When a user submits user id and password using a login form, a Servlet retrieves the same and validates against the stored values.Store the values of user id and password as init parameters in web.xml file.
24. 24.Write a program to develop a contact database with name, mobile number and email of a person. Use a HTML form to add a contact and a Servlet will retrieve the form data and store in the database table using JDBC
25. 25.Write a program to develop a Spring framework for displaying a simple message

Database Management System Lab

Code: PLCCA203 **Credit: 1** **Hours: 2/week**

Note:This course shares the objectives and outcomes of its associated theory course PPCCA204. Suitable RDBMS such as ORACLE or MySQL will be used to carry out laboratory exercises. The exercises suggested below are illustrative in nature. Additional exercises may be suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Create ER Models for the following: Inventory Control System, Hospital Management System, Hotel Management System, Timetable Management System, Railway Reservation System
2. Derive Database Schema from the ER Models of the above systems
3. Create a database and set constraints, relationships.
4. Perform Insertion, Deletion, Modifying, Altering, Updating and Viewing recordsbased on conditions.
5. Write SQL queries using aggregate functions.
6. Write SQL queries involving sub-queries
7. Write SQL queries using group by, order by, having etc.
8. Write SQL queries involving joins.
9. Create Views, Synonyms, Sequence, Indexes, savepoint.
10. Write a PL/SQL block to satisfy some conditions by accepting input from theuser.
11. Write a PL/SQL block that handles all types of exceptions.

12. Create Stored Procedures.
13. Create database triggers and functions

Python Lab

Code: PLCCA204 **Credit: 2** **Hours: 4/week**

Note: This course shares the objectives and outcomes of its certain lab work. Suitable IDE will be used to carry out laboratory exercises. The programs will follow proper object-oriented modeling. The exercises suggested below are illustrative in nature. Additionally, suitable exercises may be suggested by the faculty concerned to meet the course objectives. Some of the bulleted points to be covered as given below:

Strings and text files: String manipulations, subscript operator, indexing, slicing a string; strings and number system: converting strings to numbers and vice versa; text files: reading/writing text and numbers from/to a file; creating and reading a formatted file (csv or tab-separated); manipulating files and directories, OS and SYS modules.
Lists, Sets, Tuples, and Dictionaries: basic list operations, replacing, inserting, removing an element; searching and sorting lists; dictionary, adding and removing keys, accessing and replacing values; traversing dictionaries.
Regular Expressions: re module, pattern-string syntax, find, match, search, split, sub functions,
Functions: defining and calling functions, arguments and return values, Recursive functions.
Classes and OOP: defining classes, attributes and methods, constructor, static and class methods, inheritance, polymorphism, operator overloading, exception handling, try, except, raise, assert, finally.
GUI Programming: Event-driven programming, tkinter module, GUI Basics, creating simple GUI, buttons, labels, entry fields, dialogs, widget attributes - sizes, fonts, colors layouts, nested frames.
Database and Persistence Programming: Persistence options in Python, Using DBM Files, Using Object Pickling, Using Shelves, Data Persistence in RDBMS, creating connection, executing queries, processing query results.
Numerical Python: NumPy basics, creating ndarray, data types for ndarray, indexing and slicing, basic operations and manipulations on n-dimensional array.
Data Analysis: introduction to Pandas data structures, Series, DataFrames, indexing, selection, filtering, sorting, ranking, handling missing values, data aggregation, plotting with Matplotlib

List of Exercises:

1. Write program using input and output functions.
2. Write programs using control statements.
3. Write programs involving string manipulations.
4. Write program to read from and write to text file.
5. Write program to create and read from CSV file.
6. Write program involving regular expression.
7. Write programs to perform operations on lists.
8. Write programs to perform operations on dictionary.
9. Write program involving function.

10. Write program involving classes and objects.
11. Write program involving inheritance.
12. Write program involving operator overloading
13. Write program involving exception handling.
14. Write program to perform CRUD operations on SQL database.
15. Write GUI programs using basic UI objects.
16. Write programs to perform operations on n-dimensional array.
17. Write programs to perform operations on series.
18. Write programs to perform operations on data frames.
19. Write programs involving data aggregation.
20. Write programs using plotting functions.

Text Books:

1. The Fundamentals of Python: First Programs, Kenneth A. Lambert, 2nd Edition, Cengage Learning, 2011
2. Python Data Science Handbook, First Edition, Jake VanderPlas, O'Reilly Media, 2017

Reference Books:

1. Programming in Python 3, Second Edition, Mark Summerfield, Addition-Wesley, 2010
2. Learning Python, by Mark Lutz, 5th Edition, O'Reilly Media, 2013
3. Python Programming for the Absolute Beginner, Third Edition, Michael Dawson, Cengage Learning, 2010
4. Python for Data Analysis by Wes McKinney, O'Reilly Media, 2013
5. Python in a Nutshell, by Alex Martelli, Anna Ravenscroft & Steve Holden, 3rd Edition, O'Reilly Media, 2017

Semester-3

Core 5: Artificial Intelligence

Code: PPCCA301 Credit: 3 Hours: 3 hrs/week

Course Objectives:

The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence. More specifically:

1. To introduce AI and its basic principles towards problem solving using various search techniques.
2. To provide understanding on knowledge representation and reasoning techniques used in intelligent systems.
3. To introduce machine learning, natural language processing and perception.

Course Outcomes:
On successful completion of the course, the students will be able to:

1. Understand basic principles and techniques of AI towards problem solving
2. Demonstrate their proficiency in knowledge representation in Intelligent systems
3. Demonstrate understanding of basic learning, communication and perception in intelligent system.

Course Prerequisites:
This course requires basic knowledge of computer algorithms and data structures.

Detailed Syllabus

Module-I
Artificial Intelligence: Introduction, Intelligent Agents: Agents & Environments, Concept of Rationality, Nature & Structure of Agents; Problem Solving: Solving Problems by Searching, Classical Search, Adversarial Search, Constraint Satisfaction Problems. Knowledge, Reasoning and Planning: Logical agents, First order logic, Inference in First order logic.

Module-II
Classical planning, Knowledge Representation; Uncertain Knowledge and Reasoning: Probabilistic Reasoning, Learning from Examples, Knowledge in Learning; Natural Language Processing: Language models, Text Classification, information retrieval, information extraction

Module-III
Natural Language for Communication: Phrase structure Grammars, Syntactic Analysis, Augmented grammars and semantic interpretation, Machine translation, Speech recognition; Perception; Expert Systems: Introduction, Design of Expert systems.

Text books:

1. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Third Edition, 2010, Pearson Education, New Delhi.
   Chapters: 1, 2, 3, 4 (4.1, 4.2), 5 (5.1, 5.2, 5.3), 6, 7, 8, 9, 10 (10.1, 10.2, 10.3, 10.5), 12, 14 (14.1-14.6), 18 (18.1- 18.7), 19 (19.1, 19.2, 19.3), 22, 23, 24 (24.1-24.3, 24.5).

Reference books;

1. Elaine A. Rich and Kevin Knight, "Artificial Intelligence", 3rd Edition, 2009, McGraw-Hill Education (India), New Delhi.
2. Nills J. Nilsson, "Artificial Intelligence: A New Synthesis", 2nd Edition, 2000, Elsevier India Publications, New Delhi.
3. Michael Negnevitsky, "Artificial Intelligence: A Guide to Intelligent Systems", Second Edition, 2005, Pearson Education, Inc. New Delhi.
4. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", 1st Edition, 1996, PHI Learning Pvt. Ltd., New Delhi.

5. Ben Coppin, “Artificial Intelligence Illuminated”, 2005, Narosa Publication, New Delhi. ISBN: 978-81-7319-671-3
6. Joseph Giarratano and Gary Riley, “Expert Systems: Principles and Programming”, Fourth Edition, CENGAGE Learning India Pvt. Ltd., New Delhi. Chapters: 1 and 6.

Core 6: Computer Networks

Code: PPCCA301 Credit: 3 Hours: 3 hrs/week

Course Objectives:

1. To provide students with broad concepts and fundamentals of computer networks.
2. To familiarize students with the layered approach to computer network.
3. To provide adequate knowledge on issues and protocols involved in different layers of network.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand the basic concepts of computer network and data communication.
2. Understand the functions of each layer in the OSI and TCP/IP reference model.
3. Understand the working of essential protocols of computer networks, and how they can be applied in network design and implementation.

Course Prerequisites:

This course requires understanding of computer programming and Data structures.

Detail Syllabus

Module I

Network architecture, Layers, Transmission Media, Data Link Layer: Issues in the data link layer, Framing, Error detection and correction, Link-level Flow Control, Medium access, CSMA, Ethernet, Token ring, FDDI, Wired LAN, Wireless LAN

Module II

Connecting Devices, Bridges and Switches, Circuit switching vs. packet switching, Packet switched networks, Network Layer: Design Issues, Logical Addressing, Subnetting, CIDR, IPv4, IPv6, Address Mapping, ARP, RARP, DHCP, ICMP; Delivery, Forwarding, Routing algorithms, RIP, OSPF, BGP –Multicasting – Congestion avoidance in network layer

Module III

Transport Layer: Process-to-process delivery, UDP, TCP, Adaptive Flow Control, Adaptive Retransmission, Congestion control, Congestion avoidance and QoS
Application Layer: Email (SMTP, MIME, IMAP, POP3), Remote Logging (Telnet), File Transfer (FTP), WWW and HTTP, Domain Name System (DNS), Network management (SNMP)

Text Books:

1. Data Communications and Networking by Behrouz A. Forouzan. Third Edition, TMH.

Reference Books:

1. Computer Networks by Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI
2. Computer Networks: A Systems Approach by Larry L. Peterson, Bruce S. Davie, Morgan Kauffmann Inc., 2003.
3. Computer and Communication Networks by Nader F. Mir, Pearson Education, 2007
4. Data and Computer Communication, William Stallings, Sixth Edition, Pearson Education, 2000

Core 7 : Software Engineering

Code: PPCCA303 Credit: 3 Hours: 3 hrs/week

Course Objectives:

1. To provide understanding of software process, requirement engineering and design engineering.
2. To provide knowledge on UML to develop various models for the software-to-be.
3. To provide understanding of various software testing techniques and strategies.
4. To provide understanding of quality metrics for software product.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand software requirements and develop SRS.
2. Analyze requirements and develop various UML models for a software-to-be at hand.
3. Display ability to test software using appropriate techniques.
4. Access quality of software product.

Course Prerequisites:

This course requires basic understanding of computer and programming.

Detailed Syllabus:

Module-I

Introduction: The evolving role of software, changing nature of software, software myths, Software engineering- a layered technology, a process framework.

Process models: The waterfall model, incremental process models, evolutionary process models, specialized process models, the unified process, agile process models.

Requirements Engineering: Requirement engineering tasks, functional and non-functional requirements, the software requirements specification (SRS), IEEE 830 guidelines, requirements validation, requirements management.

Module-II

Building Analysis Models: Analysis Modeling Approaches: Structure Analysis, Object-Oriented Analysis; Data Modeling: E-R Diagram; Scenario-Based Modeling: Use-Case Model, Activity Diagram; Flow-Oriented Modeling: Data Flow Model, DFD, Class-Based Modeling: Identifying Analysis Classes, Class Model, CRC Model, Association and Dependency, Object Diagram; Behavior Modeling: State Diagram, Sequence Diagram, and Collaboration Diagram.

Building Design Models: Design process and design quality, Design Concepts, Elements of Design Model, Software Architecture: importance of software architecture, architectural styles and patterns, architectural design; Component-level Design: What is a component, Basic Design

Principles, Design guidelines, Cohesion, Coupling, Component Diagram, Deployment diagram; User Interface Design: The Golden Rules, interface design evaluation cycle.

Module-III

Software Testing: Verification and Validation, Unit testing and Integration Testing, Validation testing, System testing, the Art of Debugging, Black-box and White-box testing, Basis path testing, Control structure testing, Graph-Based testing, Equivalence partitioning, Boundary value analysis, Object-Oriented testing, Scenario-Based Testing, Radom testing, Partition testing, Inter-class testing, GUI testing, Client Server testing.

Product Metrics: Software quality factors, Attributes of effective software metrics, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

Text Books:

1. Roger S. Pressman, Software Engineering, A practitioner's Approach, 6th Edition, McGraw Hill International Edition.
2. Rajib Mall, Fundamentals of Software Engineering, PHI, 2014.

Reference Books:

1. I. Sommerville, Software Engineering, 9th Edition, Pearson Education.
2. Waman S. Jawadekar Software Engineering: Principles and Practice, Tata McGraw-Hill Education, 2004.
3. Michael R. Blaha, and James R Rumbaugh, Object-Oriented Modeling and Design with UML, 2nd Edition 2005, Pearson Education.
4. Meilir Page-Jones, Larry L. Constantine, Fundamentals of Object-oriented Design in UML, Pearson Education.

Core 8 : Design and Analysis of Algorithms

Code: PPCCA 304 Credit: 3 Hours: 3 hrs/week

Course Objectives:

1. To understand basics of algorithmic for problem solving.
2. To provide knowledge on algorithm design techniques.
3. To provide knowledge on complexity analysis.
4. To introduce some important algorithms.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Design new and efficient algorithms for specific problems.
2. Learns various algorithm design techniques.
3. Learn complexity analysis.

Course Prerequisites:

1. This course does not require any prerequisite.

Module I

Introduction, Growth of Functions (Asymptotic notations, standard notations and common functions), Recurrences, solution of recurrences by substitution, recursion tree and Master methods, Divide and conquer algorithms: Merge sort, Quick sort, Strassen's Matrix Multiplication, Heap sort: Heaps, Building a heap, The heap sort algorithm.

Module II

Dynamic programming algorithms (Matrix-chain multiplication, All- pairs shortest paths (Floyd – Warshall), Single source shortest paths (Bellman-Ford Algorithm), Longest common subsequence, Assembly-line scheduling, Greedy Algorithms - (Achivity- selection Problem, Fractional knapsack problem, Huffman codes, Prim's algorithm- Kruskal's Algorithm- Dijkstra's Algorithm).

Module III

BRUTE FORCE– Closest-Pair and Convex-Hull Problems-Exhaustive Search – Traveling Salesman Problem, Backtracking – n-Queens problem, Graph coloring, Hamilton Cycle; branch and bound: 15-puzzle problem.
String matching (Rabin-Karp algorithm)

Module IV

Linear Programming: The Simplex Method; The Maximum-Flow Problem: The Ford-Fulkerson Method; NP Completeness; Approximate algorithms: Traveling Salesman Problem, vertex cover, set cover.

Text Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Third Edition, PHI Learning Private Limited, 2012.
2. ELLIS HOROWITZ and SARTAJ SAHNI. Fundamentals of Computer Algorithms. 2nd Edition, Universities Press

References:

1. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, Reprint 2006.
2. Donald E. Knuth, "The Art of Computer Programming", Volumes 1& 3 Pearson Education, 2009. Steven S. Skiena, "The Algorithm Design Manual", Second Edition, Springer, 2008
3. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education, 2012.
4. http://nptel.ac.in/

(Elective 1)

Theory of Computation

Code: PPECA301 Credit: 3 Hours: 3 hrs/week

Course Objectives:

1. To understand how theoretical machine is designed.
2. To provide knowledge on theoretical machine design techniques.
3. To provide knowledge on complexity class of problems.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Acquire knowledge on theoretical machine design.
2. Realize the difference between determinism and non-determinism.
3. Get idea on recursive properties of languages.

Course Prerequisites:

1. This course requires data structures as prerequisite.

Module I

Alphabet, languages and finite automata (deterministic and nondeterministic), Minimization of finite automata. DFA/NFA to regular expression and vice versa using Arden's Formula.
grammars: Production rules and derivation of languages. Chomsky's hierarchy of languages and Grammars. Regular grammars, regular expressions. Closure and decision properties of regular languages. Pumping lemma of regular sets.

Module II

Context free grammars and pushdown automata. Chomsky and Griebach normal forms. Parse trees, Cook- Younger- Kasami-parsing algorithms. Ambiguity and properties of context free languages. Pumping lemma. Deterministic pushdown automata, closure properties of deterministic context free languages.

Module III

Turing machines and variation of Turing machine model, Turing computability, Type of languages. Linear bounded automata and context sensitive languages. Primitive recursive functions.
Gödel numbering. Ackermann's function, recursiveness of Ackermann and Turing computable functions. Church Turing hypothesis. Recursive and recursively enumerable languages.

Module IV

Universal Turing machine and undecidable problems. Undecidability of Post correspondence problem. Valid and invalid computations of Turing machines and some undecidable properties of context free language problems. Time complexity class P, class NP, NP completeness.

Text Books:

1. Introduction to the theory of computation: Michael Sipser, Cengage Learning
2. Introduction to Automata Theory, Languages and Computation: J.E. Hopcroft and J.D Ullman, Pearson Education, 3rd Edition.

Reference Books:

1. Automata Theory: Nasir and Srimani, Cambridge University Press.
2. Introduction to Computer Theory: Daniel I.A. Cohen, Willey India, 2nd Edition.

Computer Security

Code: PPECA302 Credit: 3 Hours: 3 hrs/week

Objective

1. To provide an understanding of requirement, different approaches and major issues in information security.
2. To develop a basic understanding of cryptography and its use in maintaining security.
3. To provide knowledge of computer security technologies used in computer operating systems, distributed systems, networks and representative applications.
4. To provide an understanding of different issues of overall social, economic and professional contexts, and getting aware of the ethical and legal responsibilities related to computer security.

Course Outcomes:

1. To understand about different external and internal threats to an organization.
2. To understand different procedures used to discover, analyze and how to deal with threats to an organization.
3. To understand and apply the cryptography techniques to transmit data securely.
4. To understand network security threats and solutions to handle those.
5. To understand security related legal and regulatory issues.

Course Prerequisite:

This course requires understanding of computer organization, operating system, network and discrete mathematics.

Module-I

Introduction to Cryptography: Security goals, cryptography attacks, services & mechanisms, techniques.

Symmetric-Key Encipherment: Mathematics of cryptography, Traditional Symmetric-key Ciphers, Data Encryption Standard- Introduction, structure, analysis & security, Advanced Encryption Standard –Introduction, Transformations, key expansion, AES ciphers

Module-II

Asymmetric-Key Encipherment: Mathematics of Asymmetric-Key Cryptography: primes, primality testing, factorization, Chinese remainder theorem, quadratic congruence, exponentiation and logarithm; Asymmetric-Key Cryptography: RSA cryptosystem, Rabin cryptosystem

Integrity, Authentication and Key Management: Message Integrity, Message Authentication, Cryptography Hash Functions – MD4 Hash, SHA-512, Digital Signature, Entity Authentication, Key Management: Symmetric key distribution- KDC, Kerberos, Symmetric key agreement – Diffie-Hellman, Public Key Distribution – X.509

Module-III

Network Security: Authentication Service – Electronic Mail Security – PGP – S/MIME - IP Security – Web Security.

System Security: Intrusion detection – password management – Viruses and related Threats – Virus Counter measures – Firewall Design Principles – Trusted Systems.

Text Books:

1. Behrouz A. Forouzan & Debdeep Mukhopadhayay, "Cryptography & Network Security", McGraw Hill, New Delhi

Reference Books:

1. William **Stallings** & Lawrie **Brown**, "Computer Security: Principles and Practice", Pearson Education, Inc. New Delhi.
2. Charlie **Kaufman**, Radia **Perlman** & Mike **Speciner**, "Network Security: Private Communication in a Public World", 2nd Edition, 2003, PHI Learning. New Delhi.

Machine Learning

Code: PPECA303 Credit: 3 Hours: 3 hrs/week

Course Objectives:

Machine learning uses interdisciplinary techniques such as statistics, linear algebra, optimization, and computer science to create automated systems that can sift through large volumes of data at high speed to make predictions or decisions without human intervention. Machine learning as a field is now incredibly pervasive, with applications spanning from business intelligence to homeland security, from analyzing biochemical interactions to structural monitoring of aging bridges, and from emissions to astrophysics, etc. This class will familiarize students with a broad cross-section of models and algorithms for machine learning, and prepare students for research or industry application of machine learning techniques.

Course Outcomes: By the end of the course, students should be able to:

1. Develop an appreciation for what is involved in learning models from data.
2. Understand a wide variety of learning algorithms.
3. Understand how to evaluate models generated from data.
4. Apply the algorithms to a real-world problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.

Course Prerequisites:
The student must have studied courses on Statistics and Computer Algorithms

Detailed Syllabus:

Module-I
Introduction to Machine Learning: Types of Learning, Unsupervised, Supervised, Reinforcement Learning
Linear Models for Regression: Linear Basis Function Models, Least Squares, Bayesian Linear Regression, Evidence Approximation.
Linear Models for Classification: Discriminant Functions, Fisher's Discriminant, Probabilistic Discriminant Models, Fixed Basis Functions, Logistic Regression, Multiclass Logistic Regression

Module-II
Kernel Methods: Radial Basis Function, Gaussian Processes for Classification, SVM for Regression and Classification,
Mixture Models and EM: K-means Clustering, Mixtures of Gaussian, EM Algorithm.
Dimensionality Reduction: Principal Component Analysis, PCA for High Dimensional Data

Module-III
Neural Network: Introduction, Feed forward NN, Training Neural Network, Gradient Descent, Error Back Propagation, Regularization in Neural Network, Bayesian Neural Network.
Deep Learning: Introduction, Deep Feed Forward Network, Regularization for Deep Learning, Convolutional Network.

Text Books:
1. Christopher Bishop, Pattern Recognition and Machine Learning, Springer
2. Ian Good Fellow, Yoshua Dengio and Aaron Courville, Deep Learning, MIT Press

Reference Books:
1. Shai Shalev-Shwartz and Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press.

Computer Graphics

Code: PPECA304 Credit: 3 Hours: 3 hrs/week

Course Objectives:

1. To introduce the use of the components of a graphics system and become familiar with building approach of graphics system components and algorithms related with them.
2. To learn the basic principles of 3- dimensional computer graphics.
3. Provide an understanding of how to scan convert the basic geometrical primitives, how to transform the shapes to fit them as per the picture definition.
4. Provide an understanding of mapping from a world coordinates to device coordinates, clipping, and projections.
5. To be able to discuss the application of computer graphics concepts in the development of computer games, information visualization, and business applications

Course Outcomes:

On successful completion of the course, the student will be able:

1. To understand the basic concepts using computer graphics
2. To describe the importance of viewing and projections
3. To implement various algorithms to scan, convert the basic geometrical primitives, transformations, Area filling, clipping

Module I

An Introduction Graphics System: Computer Graphics and Its Types, Application of computer graphics, Graphics Systems: Video Display Devices, Raster Scan Systems, Random Scan Systems, Graphics Monitors and Work Stations, Input Devices, Hard Copy Devices, Graphics Software.

Module II

Output Primitives and Attributes of Output Primitives: Output Primitive Points and Lines, Line Drawing Algorithms, Circle Generating Algorithms, Scan-Line Polygon Fill Algorithm, Inside-Outside tests, Boundary-Fill Algorithm, Flood Fill Algorithm, Cell Array, Character Generation, Attributes of Output Primitives: Line Attributes, Color and Grayscale Levels, Area fill Attributes, Character Attributes, Bundled Attributes, Antialiasing. Two-dimensional Geometric Transformations: Basic Transformations, Matrix Representation and Homogeneous Coordinates, Composite Transformations, Reflection and Shearing.

Module III

Two-Dimension Viewing: The viewing Pipeline, Window to view port coordinate transformation, Clipping Operations, Point Clipping, Line Clipping, Polygon Clipping, Text Clipping, Exterior Clipping Three-Dimensional Concepts: Three Dimensional Display Methods, 3D Transformations, Parallel Projection and Perspective Projection. Three Dimensional Object Representations: Curved Surfaces, Quadratic Surfaces, Spline Representations, Bezier Spline Curves and Surfaces, B-Spline Curves and Surfaces, Octrees, BSP Trees, Fractal Geometry Methods, Shape Grammars. Visible Surface Detection Methods: Classification of Visible-Surface Detection Algorithms, Back-Face Detection, Depth-Buffer method, A-Buffer Method, Scan line and Depth Sorting, Area subdivision Method, Ray Casting Method.

Text Books:

1. Donald **Hearn** & M. Pauline **Baker**, "*Computer Graphics with OpenGL*", Third Edition, 2004, Pearson Education, Inc. New Delhi.

2. Zhigang **Xiang**, Roy A. **Plastock**, "*Computer Graphics*", Second Edition, 2007, McGraw-Hill Education (India), New Delhi.

Reference Books:

1. Plastock: Theory & Problem of Computer Gaphics, Schaum Series.
2. Foley & Van Dam: Fundamentals of Interactive Computer Graphics, Addison-Wesley.
3. Newman: Principles of Interactive Computer Graphics, McGraw Hill.

Enterprise Java Technologies Laboratory

Code: PLCCA302 Credit: 2 Hours: 4 hrs/week

Note: This course shares the objectives and outcomes of its associated theory course PPCCA402. Suitable IDE will be used to carry out laboratory exercises. The programs will follow proper object-oriented modeling. The exercises suggested below are illustrative in nature. Additional suitable exercises may be suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Write a page counter Servlet to display how many times the webpage is accessed.
2. Write a Servlet program to display the date and time the webpage was last accessed by any user.
3. Write a Servlet program that displays the details of a student like Name, Roll No, Sex, Semester, Hobbies, Programming Languages known when user submits a form using suitable controls such as text field, combo box, list box, radio button, checkbox etc.
4. Write a Servlet program that validates user login. When a user submits user id and password using a login form, a Servlet retrieves the same and validates against the stored values. Store the values of user id and password as init parameters in web.xml file.
5. Write a program to develop a contact database with name, mobile number and email of a person. Use a HTML form to add a contact and a Servlet will retrieve the form data and store in the database table using JDBC.
6. Write a JSP program that will search the contact database (developed in Sl. No.5) using the person's mobile number and display it.
7. Write a JSP program that will accept student details such as name, roll no, gender, and semester using HTML form. When the form is submitted it will display the details of the student just submitted. Use a Java Bean to hold the data.
8. Develop a web application following MVC model (combine Servlet, JSP and Java Bean) to develop an employee database. Perform operations such as insert employee records, search employee by Employee ID.
9. Using Java Server Faces (JSF) technology, develop an application to accept student details and display it. Use appropriate components such as text field, radio button, check box, combo box, list box to accept student name, gender, semester, languages known, and hobbies.
10. Develop a JSF application for login. Hardcode the userid and password values in the managed bean. If the login is success, it displays a success page and if the login fails it

will display the login form with the error message. Define page navigation in the faces-config.xml for page navigation.

11. Develop a JSF application for simple arithmetic operations using two numbers. The webpage will have two input text fields to accept the numbers, one output text field to display the result and four command buttons to perform the operations. Use number converter as required by the application.
12. Develop a JSF application to check if a student scores a pass or fail in a subject. The webpage will have one input text field to accept the score, one output text field to display the result and one command button to submit data. Use range validator as required by the application to validate a student's score between 0 and 100.
13. Develop JSF application using AJAX
14. Develop application using Sateless Session Bean
15. Develop application using Sateful Session Bean
16. Develop application using Message Driven Bean
17. Develop application involving Entity Bean and JPA
18. Develop application to perform CRUD operation using JPA
19. Develop application using Hibernate
20. Develop application using web service