INT. M.SC. SYLLABUS

Effective from Academic Year 2022-2023

Department of Computer Science

School of Mathematics, Statistics and Computational Sciences

Central University of Rajasthan

NH-8 Jaipur- Ajmer Highway, Bandarsindri
Kishangarh -305817
District-Ajmer, Rajasthan
Website: www.curaj.ac.in

Scheme of Integrated M.Sc. Computer Science

The details of the courses with code, title and the credits assign are as given below.

Course Category

CC: Compulsory Course, EC: Elective Course,

Course Code: First 3 Characters (Departmental Code), First digit (Course level), Next 2 digits (Serial of the course).

Semester – I

S. No.	Course Code	Course t Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-101	Computer Fundamentals & Programming in C	CC	2	1	0	3
2	CSC-102	C Programming Lab	CC	0	0	2	1

Semester-II

S. No.	Course Code	Course Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-103	Object Oriented Programming in C++	CC	2	1	0	3
2	CSC-104	C++ Programming Lab	CC	0	0	2	1
3	CSC-105	ICT	EC	1	0	2	2

Semester – III

S. No.	Course Code	Course Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-201	Data Structures	CC	2	1	0	3
2	CSC-202	Data Structure Lab	CC	0	0	2	1

Semester – IV

S. No.	Course Code	Course Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-203	Database Management System	CC	2	1	0	3
2	CSC-204	Database Management Lab	CC	0	0	2	1

Semester-V

S. No.	Course Code	Course Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-301	Computer Networks	CC	3	0	2	4
2	CSC-302	Theory of Computation	CC	3	1	0	4
3	CSC-303	Software Engineering	CC	3	1	0	4
4		Elective-I (Science)	EC	3	0	0	3

5		Elective-II	EC	3	0	0	3
Total							18

Semester-VI

S. No.	Course Code	Course Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-304	Design & Analysis of Algorithms	CC	3	0	2	4
2	CSC-305	Operating System	CC	3	0	0	3
3	CSC-306	Computer Systems Architecture	CC	3	0	0	3
4	CSC-307	Project + Seminar	CC	0	0	4	2
5		Elective III (Science)	EC	3	0	0	3
6		Elective IV	EC	3	0	0	3
Total							18

Semester-VII

S. No.	Course Code	Course Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-401	Introduction to Artificial Intelligence	CC	3	1	0	4
2	CSC-402	Discrete Structures & Graph Theory	CC	3	1	0	4
3	CSC-403	Probability & Statistics	CC	3	1	0	4
4	CSC-404	Advanced Algorithms	CC	3	1	0	4
5	CSC-405	Programming in Python	CC	2	0	4	4
6	CSC-406	Professional Communication	CC	2	0	0	2
7	CSC-407	Artificial Intelligence Lab	CC	0	0	2	1
8	CSC-408	Advanced Algorithms Lab	CC	0	0	2	1
Total							24

Semester – VIII

S. No.	**Course Code**	**Course Title**	**Type of Course (CC/EC)**	**L**	**T**	**P**	**Credits**
1	CSC-409	Machine Learning	CC	3	0	0	3
2	CSC-410	Big Data Analytics	CC	3	0	0	3
3	CSC-411	Natural Language Processing	CC	3	0	2	4
4	CSC-412	Soft Computing	CC	3	0	2	4
5		Elective-V	EC	3	1	0	4
6	CSC-413	Dissertation – I *	CC	0	4	0	4
7	CSC-414	Machine Learning Lab	CC	0	0	2	1
8	CSC-415	BDA Lab	CC	0	0	2	1
Total				24			

* Students will prepare a report on a research topic.

Semester – IX

S. No.	**Course Code**	**Course Title**	**Type of Course (CC/EC)**	**L**	**T**	**P**	**Credits**
1	CSC-501	Data Warehousing & Mining	CC	3	0	0	3

2	CSC-502	Neural Network & Deep Learning	CC	3	0	0	3
3	CSC-503	Image Processing & Computer Vision	CC	3	0	2	4
4		Elective-VI	EC	3	0	0	3
5		Elective-VII	EC	3	0	0	3
6	CSC-504	Dissertation – II	CC	0	0	8	4
7	CSC-505	Data Warehousing & Mining Lab	CC	0	0	2	1
8	CSC-506	Neural Network & Deep Learning Lab	CC	0	0	2	1
9	CSC-507	Summer Training Presentation	CC	0	2	0	2
Total							24

* Summer Training after semester II (Duration 4 – 6 weeks).

Semester – X

S. No.	Course Code	Course Title	Type of Course (CC/EC)	L	T	P	Credits
1	CSC-508	Project Work in Industry or Institution (16 week)	CC	0	20	20	20
2	CSC-509	Self-Study Course	EC	0	0	0	4
Total							24

List of Electives:

Third Year		**Fourth Year**		**Fifth Year**
Subject Code	**Subject Title**	**Subject Code**	**Subject Title**	**Subject Code**
CSC-331	Computer Graphics	CSC-431	Web Technologies	CSC-531
CSC-332	Internet Technologies	CSC-432	Cloud Computing	CSC-532
CSC-333	E-Commerce	CSC-433	Parallel Processing	CSC-533
CSC-334	Open-Source Operating System	CSC-434	Ad-hoc & Wireless Networks	CSC-534
CSC-335	Discrete Structure	CSC-435	High Performance Computing	CSC-535
CSC-336	Information Security	CSC-436	Internet of Things	CSC-536
CSC-337	Distributed Systems	CSC-437	ADBMS	CSC-537
CSC-338	Programming in Java	CSC-438	Software Project Management	CSC-538
CSC-339	Foundation of Vedic Mathematics	CSC-439	Computing & Vedic Mathematics	CSC-539
				CSC-540

Detailed Syllabus

CSC-101: Computer Fundamentals & Programming in C

Course Outline: Basic understanding of computer fundamentals and various types of languages. Various kinds of number representation and flow chart for easy understanding flow of an algorithm. C-language basics, control and looping control structures.

Course Objectives:-

- To learn the basic principles of programming and software development.
- To demonstrate the use of various structured Programming concepts with the help of programs.
- To enhance problem-solving and programming skills in C.

UNIT – I: Introduction to Computer, Von Neumann Architecture, Generation of Computer, Storage Device- Primary Memory and Secondary Storage, Random, Direct, Sequential access methods. Concept of High-Level, Assembly and Low Level programming languages, Program Development Steps, Representing Algorithms through flow chart, pseudo code.
Binary Codes: Binary arithmetic, Addition and subtraction of Integers and floating point numbers. Multiplication of Integers. Gray code, BCD, Excess-3 and Excess-3 gray codes,
Concept of radix and representation of numbers in radix r with special cases of r=2, 8, 10 and 16 with conversion from radix r1 to radix r2. r's and (r-1)'s complement. Representation of Integer in sign-magnitude, signed 1's and 2's complement

UNIT-II : Structure of C program, A Simple C program, identifiers, basic data types and sizes, Constants, variables, arithmetic, relational and logical operators, increment and decrement operators, conditional operator, bit-wise operators, assignment operators, expressions, type conversions, conditional expressions, precedence and order of evaluation. Input-output statements, statements and blocks, if and switch statements, loops- while, do-while and for statements, break, continue, goto and labels, programming examples.

UNIT-III: Designing structured programs, Functions, basics, parameter passing, storage classes- extern, auto, register, static, scope rules, block structure, user defined functions, standard library functions, recursive functions, header files, example c programs.

UNIT – IV: Introduction to Arrays- concepts, declaration, definition, accessing elements, storing elements, arrays and functions, two-dimensional and multi-dimensional arrays, applications of arrays. String and String functions.
Derived types- structures- declaration, definition, Pointers- concepts, Character pointers and functions, pointers to pointers, pointers and multidimensional arrays

Text/Reference Books

1. Ritchie & Kernighan, The C Programming language, ANSI C Version 2nd Ed., PHI.
2. P. Dey, N. Ghosh, Computer Fundamentals and programming in C, 1st Edition, 2006 Oxford University Press
3. Ashok Kamthane, Programming in C, 2nd Ed., Pearson 2011
4. Schildt, C- The Complete Reference, 4th Ed., TMH 2000
5. E. Balaguruswamy, Programming in ANSI C, 6th Ed., TMH 2012
6. V. Rajaraman, Fundamentals of Computers, 5th Ed. PHI, 2011.
7. P.K. Sinha (Fundamental of Computers) 6th Edition BPB Publications 2003

Course Outcomes:

- Basic understanding of programming language and storage representation.
- Demonstration of data types, logical operators, arithmetic operators and increment and decrement operators.
- Capability to write C-Program for Simple C-Program.
- Able to decide the data types and Data Structures for various kinds of data.

CSC-103: Object Oriented Programming in C++

COURSE OUTLINE

This course provides in-depth coverage of object-oriented programming principles and techniques using C++. The course begins with a brief review of control structures and data types with emphasis on structured data types and array processing. It then moves on to introduce the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design. Other topics include overloading, data abstraction, information hiding, encapsulation, inheritance, and polymorphism.

Prerequisites: Experience with C language is a prerequisite.

Objectives:

The course is designed to provide complete knowledge of Object-Oriented Programming through C++ and to enhance the programming skills of the students by giving practical assignments to be done in labs. The following are the main objectives of this course:

- To learn advanced features of the C++ programming language as a continuation of C programming.
- To learn the basic principles of object-oriented design and software engineering regarding software reuse and managing complexity.
- To demonstrate the use of various OOPs concepts with the help of programs.
- To enhance problem-solving and programming skills in C++.

UNIT I: Introduction to programming paradigms- (Process oriented and Object oriented). Concept of object, class, objects as variables of class data type, difference in structures and class in terms of access to members, private and public members of a class, data & function members. Characteristics of OOP- Data hiding, Encapsulation, data security.

UNIT II: Basics of C++: Structure of C++ programs, introduction to defining member functions within and outside a class, keyword *using*, declaring class, creating objects, constructors & destructor functions, Initializing member values with and without use of constructors, simple programs to access & manipulate data members, *cin* and *cout* functions. Dangers of returning reference to a private data member, constant objects and members function, composition of classes, friend functions and classes, using *this* pointer, creating and destroying objects dynamically using *new* and *delete* operators. Static class members, container classes and iterators, proxy classes

UNIT III: Operator overloading: Fundamentals, Restrictions, operator functions as class members v/s as friend functions. Overloading stream function, binary operators and unary operators. Converting between types.

UNIT IV: Inheritance: Base classes and derived classes, protected members, relationship between base class and derived classes, constructors and destructors in derived classes, public, private and protected inheritance, relationship among objects in an inheritance hierarchy, abstract classes, virtual functions and dynamic binding, virtual destructors.
Multiple inheritance, virtual base classes, pointers to classes and class members, multiple class members. Templates, exception handling.

Text/ References

1. E. Balagurusamy, Object Oriented Programming with C++, 5th Edition, TMH Education 2011
2. Robert Lafore, Object Oriented Programming, Pearson Publication 2008
3. Rajesh Kumar Shuka, Wiley Publication, 2008

4. Bjarne Stroustrup, The C++ Programming Language, 3rd Edition, Pearson Publication 2002

Learning Outcomes

After the completion of the course, the students will be able to do the following:

- Understand the features of C++ supporting object-oriented programming.
- Use the characteristics of an object-oriented programming language in a program.
- Use the basic object-oriented design principles in computer problem-solving.
- Use the basic principles of software engineering in managing a complex software project.

CSC105: Information Communication and Technology

Objectives:

1) Basic understanding of Hardware, Software and Networking fundamentals to under graduate students.
2) Demonstrate the Detail Hardware configurations of available IT Infrastructure and current status of Hardware configuration includes processor, Memory and Various I/O devices.
3) Basic understanding of most commonly usage software such as Microsoft Power Point, MSWORD and EXCEL.
4) Hands on practice of DOS and Linux Operating System Commands.
5) Basic understanding and terminology of Computer Networks and usage of Internet.

Learning Outcomes:-

1. Students should be able to input experimental data into Microsoft Excel Work Sheet in the form of rows and columns.
2. Perform calculations in Microsoft Excel using both manually inputting formulas and built-in functions.
3. Generate simple and effective tables and graphs to describe experimental data in Microsoft Excel.
4. Properly format and organize a formal laboratory report in Microsoft Word.
5. Integrate both graphs and tables created in Microsoft Excel into a laboratory report in Microsoft Word and Microsoft Power Point Presentation.
6. Students should be able to use equations, sample calculations in Excel using Formulas option.
7. Critically evaluate experimental results on a basic level in the form of various graphs to understand easily.

UNIT-1 Hardware Details:-

System Unit, Monitor, Keyboard, Mouse, Flash Memory, Cache Memory and functioning and
Internal structure of Hard Disk. Evolution and Types of latest Microprocessor. Printer, Scanner,
Functionality and basic differences in printing technology.

UNIT-2 Basic Software Fundamentals:-

Basics of software, Types of Software – System software, Application software, Utility Software,
Open source software, compiler, assembler, and interpreter.
Operating Systems – Functions, Types, Dos, windows, Linux. Basic DOS Commands, Linux Commands.
Understanding Word Processing: Word Processing Basics, Opening and Closing of

documents,
Text creation and Manipulation, Formatting of text, Table handling, Spell check, language setting
and thesaurus, Printing of word document.
Using Spread Sheet: Basics of Spreadsheet, Manipulation of cells, Formulas and Functions, Editing of Spread Sheet, printing of Spread Sheet. Basics of presentation software, Creating Presentation, Preparation and Presentation of Slides, Slide Show, Taking printouts of presentation/handouts.
UNIT-3 Networking Fundamentals:-
Types of Networks :- What is Computer Network, LAN , WAN and MAN. Networking Devices such as Repeater, Hub, Switch and Router. Introduction about Layers in Computer Networks. Various kinds of addresses in Networking. Demonstration of Wireshark Tools and
some of the applications using Wireshark.
Introduction to Internet, WWW and Web Browsers: Concept of Internet, Applications of Internet,
connecting to internet, ISP, World Wide Web, Web Browsing software, Search Engines, Understanding URL, Domain name.

Reference Books

- P. K. Sinha and Priti Sinha. Computer fundamentals. BPB publications.
- B. Ram, Computer fundamentals: architecture and organization. New Age International.
- D. H. Sanders, Computers today, McGraw-Hill.
- Anita Goel, Computer fundamentals. Pearson Education India.
- Computer Networks, Tanenbaum.
- Red Hat Linux 9, Bible Christstopher Negus.

Programming Assignment:-

1) Perform various DOS Commands.
2) Practise Various Linux Commands including Networking and Backup.
3) Create a Word Document and practice the options like word count, right and left justification, spell check, font and character size modification, etc.....
4) Preparation of ppt with animation and various kind of slide design options.
5) Learn Excel Basics
    - Creating a new workbook
    - Entering data into a spreadsheet
    - Resizing columns to show all contents
    - Saving a spreadsheet
    - Printing a spreadsheet
6) Basic Formatting and Spreadsheet Manipulation
    - Add rows and columns to an existing spreadsheet
    - Reformat data (center, comma and currency styles, bold, text color)
    - Work with a simple formula (product) and function (sum)
7) Hands on practise of Sort & Filter, Find & Select Commands in Excel.
8) Exporting various types of graphs (Column, Pie, Bar, Area, Scatter etc...) from Excel to Power Point.
9) Install Wireshark Tool and Connect internet to the desktop and demonstrate various layer protocols and addresses using in it.
10) Save the traffic files and analyse the various network parameter such as throughput, End-to-End Delay and packet loss.

Note:- Course Instructor can prepare own practical assignments to cover the Syllabus Unit-I,II & III.

CSC-201: Data Structures

COURSE OUTLINE
The course focuses on basic and essential topics in data structures, including array-based lists, linked lists, recursion, stack, queue, heaps, sorting algorithms, and binary tree.

Prerequisites: Experience with C language is a prerequisite.

Objectives:
- To impart the basic concepts of data structures and algorithms.
- To introduce various techniques for the representation of the data in the real world.
- To understand basic concepts about the array, stacks, queues, lists, and trees.
- To understand concepts about searching and sorting techniques.
- To understanding writing algorithms and step by step approach in solving problems with the help of fundamental data structures.

UNIT I: Arrays: Array as storage element, Row major & column major form of arrays, computation of address of elements of n dimensional array. Arrays as storage elements for representing polynomial of one or more degrees for addition & multiplication, sparse matrices for transposing & multiplication
UNIT II : Stack, queue, dequeue, circular queue for insertion and deletion with condition for over and underflow, transposition of sparse matrices with algorithms of varying complexity
UNIT III: Linear linked lists: singly, doubly and circularly connected linear linked lists insertion, deletion at/ from beginning and any point in ordered or unordered lists. Comparison of arrays and linked lists as data structures.
Linked implementation of stack, queue and dequeue. Algorithms for/of insertion, deletion of stack, queue, dequeue implemented using linked structures. Polynomial representation using linked lists for addition, Concepts of Head Node in linked lists.
Searching: Sequential and binary search.
UNIT IV: Non-Linear Structures: Trees definition, characteristics concept of child, sibling, parent child relationship etc, binary tree: different types of binary trees based on distribution of nodes, binary tree (threaded and unthreaded) as data structure, insertion, deletion and traversal of binary trees, constructing binary tree from traversal results. B-Trees and introduction to B+ Trees. Graph, Traversersing
Text/References:

1. An introduction to data structures with applications By Jean-Paul Tremblay, P. G. Sorenson, TMH
2. A. Drozdek, Data Structures and Algorithms in C++, 3rd Edition, Course Technology
3. Data Structures in C & C++, Tanenbaum, PHI
4. S. Sahni, Data Structure Algorithms and Applications in C++, Wiley 2003.

Learning Outcomes
After the completion of the course, the students will be able to do the following:
- Use of basic data structures for the storage and retrieval of ordered or unordered data. Data structures include arrays, linked lists, binary trees, heaps, and hash tables.
- Students develop knowledge of applications of data structures including the ability to implement algorithms for the creation, insertion, deletion, searching, and sorting of each data structure.
- Students know searching and sorting algorithms.
- Students can implement projects using different data structures.

CSC-203: Database Management System

Prerequisites to course:

- Object oriented and GUI programming
- Problem solving and structured programming

Objectives:

The objective of the course is to present an introduction to database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information from a DBMS. Also provide fundamental knowledge of, and practical experience with, database concepts. Include study of information concepts and the realization of those concepts using the relational data model. Practical experience gained designing and constructing data models and using SQL to interface to user DBMS packages.

UNIT-I: Introduction to database, Overview and History of DBMS, File System vs DBMS, Purpose of Database, Overall System Structure, Entity Relationship Model, Mapping Constraints - Keys - E-R Diagrams.

UNIT-II: Overview of Data Design Entities, Attributes and Entity Sets, Relationship and Relationship Sets, Features of the ER Model-Key Constraints, Participation Constraints, Weak Entities, Class Hierarchies, Aggregation.
Relationship Algebra: Selection and Projection, Set Operations, Renaming, Joints, Division, Relation Calculus.

UNIT-III: Relational Database Design: Pitfalls, Normalization Using Functional, Dependencies, First Normal Form, Second Normal Form, Third Normal Form and BCNF.

UNIT-IV: Structured Query Language (SQL), Basic Structure, Set Operations, Aggregate, Functions, Date, Numeric, and Character Functions, Nested Sub queries, Modification of Databases, Joined Relations.
Transaction Processing: ACID Properties, Concurrency Control, Recovery.

Text/References:

1. Elmasri R and Navathe SB, Fundamentals of Database Systems, 3rd Edition, Addison Wesley, 2000.
2. Connolly T, Begg C and Strachan A, Database Systems, 2nd Edition, Addison Wesley, 1999
3. Ceri Pelagatti , Distributed Database: Principles and System - (McGraw Hill)
4. Simon AR, Strategic Database Technology: Management for the Year 2000, Morgan Kaufmann, 1995
5. A. Silversatz, H. Korth and S. Sudarsan: Database Cocepts 5th edition, Mc-Graw Hills 2005.

Outcomes:

Upon successful completion of the course, the student will be able to:

- Differentiate database systems from file systems by enumerating the features provided by database systems and describe each in both function and benefit.
- Define the terminology, features, classifications, and characteristics embodied in database systems.

- Student will be able to model an application's data requirements using conceptual modeling tools like ER diagrams and design database schemas based on the conceptual model.
- Student will be able to write SQL commands to create tables and indexes, insert/update/delete data, and query data in a relational DBMS.
- Analyze an information storage problem and derive an information model expressed in the form of an entity relation diagram and other optional analysis forms, such as a data dictionary.
- Demonstrate an understanding of the relational data model.
- Transform an information model into a relational database schema and to use a data definition language and/or utilities to implement the schema using a DBMS.
- Formulate, using SQL, solutions to a broad range of query and data update problems.
- Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database.
- Use an SQL interface of a multi-user relational DBMS package to create, secure, populate, maintain, and query a database.
- Use a desktop database package to create, populate, maintain, and query a database.
- Demonstrate a rudimentary understanding of programmatic interfaces to a database and be able to use the basic functions of one such interface.

Semester V

CSC-301: Computer Networks

Course outline: The designed course covers the topics of computer networks covers the fundamentals of computer networks , basics of signals , convention of signals from analog to digital and from digital to analog. Course will give introduction of OSI Model for computer communication system and practical explore of communication protocol model which is TCP/IP layer architecture. Course will cover detail functionalities and basic services provided by each and every layer.

Objectives of the Course: -

- The course demonstrates OSI and TCP/IP Model.
- Clear understanding of Guided Media characteristics and various Network Topology and Hardware building blocks.
- Demonstration of challenges and issues in Data Link Layer functionalities.
- Demonstration and Explanation of routing algorithms in Network Layer.
- Demonstration of Various Application layer concepts.

UNIT I: Introduction to Networks and Layered Architectures (OSI, TCP/IP), Categories of Networks Network performance measures e.g. bandwidth, latency, Delay/bandwidth product. Transmission Media: Guided Media (twisted pair cable, Coaxial Cable, fibre optic cable), Unguided media (radio waves, microwaves, infrared), Topology. Hardware building Blocks of a network e.g. switches, routers, gateways etc.

UNIT II: Data Link Layer: Data Link Layer Design Issues - Error Detection and Correction. Elementary data link protocols - Sliding Window Protocols - Protocols Verification - Channel Allocation Problem- Multiple Access Protocols

UNIT III: Network Layer: Network Layer Design Issues- Routing Algorithms-Congestion Control Algorithms- Quality of Service -Internetworking
Transport Layer: Transport Services – elements of transport protocols – simple transport protocols.

UNIT IV: Application layers: Domain name system – Electronic mail – The World Wide Web. Introduction to Network security.

Text/References:

1. Computer Networks, Andrew S. Tanenbaum , Fourth edition,PHI private Ltd, New Delhi , 2008
2. Computer Networking Top Down approach 3rd editionBy Jim kurose and keithross

Outcomes:

- Students able to understand various layer functionalities.
- Able to understand various addressing schemes.
- Students able to understand various functionalities of each layer.
- Students able to understand difference between TCP and UDP.
- Application layer protocols are going understand properly.

CSC-302: Theory of Computation

Prerequisites to course: None

Course Outline:

The proposed course describes various kind of automata which are mathematical models to accept or reject sentences belongs to various kinds of formal languages.

Objectives: After studying this course students will be able to design and test a formal automata for different types of languages.

UNIT I: Languages: Alphabets, string, language, basic operations on language, concatenation, Kleene Star

UNIT II: Regular languages model: finite state machine (deterministic and non deterministic), regular grammars, regular expressions, equivalence of deterministic and non deterministic machine and of the three models; Properties: closure, decidability, minimality of automata.

UNIT III: Context Free Grammar, Derivation trees, Simplification of Context Free Grammar, Chomosky Normal Form, Greibach Normal Form, pushdown automata and their equivalence, Properties of Context Free Languages.

UNIT IV: Turing machines, grammars, recursive functions and their equivalence, language acceptability, decidability, halting problem

Text/References:

1. Hofcroft J.E., Ullman J.D., Introduction to Automata Theory, Languages and Computation, Narosa Publishing House.
2. Lewis H. R. and Papadimitriou C. H., Elements of the theory of computation, Pearson Education Asia
3. Martin J. C., Introduction to Languages and the Theory of Computation, 2e, Tata McGraw-Hill .
4. Daniel I A Cohen, Introduction to computer Theory, Wiley II Edition

Learning Outcomes:

- Demonstrate the various kinds of automata models to accept or reject strings belong to the various kinds of formal languages.
- To identify the limitation of the proposed model and try to find the models overcome the limitations.
- Basic understanding of Finite Automata, Push down Automata and Turing Machine Design.
- Various forms to represent the formal languages and simplification of grammar.

CSC-303: Software Engineering

Prerequisites to course: Data base management system & operating system.
Objectives:- The Software Engineering course provides students with knowledge and skills that enable them to design, code, test and manage quality-measured software systems. Software Engineering major includes studying and practicing the software development process, in addition to the algorithm and data process needed to develop innovative software that solves a specific problem.

- Knowledge of basic SW engineering methods and practices, and their appropriate application.
- A general understanding of software process models such as the waterfall and evolutionary models.
- Understanding of software requirements and the SRS documents.
- Understanding of software testing approaches such as unit testing and integration testing.
- Understanding on quality control and how to ensure good quality software.

UNIT I: System Analysis: Characteristics, Problems in system Development, System Level project Planning, System Development Life cycle (SDLC), computer system engineering & system analysis, modeling the architecture, system specification. Capability Maturity Model Integration (CMMI)
UNIT II: Requirement Analysis: Requirement analysis tasks, Analysis principles, Software prototyping and specification data dictionary finite state machine (FSM) models.
Structured Analysis: Data and control flow diagrams, control and process specification behavioural modelling, extension for data intensive applications.
UNIT III: Software Design: Design fundamentals, Effective modular design: Data architectural and procedural design, design documentation, coding – Programming style, Program quality, quantifying program quality, complete programming example
UNIT IV: Testing Strategies and tactics: Testing fundamentals, strategic approach to software testing, Validation testing, system testing, Black-Box testing, white-box testing and their types, basic path testing.

Text/Reference Books

1. R.S. Pressman, Software Engineering: A Practitioner's Approach, Mc Graw Hill
2. P. Jalote, An Integrated Approach to Software Engineering (II Edition)
3. KK Agarwal an Y. Singh, Software Engineering, New Age International Publishers
4. I. Somerville, Software Engineering, Addison Wesle, 2006

Outcomes:-
Upon successful completion of the course, the student will be able to:

- Understand and implement the best practices of software engineering to develop high-performance, maintainable software:
- Capture, analyze and document computing system requirements.
- Translate system requirements into an implementable software design.
- Design test strategies to verify correct and robust software functionality.
- Design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
- Use the techniques, skills, and modern engineering tools necessary for engineering practice.
- Apply knowledge of security issues to the implementation of information technology solutions.

Semester VI

CSC-304: Design & Analysis of Algorithms

Outline of the Course:-

The proposed DAA course covers algorithm and its design strategies. The course will illustrate complexity of designed algorithm in space and time of algorithms. Various Asymptotic notations illustrated during the course. Designed strategies have explained by taking well known algorithms, NP-Complete and NP-Hard topic explained.

Objectives:

- Demonstrate various algorithm analysis parameters to explain best, average and worst case.
- Various strategies to explain to design algorithms.
- Demonstration of various Graph Theory algorithms such as Minimal Spanning Tree and All pair shortest Path.
- Demonstration of P and NP Complete Problems.

UNIT I: Definition & characteristics of algorithms, structures. Difficulties in estimating exact execution time of algorithms. Concept of complexity of program. Asymptotic notations: Big-Oh, theta, Omega- Definitions and examples, Determination of time and space complexity of simple algorithms without recursion. Representing a function in asymptotic notations.

UNIT II: Divide-and-conquer, Dynamic Programming, Greedy methods, Backtracking, Branch-and Bound Technique.

UNIT III: Minimum Spanning Trees, Single-Source Shortest Paths, All-Pairs Shortest Paths, Maximum Flow. String Matching, Computational Geometry.

UNIT IV: P and NP class, NP-completeness and reducibility, NP-complete problems.

Text/References:

1. T. Cormen, C. Leiserson, R. Rivest. Introduction to Algorithms, Indian Reprint, PHI
2. V. Aho, J. Hopcraft, J. Ulmann. The Design and analysis of computer Algorithms. Addison Wesley
3. S. Basse, A. V. Gelder, Computer Algorithms: Introduction to design and Analysis, 3rd., Pearson Education Asia Pvt. Ltd.

Outcomes:-

- Students able to design and analyze algorithms before solving problem.
- Able to decide which algorithms are space and time wise efficient algorithms.
- The main outcome of the course is to decide type of the algorithm which can solve the problem more effectively with less number of computational resources.
- Understand difference between NP-Hard and NP-Complete Problems.

CSC-305: Operating System

Operating systems are the heart of the Computer system. They act as an interface between the Hardware and the user. This course is designed to provide in-depth understanding of the operating systems.

Course Objectives

- Provide basic understanding of the functions and types of operating systems.
- To introduce the concepts of process management, memory management, file management and deadlocks.
- Do practical exercises on scheduling techniques.
- Laboratory exercises to be covered in Lab sessions.

UNIT-I: Introduction to Operating Systems, Types of operating systems, Multiprogramming, Time-sharing systems, Operating system services, System calls and System programs, Storage structures

UNIT II: Process concepts, process scheduling, operations on process, threads, Inter process communication, precedence graphs, critical section problem, semaphores, classical problems of synchronization, CPU Scheduling.

UNIT-III: Memory Management, Single and multiple partitioned allocations, paging segmentation, Virtual Memory Management, Demand paging and Page Replacement Algorithms

UNIT-IV: Deadlock: Introduction, problem, characterization, prevention, avoidance, detection, recovery from deadlock, Methods for deadlock handling.
File concept, Access methods, Directory structure, allocation methods, free space management, disk scheduling,

Text/References:

1. Abraham Silberschatz and P. B. Galvin - Operating system concepts – Addison Wesley Publication
2. A. Tanunbaum, Modern Operating Systems, 3rd Edition, Pearson Publication

Outcomes:-

At the end of this course, the student will be able to:

- Develop in-depth understanding of the functions and concepts related to operating systems.
- Demonstrate understanding of CPU and disk scheduling algorithms.
- Understand how different kinds of Operating systems work.

CSC-306: Computer Systems Architecture

Outline of the Course: -
The outline of the course is basic understand of circuit logic design and storage information in various formats in the memory, various addressing modes and various registers.

Course objectives:
The course is designed to train the graduates in:

- Architecture of digital computers.
- Architecture of various digital units of a computer.
- Usage of digital computers in industry and research.

UNIT I
Digital Logic, Number Systems & codes, Computer Arithmetic: Logic Gates, Boolean Algebra Adder: Half Adder, Full Adder, Flip-flops: S-R, D, J-K and T- Flip Flop.

UNIT II
Digital Component: Multiplexer, Decoder, Encoder, Registers: Shift Register, Counters, Floating Point Arithmetic: Floating point representation, Add, Subtract, Multiplication, Division.

UNIT III
Register Transfer and Micro-operation: Register Transfer, Bus and Memory Transfers, Arithmetic Micro-operation: Binary Adder, Binary Adder Sub tractor and Binary Incremental, Micro-Operations: Logic and Shift, Instruction life cycle.

UNIT IV
Central Progressing Unit (CPU): General Register Organization, Control Word, Example of Micro operation, Stack Organization: Register stack, Memory Stack, Reverse Polish Notation. Instruction Formats, Three, Two, One, Zero Address Instructions, RISC Instructions, Addressing Modes, CISC Characteristics and RISC Characteristics.

Text/References:

1. Computer System Architecture- M. Morris Mano, Pearson Publication 3rd Edition, PHI
2. Computer Organizations and Architecture - William Stallings (Pearson Education Asia), 2008
3. Computer Organization and Architecture -John P. Hayes (McGraw -Hill), 1998
4. Computer Organization -V. Carl. Hamacher (McGraw-Hill), 2011
5. Nicolas Carter, Computer Architecture, Schaum's Series, TMH

Course Outcomes:
Graduates after completing the course shall gain:

- Ability to understand architecture of digital computers.
- Ability to apply digital computers in solving complex problems in industry and research.
- Ability to take up advanced course in Computer Architecture.

Semester VII

CSC-401: INTRODUCTION TO ARTIFICIAL INTELLIGENCE

Prerequisites to course: Discrete Mathematics, Software Engineering.

Objectives & Outline of the course: -
The objective of the course is to present an overview of artificial intelligence (AI) principles and approaches. Develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic, and learning. Students will implement a small AI system in a team environment. The knowledge of artificial intelligence plays a considerable role in some applications students develop for courses in the program.

1. To have a basic proficiency in a traditional AI language including an ability to write simple to intermediate programs and an ability to understand code written in that language.
2. To have an understanding of the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other topics such as minimax, resolution, etc. that play an important role in AI programs.
3. To have a basic understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and robotics, expert systems, and planning.

UNIT 1: Introduction: Introduction to AI, Historical Development, Turing Test. Problem Solving, Search Algorithms, State-space and Solution Space Search, State space as graph-state v/s node; Evaluating Search Strategies- Time, Space, Completeness, Optimality.
Uninformed search: Breadth First Search, Depth First Search, Iterative Deepening Search, Bi-directional Search, Uniform Cost Search.

UNIT 2: Informed search: Best First Search, Heuristic Search, A* Search, Admissible heuristic, Consistent heuristic, optimality and admissibility, IDA* search, Weighted A* search and inconsistency. Hill Climbing, Local Search, Simulated Annealing, local beam search and Genetic Algorithm.

UNIT 3: Adversarial search: Adversarial Search and Game Playing, Min-max Algorithm, Alpha-beta pruning, partially observable games, stochastic games.

UNIT 4: Constraint satisfaction problems: Introduction to CSPs, Constraint Networks, Binary and non-binary constraints, qualitative and quantitative CSPs, Consistencies- Local and global consistencies; Constraint propagation and generalizations – Related Methods: backtracking search; dynamic programming; variable elimination; Handling Spatial and Temporal constraints.

UNIT 5: AI planning: Introduction, complexity, PDDL, Domain Independent Planning, Domain Description, PDDL (syntax), forward vs. backward search, planning graph. Graph Plan,

UNIT 6: Probabilistic reasoning: Uncertainties in AI; Markov random fields; Markov networks; Baye's Theorem; Bayesian networks – Concepts, Representation and Inference; Hidden Markov Model and Dynamic Bayesian Network. Dempster-Shaffer Framework of Evidential Reasoning.

BOOKS

1. Artificial Intelligence: A Modern Approach (third Edition): S. Russel and P. Norvig.
2.Artificial Intelligence: Foundation of Computational Agents: D Poole and AMckworth.

Learning Outcomes:-

Upon successful completion of the course, the student will be able to:

- Design a knowledge based system.
- Explain what constitutes "Artificial" Intelligence and how to identify systems with Artificial Intelligence.
- Ability to apply Artificial Intelligence techniques for problem solving.
- Use classical Artificial Intelligence techniques, such as search algorithms, minimax algorithm, and neural networks.
- Have read and analyzed important historical and current trends addressing artificial intelligence.

CSC-402: DISCRETE STRUCTURES & GRAPH THEORY

Course Outline:

Discrete mathematics is the backbone of computer sciences. It provides the base for algorithm development and semantics of programming languages. This course is designed to given an introductory idea of different discrete structures, including graph theory. In particular, this course introduces logic, proofs, sets, relations, functions, counting, and abstract structure, with an emphasis on applications in computer science.

UNIT I: Sets: Definition and types, Set operations, Partition of set, Cardinality (Inclusion-Exclusion & Addition Principles), Recursive definition of set. Functions, Relations, Properties of binary relations, closure, Partial Ordering Relations, The Pigeonhole & Generalized Pigeonhole Principles, Composition of Functions Concept, Mathematical induction

UNIT II: Graph Theory: Graphs – Directed, Undirected, Simple,. Adjacency & Incidence, Degree of Vertex, Subgraph, Complete graph, Cycle & Wheel Graph, Bipartite & Complete Bipartite Graph, Weighed Graph, Union of Simple Graphs. Complete Graphs. Isomorphic Graphs, Path, Cycles & Circuits Euclerian & Hamiltonian Graphs.

UNIT III: Planar Graph: Kuratowski's Two Graphs, Euler's Formula, Kuratowski's Theorem.
Trees: Spanning trees- Kruskal's Algo, Finding Spanning Tree using Depth First Search, Breadth First Search, Complexity of Graph, Minimal Spanning Tree, Graph Coloring.

UNIT IV: Language of Logic: Proposition, Compound Proposition, Conjunction, Disjunction, Implication, Converse, Inverse & Contrpositive, Bi-conditional Statements, tautology, Contradiction & Contingency, Logical Equivalences, Quantifiers, Arguments Groups, Ring, fields and Lattice

UNIT V: Linear Programming: Linear programming problem, Simplex method, Revised Simplex method, Duality, Dual Simplex, Interior Point Method.

UNIT VI: Combinatorial Optimization Problems: Transportation problem, Assignment problem

Text/Reference Books

1. C.L Liu and D.P. Mohapatra, Elements of Discrete Mathematics: A Computer Oriented Approach, TMH, 3rd Edition
2. Rosen, Discrete Mathematics and its applications, 6th Edition
3. Schaum's Outlines of Discrete Mathematics, Seymour Lipschutz & Marc Lipson, 2nd Edition
4. Narsingh & Deo, Graph Theory with Applications to Engineering and Computer Science, PHI 2004 Publication
6. Combinatorial Optimization -Theory and Algorithms, Bemhard Korte, Jens Vygen.

CSC-403: PROBABILITY AND STATISTICS

Objectives & Outline of the Course:

- To provide students with a formal treatment of probability theory.
- To equip students with essential tools for statistical analyses.
- To foster understanding through real-world statistical applications.

UNIT 1	Elementary probability, Axioms of Probability theory, Probability Spaces, Conditional Probability, Bayesian probabilities. Random variables, Type of random variable and its properties.
UNIT 2	Probability densities, joint densities, marginal densities, conditional densities, Expectation and covariance with properties. Binomial and Normal distributions
UNIT 3	Descriptive statistics, presentation of data, averages, measures of variation. Sampling distributions, Statistical inference, estimation, confidence intervals, exponential family of distribution, testing of hypotheses, nonparametric methods.
UNIT 4	Linear regression and correlation. Decision theory, basic terminology and steps involved in decision making process, decision making environment. Introduction to information theory.

BOOKS

1. Probability and Computing, by Michael Mitzenmacher and Eli Upfal, Cambridge University Press
2. Christopher M. Bishop. Pattern Recognition and Machine Learning, Springer, 2006. (For first two chapters)
3. Artificial Intelligence: A Modern Approach (third Edition): S. Russel and P. Norvig.

Learning Outcomes: At the end of the course students can be able to

1. understand the terminologies of basic probability, two types of random variables and their probability functions.
2. observe and analyze the behavior of various discrete and continuous probability distributions.
3. Present the analysis of derived statistics to all audiences.
4. Infer the statistical inferential methods based on sampling tests.
5. Interpret the association of characteristics and through correlation and regression tools.

CSC-404: Advanced Algorithms		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	Data Structure and Design Analysis and Algorithm

Course Objectives:

	The course is designed to train the graduates in: • Advanced topics in algorithm. • To develop concept, ideas for any problem. • To be enable to formalize with theoretical computer algorithms.

Course Outcomes: The student will be able to

1	Ability to understand algorithms
2	Ability to develop concepts, logics towards solving a unknown problem in IT and research.
3	Ability to get formalizes theoretical concepts of computer algorithms.
4	Elabrate advanced techniques for the design and analysis of algorithms, and explores a variety of applications

Course Content:

UNIT- 1	Design Paradigms Overview: Overview of complexity notations, Divide and Conquer method, Greedy and Dynamic Programming	**(08 Hours)**
UNIT- 2	Backtracking, Branch and Bound, Max Flow Problem, String Matching etc.	**(08 Hours)**
UNIT- 3	Brief overview of Notations and Recurrence analysis, Amortized analysis, B- Trees, AVL trees	**(08 Hours)**
UNIT- 4	Dictionaries and tries, Binomial Heaps, Fibonacci Heaps, Disjoint Sets, Union by Rank and Path Compression	**(06 Hours)**
UNIT- 5	Randomized Algorithms and Parallel Algorithms: Randomized Algorithms: Las Vegas and Monte Carlo algorithms, Applications on graph problems, Finger Printing, Pattern Matching, Primality testing algorithm	**(06 Hours)**
UNIT- 6	Introduction, Combinatorial optimization, approximation factor, PTAS, FPTAS, Approximation algorithms for vertex cover, set	**(06 Hours)**

	cover, TSP, subset-sum problem etc., Analysis of the expected time complexity of the algorithms	
Theory Assessment:		
UNIT TEST-1	UNIT- 1, 2, 3	
UNIT TEST-2	UNIT- 4, 5, 6	
EoSE	UNIT-1,2,3,4,5,6	
Term Work:		
	Assignments: Students should perform experimental assignment/s from the list below	
	1) Implement all sorting algorithms and analyze complexity	
	2) Implement Stassen's matrix multiply and Greedy Approach	
	3) Implement Fermat theorem, square root test algo, Miller-Rabin primality testing algo	
	4) Implement randomised search algo for las vegas and Monti Carlo algo, fingerprinting algo, pattern matching algo	
	5) Implement all sorting algorithms and analyze complexity	
	6) Implement Stassen's matrix multiply and Greedy Approach	
Text Books:		
1. Introduction to Algorithms: T.H. Cormen, C. E. Leiserson and R.L. Rivest		
2. Fundamentals of Algorithmic : G. Brassard and P. Bratley		
3. Approximation Algorithms: Vijay V.Vazirani		
Reference Books:		
(1) Randomized Algorithms: R. Motwani and P.Raghavan (2) Parallel Computing: Theory and Practice: M. J. Quinn (3) Introduction to Parallel Computing: T. G. Lewis and H. El-Rewini		
Topics for Project based learning: Analysis of theoretical computation.		

CSC-405: Programming in Python

Course Outline: Python programming develops the basic skills of programming using python.

Course Objectives

- To introduce the basics of Python programming.
- Laboratory exercises to cover in Lab sessions.

UNIT I: Basics of python programming: python identifiers, indentation, comments in Python, data types, python strings.

UNIT II: Python operators: arithmetic, assignment, relational operators etc. Decision making and loop control structures.

UNIT III: Built-in functions in python, built-in string methods. User-defined functions, keyword arguments. Lambda functions.

UNIT IV: Python lists, tuples, dictionaries. Performing basic operations on lists, tuples and dictionaries.

UNIT V: Python modules, namespace and scoping. File handling, access modes, reading and writing files, renaming and deleting files.

UNIT VI: Plotting graphs in python, Introduction to Matplotlib. Developing basic GUI applications using Tkinter.

Text/References

1. Introduction to computation and programming using Python, John V. Guttag, MIT Press.

Learning Outcomes:- At the end of this course, the student will be able to:

- Develop basic programs in Python.
- Plot graphs using Python.

CSC-406 -PROFESSIONAL COMMUNICATION

Outline

The course has been designed keeping in mind the communicative needs of the students as lack of proficiency and fluency of students is one of the major barriers in getting employment in the job market.

Objectives

The objectives of the course are:

- to make the student proficient and fluent in speaking
- to enable the student to comprehend what is spoken and written
- to ensure that they become fast readers
- to make them handle basic correspondence effectively
- to enhance their vocabulary base

Course Content:-

UNIT-1:
Grammar and Vocabulary: Tenses, subject–verb agreement. Sentence Analysis: Simple, Compound and Complex sentences. Phrases: Adjective, Adverb and Noun Phrase, Clauses: Adjective, Adverb and Noun Phrase. Voice, Narration, Gerund, Participle.

Unit-2: Oral Communication

UNIT-3
Listening Skill – Active listening, Barriers to active listening**.**
Speaking Skill-Stress patterns in English, Questioning skills, Barriers in Speaking**.**

UNIT-4: Reading Skill-Skimming, Scanning, Intensive reading, linking devices in a text, Different versions of a story/ incident**.**

UNIT-5:
Written communication:
Writing process, paragraph organization, writing styles.Types of Writing **-** Technical vs. creative; Types of technical writing, Scientific Writing:Writing a Scientific Report Soft Skills:

Unit-6: Body Language– Gesture, posture, facial expression.
Group Discussion– Giving up of PREP, REP Technique.
Presentation Skills:
(i) How to make power point presentation (ii) Body language during presentation (iii)
Resume writing:Cover letter, career objective, Resume writing (tailor made)
Interview Skills: Stress Management, Answering skills.

BOOKS:
1. Advanced English Usage: Quirk & Greenbaum; Pearson Education.
2. Developing Communication Skills: Banerjee Meera & Mohan Krishna; Macmillan Publications, 1990.
3. Business Communication: Chaturvedi, P.D.; Pearson Publications.
4. Business Communication; Mathew, M.J.; RBSA Publications, 2005.
5. Communication of Business; Taylor, Shirley; Pearson Publications.
6. Soft Skills: ICFAI Publication
7. Collins English Dictionary and Thesaurus, Harper Collins Publishers and Times
8. Longman Language Activator, Longman Group Pvt Ltd
9. Longman Dictionary of contemporary English, Longman
10. The new Penguin Dictionary – a set of dictionaries of abbreviations, spelling, punctuation, plain English, grammar, idioms, thesaurus, 2000.
11. New Oxford Dictionary.
12. Wren & Martin: High School English Grammar and Composition
13. Raymond Murphy: English Grammar in Use (4th edition)
14. Martin Hewings: Advanced Grammar in Use
15. Betty Schrampfer: Understanding and Using English Grammar

Learning Outcomes:
After completion of the course student will become fluent speakers. Not only this, they will be able to comprehend the spoken and the written word in a better way. With enhanced vocabulary they will become confident users of English and be more market-ready to get a job.

CSC-408: Advanced Algorithms Lab		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Practical: 4 hours / Week	Term Work & OR: 50 Marks	Term Work & OR: 2
		Total: 2

Course Pre-requisites: The students should have knowledge of	
1	Data Structure and Design Analysis and Algorithm

Course Objectives:	
	The course is designed to train the post graduates in: • Advanced topics in algorithm. • To develop concept, ideas for any problem. • To be enable to formalize with theoretical computer algorithms.

Course Outcomes: The student will be able to	
1	Ability to understand Implementation of algorithms
2	Ability to Implementation concepts, logics towards solving a unknown problem in IT and research.
3	Ability to get formalizes theoretical concepts of computer Implementation.
4	Elabrate advanced techniques to implement for the design and analysis of algorithms, and explores a variety of applications

Course Content:		
	Assignments: Students should perform experimental assignment/s from the list below	
	Implement all sorting algorithms and analyze complexity	
	Implement Divide and conquer Greedy Approach , Dynamic Programing	
	Implement Fermat theorem, square root test algo, Miller-Rabin primality testing algo	
	Implement randomised search algo for las vegas and Monti Carlo algo, fingerprinting algo, pattern matching algo	
	Design, develop and run program in any language to implement the Bellman-Ford algorithm and determine its performance.	

Text Books:
1. Introduction to Algorithms: T.H. Cormen, C.E.Leiserson and R.L. Rivest
2. Fundamentals of Algorithmic : G. Brassard and P. Bratley

3. Approximation Algorithms: Vijay V.Vazirani

Reference Books:

(1) Randomized Algorithms: R. Motwani and P.Raghavan
(2) Parallel Computing: Theory and Practice: M. J. Quinn
(3) Introduction to Parallel Computing: T. G. Lewis and H. El-Rewini

Topics for Project based learning: Analysis of theoretical computation.

CSC-409: Machine Learning

TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4

Course Pre-requisites: The students should have knowledge of

1	Python Programing

Course Objectives: This course will enable students to,

- Define machine learning and understand the basic theory underlying machine learning.
- Differentiate supervised, unsupervised and reinforcement learning
- Understand the basic concepts of learning and decision trees.
- Understand neural networks and Bayesian techniques for problems appear in machine learning
- Understand the instant based learning and reinforced learning
- Perform statistical analysis of machine learning techniques.

Course Outcomes: The student will be able to

1	Choose the learning techniques and investigate concept learning
2	Identify the characteristics of decision tree and solve problems associated with
3	Apply effectively neural networks for appropriate applications
4	Apply Bayesian techniques and derive effectively learning rules
5	Evaluate hypothesis and investigate instant based learning and reinforced learning

Course Content:

UNIT- 1	Basics: Introduction to Machine Learning - Different Forms of Learning, Basics of Probability Theory, Linear Algebra and Optimization. Regression Analysis: Linear Regression, Ridge Regression, Lasso, Bayesian Regression, Regression with Basis Functions.	**(08 Hours)**
UNIT- 2	Classification Methods: Instance-Based Classification, Linear Discriminant Analysis, Logistic Regression, Large Margin Classification, Kernel Methods, Support Vector Machines, Multi-class Classification, Classification and Regression Trees.	**(08 Hours)**
UNIT- 3	Neural Networks: Non-linear Hypotheses, Neurons and the Brain, Model Representation, Multi-layer Networks, Back-propagation, Multi-class Discrimination, Training Procedures, Localized Network Structure, Deep Learning.	**(08 Hours)**
UNIT- 4	Graphical Models: Hidden Markov Models, Bayesian Networks, Markov Random Fields, Conditional Random Fields. Ensemble Methods:Boosting - Adaboost, Gradient Boosting, Bagging - Simple Methods, Random Forest.	**(08 Hours)**
UNIT- 5	Clustering:Partitional Clustering - K-Means, K-Medoids, Hierarchical Clustering - Agglomerative, Divisive, Distance Measures, Density Based Clustering – DBscan, Spectral Clustering.	**(08 Hours)**
UNIT- 6	Dimensionality Reduction: Principal Component Analysis, Independent Component Analysis, Multidimensional Scaling, and Manifold Learning. Reinforcement Learning: Q-Learning, Temporal Difference Learning	**(08 Hours)**
Assessment:		
CIA -1	UNIT TEST- 1 :- UNIT- 1, 2, 3	
CIA -2	UNIT TEST- 2 :- UNIT- 4, 5, 6	
EOSE	UNIT 1,2,3,4,5,6	
Text Books:		
Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education.		
Pattern Classification. R.O. Duda, P.E. Hart and D.G. Stork.		
Data Mining: Tools and Techniques. Jiawei Han and Michelline Kamber.		
Elements of Statistical Learning. Hastie, Tibshirani and Friedman. Springer		

CSC-410: Big Data Analytics		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week		Lab: 1
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	Basic knowledge of programming concepts and logics.

Course Objectives:

- To understand the need of Big Data, challenges, and different analytical architectures
- Installation and understanding of Hadoop Architecture and its ecosystems
- Processing of Big Data with Advanced architectures like Spark.

Course Outcomes: The student will be able to

- Discuss the challenges and their solutions in Big Data
- Understand and work on Hadoop Framework and eco systems.
- Explain and Analyse the Big Data using Map-reduce programming in Both Hadoop and Spark framework.
- Demonstrate spark programming with different programming languages.

Course Content:

UNIT- 1	Introduction to Big Data, Types of Digital Data, Characteristics of Data, Evolution of Big Data, Data Storage and Analysis, Characteristics of Big Data, Big Data Architecture, Requirement for new analytical architecture, Challenges in Big Data analytics, Need of big data frameworks	**(7 Hours)**
UNIT- 2	Requirement of Hadoop Framework, Design principle of Hadoop, Comparison with other system, Hadoop Components, Hadoop versions, HDFS, Map Reduce Programming: I/O formats, Map side join, Reduce Side Join, secondary sorting, Pipelining MapReduce jobs	**(7 Hours)**
UNIT- 3	Introduction to Hadoop ecosystem technologies: Serialization: AVRO, Coordination: Zookeeper, Databases: HBase, Hive, Scripting language: Pig, Streaming: Flink, Storm	**(7 Hours)**
UNIT- 4	MapReduce workflows, unit tests with MRUnit, test data and local tests, anatomy of MapReduce job run, classic Map-reduce, YARN, failures in classic Map-reduce and YARN, job scheduling, shuffle and sort, task execution, MapReduce types, input formats, output formats.	**(8 Hours)**

UNIT- 5	Spark Framework, Writing Spark Application, Spark Programming in Scala, Python, R, Java, Application Execution	**(8 Hours)**
UNIT- 6	SQL Context, Importing and Saving data, Data frames using SQL, GraphX overview, Creating Graph, Graph Algorithms	**(8 Hours)**
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s from the list below	
	• HDFS Commends Map Reduce Program to show the need of Combiner • Map Reduce I/O Formats-Text, key-value Map ReduceI/O Formats – Nline, Multiline • Sequence file Input/Output Formats Secondary sorting • Distributed Cache & Map Side Join, Reduce side Join Building and Running a Spark Application Word count in Hadoop and Spark Manipulating RDD • Spark Sql programming, Building Spark Streaming application	

Text Books:

- Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, John Wiley & Sons, 2013.
- Douglas Eadline,"Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", Pearson Education, 2016.

Reference Books:

- Mike Frampton, "Mastering Apache Spark", Packt Publishing, 2015.
- TomWhite,"Hadoop:The Definitive Guide",O'Reilly,4thEdition,2015.
- NickPentreath,MachineLearningwithSpark,PacktPublishing,2015.
- Mohammed Guller, Big Data Analytics with Spark, Apress,2015
- Donald Miner, Adam Shook, "Map Reduce Design Pattern", O'Reilly, 2012

e-Resources

- https://www.coursera.org/specializations/big-data

CSC-411: NATURAL LANGUAGE PROCESSING

Objectives:

- To understand Levels of Language Analysis, Organization of Natural language Systems
- To learn Linguistic Background: An outline of English syntax.
- To learn Grammars and Parsing, Morphological Analysis, Parsing with Features, Various Lexicon Resource & Knowledge Source
- To understand Grammars for Natural Language, Ambiguity Resolution

UNIT I : Introduction to Natural Language Understanding: The study of Language, Evaluating Language Understanding Systems, Different levels of Language Analysis, Representations and Understanding, Organization of Natural language Understanding Systems

UNIT II: Linguistic Background: An outline of English syntax.

UNIT IIII: Grammars and Parsing: Grammars and sentence Structure, Top-Down and Bottom-Up Parsers, Transition Network Grammars, Top-Down Chart Parsing, Morphological Analysis and the Lexicon.

UNIT IV: Parsing with Features, Augmented Transition Networks, Various Lexicon Resource & Knowledge Source, Study of Word Net and Indo Net

UNIT V: Grammars for Natural Language: Auxiliary Verbs and Verb Phrases, Movement Phenomenon in Language, Handling questions in Context-Free Grammars, Hold mechanisms in ATNs. Human preferences in Parsing, Encoding uncertainty, Deterministic Parser, Study of POS Tagger, Stemmer

UNIT VI
Ambiguity Resolution: Statistical Methods, Estimating Probabilities, Part-of-Speech tagging, Obtaining Lexical Probabilities, Probabilistic Context-Free Grammars, Best First Parsing. Semantics and Logical Form: Word senses and Ambiguity, Encoding Ambiguity in Logical Form. Discourse Analysis and Pragmatic Analysis

Books Recommended:
1. JAMES ALLEN, Natural Language Understanding, 2/e, Pearson Education, 2003.
2. D. JURAFSKY, J. H. MARTIN, Speech and Language Processing, Pearson Education, 2002.
3. CHRISTOPHER D. MANNING, HINRICH SCHÜTZE, Foundations of Statistical Natural Language Processing, The MIT Press, Cambridge, Massachusetts.1999.
4. U. S. TIWARY, TANVEER SIDDIQUI, Natural Language Processing and Information Retrieval, Oxford University Press (2008).
5. AKSHAR BHARATI, VINEET CHAITANYA, RAJEEV SANGAL, Natural Language Processing: A Paninian Perspective

Outcome: After completion of this course students will be able to design a model of a prototype language.

CSC-412: SOFT COMPUTING

Pre-requisites: *Basic understanding of Neural Networks*

Soft computing is an emerging approach to computing which can parallel the remarkable ability of the human mind to reason and learn in an environment of uncertainty and imprecision. Soft computing is based on some biologically inspired computing paradigms such as Genetic Algorithms, Evolutionary computing, Particle Swarm Optimization, human nervous system, etc. Applying Soft Computing can help in solving real-time problems that cannot be solved using mathematical modeling. Varied applications of Soft Computing include Computer Vision, medical diagnosis, pattern recognition, network optimization etc.

Course Objectives

This course will uncover the fundamental concepts used in Soft computing. The concepts of fuzzy logic, Neuro-fuzzy computing, genetic algorithms will be discussed. Applications of Soft Computing techniques to solve a number of real life problems will be covered to have hands on practices. In summary, this course will provide exposure to theory as well as practical systems and software used in soft computing. Broad Objectives of the course are:

- To develop understanding of Fuzzy logic and its applications.
- Solve problems using Neuro-fuzzy computing and its applications.
- In-depth study and discussion on Genetic Algorithms and Evolutionary computing.
- Apply Particle Swarm optimization, ant colony optimization and other algorithms to real life problems.
- Laboratory exercises to be covered in Lab sessions.

Unit 1: Foundations: Stochastic processes; Principal Component Analysis; Learning theory; Generalization and Regularization; Simulated Annealing.
Unit 2: Fuzzy Sets, membership functions in one dimension, membership functions in two dimensions.
Unit 3: Fuzzy relations, fuzzy if-then rules: single rule with single antecedent, single rule with multiple antecedents, multiple rules with multiple antecedents. Fuzzy reasoning.
Evolutionary algorithms and genetic programming
Unit 4: Fuzzy inference system.
Unit 5: Evolutionary computation, Genetic algorithms, encoding, selection, crossover and mutation.
Unit 6: Particle swarm optimization, Artificial Bee colony search, Ant colony algorithm and similar algorithms.

Books

1.Neuro Fuzzy & Soft Computing - J.-S.R.Jang, C.-T.Sun, E.mizutani, Pearson Education
2.Digital Neural Network - S.Y.Kung, Prentice Hall International Inc.
3.Spiking Neural Networks - Wulfram Gerstner, Wenner Kristler, Cambridge University Press.
4.Neural Networks and Fuzzy Systems: Dynamical Systems Application to Machine Intelligence - Bart Kosko, Prentice Hall.

Outcomes:-

After completing this course, you will be able to learn:

- Understand the basic concepts of Soft Computing.
- Solve real life problems using fuzzy logic.
- Implement genetic and evolutionary computing in different scenarios.

Analyze the applicability of Particle swarm optimization and other swarm optimization techniques in practical situations.

CSC-413: Dissertation - I

Outline: The students will be select a supervisor, as per the suggestion of student will select a research field, in that field student will be instructed to read research papers from reputed journals form IEEE and Elsevier or any other peer reviewed journals. The student will write a Dissertation on review of the research work.

CSC-414: Machine Learning Lab		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	Term Work & OR: 50 Marks	Term Work & OR: 2

Course Pre-requisites: The students should have knowledge of

1	Python Programing

Course Objectives: This course will enable students to,

- Make use of Data sets in implementing the machine learning algorithms
- Implement the machine learning concepts and algorithms in any suitable language of choice

Course Outcomes: The student will be able to

1	Understand the implementation procedures for the machine learning algorithms
2	Design Python programs for various Learning algorithms.
3	Apply appropriate data sets to the Machine Learning algorithms
4	Identify and apply Machine Learning algorithms to solve real world problems

Course Content:

Ex- 1	Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file(Given by Examiner)	
Ex- 2	For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the	

	set of all hypotheses consistent with the training examples	
Ex-3	Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.	
Ex-4	Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.	
Ex-- 5	Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.	
Ex-- 6	Dimensionality Reduction: Principal Component Analysis, Independent Component Analysis, Multidimensional Scaling, and Manifold Learning. Reinforcement Learning: Q-Learning, Temporal Difference Learning	
Ex-- 7	Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.	
Ex-- 8	Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.	
Ex-- 9	Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.	
Ex-- 10	Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.	
Reference Books:		
Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming		
Python Programming: An Introduction to Computer Science by John M Zelle		
Machine Learning for Hackers by Drew Conway and John Myles White		

Semester IX

CSC-501: Data Warehousing and Mining		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week	Term Work & OR:	Term Work & OR: 1
		Total: 4

Course Pre-requisites:

1	The students should have basic knowledge of Mathematics, Statistics and Machine Learning.

Course Objectives:

	To develop the basic understanding of Data Mining Algorithms, Applications of Data Mining Algorithms and Information Retrieval

Course Outcomes: The student will be able to

1	Develop the skills to gain a basic understanding of Data Mining Algorithms and their Applications
2	Introduce students to Data Mining Algorithms from the engineering perspective.
3	To give design methodologies for Data Mining Algorithms
4	To provide knowledge for model tuning and over fitting avoidance
5	To understand the Data Mining Algorithms and Information Retrieval their implementations
6	To demonstrate Data Mining and Information Retrieval Algorithms applications in solving the real-world tasks

Course Content:

UNIT- 1	**Fundamental of Data Mining:** Data Mining, History of Data Mining, Data Mining Strategies, Data Mining Techniques, KDD process, Applications of Data Mining, Challenges and Future of data mining. **Data Preprocessing and Data Warehousing:** Data, information, knowledge, and intelligence, Types of data, Data warehouses, Data cleaning, Data de-normalization, Data transformation, Data quality measures, OLAP technology, OLAP vs OLTP. Data Sampling.	**(10 Hours)**
UNIT- 2	**The Classification Task:** Introduction to classification, Decision trees, Random forests, Naïve Bayes', K-NN, SVM, ANNs, applications of classification. Model evaluation techniques- ROC, Lift Charts, cost and utility, Parsimony, Bagging and Boosting, The model ranking approach.	**(10 Hours)**
UNIT- 3	**The Clustering Task:** Introduction to clustering, Distance measures, types of clustering-hierarchical: agglomerative and divisive, Non-hierarchical: Partition	**(10 Hours)**

	based, Density based, Probability based, K-means clustering, Self-organizing concept, self-organizing maps, SOM algorithm, cluster validation, strength and weaknesses of clustering algorithms, applications of clustering.	
UNIT- 4	**Association Rule Mining:** Concepts of association rules, relevance and functions of association rules**,** the problem of large data set, Apriori algorithm, scalable association rule mining-FP-Growth algorithm, Applications of ARM, strength and weaknesses of ARM.	**(10 Hours)**
UNIT- 5	**Information Retrieval**: Boolean Retrieval, The term vocabulary and postings lists: Document delineation and character sequence decoding, determining the vocabulary of terms. Dictionaries and tolerant retrieval: Search structures for dictionaries, spelling correction. Scoring, term weighting and vector space model, the vector space model for scoring, variant tf-idf functions.	**(10 Hours)**
UNIT- 6	**Computing scores in a complete search system**: Efficient scoring and ranking, components of an information retrieval system. Evaluation in information retrieval. Relevance feedback and query expansion: Relevance feedback and pseudo relevance feedback, global methods for query reformulation.	**(10 Hours)**
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4,5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s from the list below	
	1. Implementation of data mining and Information Retrieval algorithms	
	2. Classification using Decision Trees, Naïve Bayes', K-NN, SVM etc.	
	3. Clustering using K-means, SOM, AGNES, DIANA, DBSCAN etc.	
	4. Image and data classification using data mining algorithms	
	5. Association Rule Mining using Apriori and FP-growth	
Term Work:		
Part - A		
Text Books: Data Mining Concepts & techniques: Jai wei Han and Micheline Kamber, Morgan Kaufman. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schutze, Introduction to Information Retrieval, Cambridge University Press		
Reference Books:		
Data Mining Techniques : Arun K. Pujari, Universities Press, Fourth Edition, ck and ps2016.		

Mastering Data Mining: M. Berry and G. Linoff, John Wiley & Sons., 2000
Data Mining: Methods and Techniques: A B M Shawkat Ali, Saleh A. Wasimi, 2009, Cengage Learning
Topics for Project based learning

CSC-502: Neural networks & Deep learning		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week	Term Work & OR:	Term Work & OR: 1
		Total: 4

Course Pre-requisites:	
1	The students should have basic knowledge of Algebra, Statistics and Machine Learning.
Course Objectives:	
	To develop the basic understanding of ANNs and various ANN Algorithms, Applications of ANN Algorithms in Machine Learning and data analysis.
Course Outcomes: The student will be able to	
1	Develop the skills to gain a basic understanding of neural network theory and its applications.
2	Introduce students to artificial neural networks an engineering perspective
3	To give design methodologies for artificial neural networks
4	To provide knowledge for network tuning and over fitting avoidance
5	To offer neural network implementations
6	To demonstrate neural network applications on real-world tasks

Course Content:		
UNIT- 1	**Overview of biological neurons:** Structure of biological neurons relevant to ANNs. Fundamental concepts of Artificial Neural Networks: Models of ANNs; Feed-forward & feedback networks; learning rules; Hebbian learning rule, perception learning rule, delta learning rule, Widrow-Hoff learning rule, correction learning rule, Winner –lake all learning rule, etc.	**(10 Hours)**

UNIT- 2	**Single layer Perception Classifier:** Classification model, Features & Decision regions; training & classification using discrete perceptron algorithm, single layer continuous perceptron networks for linearly separable classifications.	**(10 Hours)**
UNIT- 3	**Multi-layer Feed forward Networks:** linearly non-separable pattern classification, Delta learning rule for multi-perceptron layer, generalized delta learning rule, Error back-propagation training, learning factors, Examples. Single layer feedback Networks: Basic Concepts, Hopfield networks, Training & Examples	**(10 Hours)**
UNIT- 4	**Self-Organizing Networks:** Introduction, Self-organizing concept, self-organizing maps, SOM algorithm, adaptive resonance theory (ART), ART algorithm, and variations of ART algorithm. Adaptive pattern classification.	**(10 Hours)**
UNIT- 5	**Associative memories:** Linear Association, Basic Concepts of recurrent. **Auto associative memory**: retrieval algorithm, storage algorithm; Bi-directional associative memory, Architecture, Association encoding & decoding, and Stability.	**(10 Hours)**
UNIT- 6	**Introduction to Deep Learning:** Deep learning vs. machine learning, significance of deep learning, deep neural networks vs. traditional neural networks. Convolutional neural networks (CNNs): convolution, pooling, padding, and stride. Image classification using CNNs.	**(10 Hours)**
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4,5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s from the list below	
	1. Implementation of Neural networks algorithms	
	2. Classification using Backpropagation	
	3. Clustering using SOM, ART	
	4. Image and data classification using CNN	
Term Work:		
Part – A		
Text Books: Neural networks a comprehensive foundation, Simon Haykin, Pearson Education 2nd Edition 2004.		
Reference Books:		
Artificial neural networks - B.Vegnanarayana Prentice Halll of India P Ltd 2005		
Neural networks in Computer intelligence, Li Min Fu TMH 2003		

"Neural Networks, Fuzzy Logic and Genetic Algorithms", S. Rajasekaran and G. A. V. Pai, PHI, 2003.
Introduction to artificial neural systems", Jacek M. Zurada, 1994, Jaico Publ. House.
Topics for Project based learning

CSC-503: Image Processing and Computer Vision		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	• Some exposure to MATLAB/Python and Open • Knowledge of basic matrix theory (linear algebra) would be helpful, but not necessary

Course Objectives:

	• know the fundamental techniques for image processing, video processing, and computer vision • understand the basics of analog and digital video: and apply machine learning in the field of computer vision.

Course Outcomes: The student will be able to

1	know the fundamental techniques for image processing, video processing, and computer vision
2	understand the basics of analog and digital video: video representation and transmission
3	acquire the basic skill of designing image/video compression
4	Familiarize himself/herself with image/video compression standards

Course Content:

UNIT- 1	Overview of image processing systems, Image formation and perception, Continuous and digital image representation, Image quantization: uniform and non-uniform, visual quantization (dithering).	**(08 Hours)**
UNIT- 2	Image contrast enhancement: linear and non-linear	**(08 Hours)**

	stretching, histogram equalization, Continuous and discrete-time Fourier Transforms in 2D; and linear convolution in 2D.	
UNIT- 3	Image smoothing and image sharpening by spatial domain linear filtering; Edge detection, Discrete Fourier transform in 1D and 2D, and image filtering in the DFT domain.	**(08 Hours)**
UNIT- 4	Median filtering and Morphological filtering, Color representation and display; true and pseudo color image processing, Image sampling and sampling rate conversion (resize).	**(08 Hours)**
UNIT- 5	Image segmentation and Feature Extraction Various methods of image segmentation, edge detection, object proposals, SIFT features. Multi-view Geometry (2 weeks) Shape from stereo and motion, feature matching, surface fitting, Active ranging Object Recognition: Traditional Methods HoG/SIFT features, Bayes classifiers, SVM classifiers	**(08 Hours)**
UNIT- 6	Object Recognition: Deep Learning Methods : Image classification, object detection and semantic segmentation, adversarial attacks. Various neural network architectures, visualization techniques. Motion analysis and Activity Recognition: Motion detection and tracking, Inference of human activity from image sequences	**(08 Hours)**
Assessment:		
CIA-1	UNIT TEST- 1 :- UNIT- 1, 2, 3	
CIA-2	UNIT TEST- 2 :- UNIT- 4, 5, 6	
EOSE	UNIT 1, 2, 3, 4, 5, 6	
Text Books:		
Forsyth and Ponce, “Computer Vision – A Modern Approach”, Second Edition, Prentice Hall, 2011.		
Emanuele Trucco and Alessandro Verri, “Introductory Techniques for 3-D Computer Vision”, Prentice Hall, 1998.		
Olivier Faugeras, “Three Dimensional Computer Vision”, MIT Press, 1993.		
Reference Books:		
Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer, 2011		
Milan Sonka, Vaclav Hlavac and Roger Boyle, “Image Processing, Analysis and Machine Vision”, Third Edition, CL Engineering, 2013.		

ELECTIVES

CSC-331: COMPUTER GRAPHICS

Objective:
After completion of this course students will be able to draw 2 dimensional graphical objects using geometrical algorithms and performs operations on them.

S. No.	Content	Hours
1	**Basic of Computer Graphics:** Basic of Computer Graphics, Applications of computer graphics, Display devices, Random and Raster scan systems, Graphics input devices, Graphics software and standards	6
2	**Graphics Primitives:** Points, lines, circles and ellipses as primitives, scan conversion algorithms for primitives, Fill area primitives including scanline polygon filling, inside-outside test, boundary and flood-fill, character generation, line attributes, area-fill attributes, character attributers. Aliasing, and introduction to Anti Aliasing (No anti aliasing algorithm).	7
3	**Two Dimensional Graphics:** Transformations (translation, rotation, scaling), matrix representation, homogeneous coordinates, composite transformations, reflection and shearing, viewing pipeline and coordinates system, window-to-viewport transformation, clipping including point clipping, line clipping (Cohen-Sutherland, liang- bersky, Mid-point), polygon clipping	8
4	**Three Dimensional Graphics:** 3D display methods, polygon surfaces, tables, equations, meshes, curved lies and surfaces, quadric surfaces, spline representation, cubic spline interpolation methods, Bezier curves and surfaces, B-spline curves and surfaces.3D scaling, rotation and translation, composite transformation, viewing pipeline and coordinates, parallel and perspective transformation, view volume and general (parallel and perspective) projection transformations.	8
5	**Illumination and Colour Models:** Light sources – basic illumination models – halftone patterns and dithering techniques; Properties of light – Standard primaries and chromaticity diagram; Intuitive colour concepts – RGB colour model – YIQ colour model – CMY colour model – HSV colour model – HLS colour model; Colour selection.	6
6	**Animations & Realism:** Design of Animation sequences – animation function – raster animation – key frame systems – motion specification – morphing – tweening. **Fractal Graphics:** Tiling the plane, Recursively defined Curves, Koch curves, C curves, Dragons, space filling curves, turtle graphics, ray tracing.	6
	Total	41

Text Book/References:

1. Computer graphics, Donald Hearn and Paulin Baker.
2. Computer graphics, Schaum's Series.
3. Computer graphics: Principles and Practice - James D Foley.
4. Principles of Interactive Computer Graphics – William M Newman
5. Mathematical Elements for Computer Graphics, David Rogers, J Alan Adams
3. Geometric Tools for Computer Graphics - Philip J Schneider, David H Eberly.

Learning Outcomes:
1. To understand basic display Devices, Input Devices.
2. To learn Line Drawing Algorithms, Circle and Ellipse Drawing Algorithms, 2D Transformations, Line and Polygon clipping, Color Fill Methods, 2D Projections.
3. To learn introduction of Fractal Graphics.

CSC-332: Internet Technologies

Internet Technologies aide in developing applications in the today's WWW world.

Course Objectives

3. To introduce the basics of Website designing languages like HTML and JavaScript.
4. To develop applications using Java.
5. Do Advanced Java programming using servlets, RMI, JDBC and swings.
6. Laboratory exercises to cover in Lab sessions.

UNIT I: The Internet: history of the World Wide Web.
Introduction of HTML HTML: common tags and controls, HTML basics, HTML forms, more complex HTML forms, internal linking, creating and using image maps.

UNIT II: Java script –Introduction to scripting: introduction- memory concepts- arithmetic- decision making. Java script control structures, Java script functions: introduction – program modules in javascript - function definitions, duration of identifiers, scope rules, recursion, java script global functions.
Java script arrays: introduction, array-declaring and allocating arrays, references and reference parameters – passing arrays to functions, multi-dimensional arrays. Java script objects: introduction, math, string, data, Boolean and number objects.

UNIT III: Introduction to Cascading style sheets and latest technologies in web development.

UNIT IV: An overview of Java: Object oriented programming, Two paradigms, abstraction, OOP principles, Java class libraries, Date types, variables and arrays, Operators.
Class fundamentals, declaring object reference variable, Introducing methods, constructors, the keywords, garbage collection, the finalize () method. Overloading methods, using objects as parameters, classes, using exceptions. Inheritance and polymorphism, String handling, Exception handling.

Text/References

1. Herbert Schildt: JAVA 2 - The Complete Reference, TMH, Delhi
2. Deitel: How to Program JAVA, PHI
3. U.K. Chakraborty and D.G. Dastidar: Software and Systems – An Introduction, Wheeler Publishing, Delhi.
4. Joseph O'Neil and Herb Schildt: Teach Yourself JAVA, TMH, Delhi.

Outcomes:-

At the end of this course, the student will be able to:

2. Develop websites using HTML and Javascript
3. Understanding of concepts of Java programming using inheritance, polymorphism.
4. Able to develop Java applets.
5. Understanding of the life cycle of servlets and do advanced java programming.

CSC-333: Ecommerce

Course Objective: This course focuses on Presents concepts and skills for the strategic use of e-commerce and related information technology from three perspectives: business to consumers, business-to-business, and intra-organizational. Examination of e-commerce in altering the structure of entire industries, and how it affects business processes including electronic transactions, supply chains, decision making and organizational performance.
In addition, some of the major issues associated with e-commerce—security, privacy, intellectual property rights, authentication, encryption, acceptable use policies, and legal liabilities. Students will build their own web presence and market it using an online platform.

UNIT-I: **Introduction to E-commerce**

Meaning and Concept; Objectives; Advantages and Disadvantages; Technical components and functions of e-commerce; E-Commerce and E-Business; Traditional Commerce vs. E-Commerce; Forces Driving E-Commerce; Growth of E-Commerce; E-Commerce Opportunities for Industries; Future of E-Commerce.

UNIT-II: **E-Commerce Model and Websites**

Forms of E-Commerce- Business to Consumer; Business to Business; Business to Government; Other Models – Brokerage Model, Aggregator Model, Community Model and Value Chain Model; Transaction Process: Basic concepts of EDI; Applications of EDI.

UNIT-III : **Electronic Payment System and E-Security & Privacy**

Special Features of Electronic Payment System; Types of E-Payment Systems-E-Cash, E-Cheque, Credit Card, Smart Card and Electronic Purses, Risk and E-Payment Systems; Secure Electronic Transaction (SET)

Security Risk of E-commerce; Types of Intruders; Types of Threats; Security Tools-Cryptography; Software packages for privacy; Security algorithms; Digital Signature and Firewalls.

Unit-IV Web Designing

Preparing Web pages and Website, Use of HTML and DHTML and scripting languages; Websites Generation- Concept and Meaning; Objectives and Advantages; Types of Websites; Website Designing Principles; Methods of Promoting Website; Searching the Website; Factors for Growth of Websites

References:

- William Stallings, Cryptography and Network security Principles and practice.
- Kalakota, Ravi and Whinston, Andrew B., Electronic Commerce – A Manager‘s Guide, Pearson Education, Inc
- Diwan, Prag and Sunil Sharma: Electronic Commerce – A manager’s Guide to E-Business, Vanity Books International, Delhi.
- Janal, D.S.: On-line Marketing Hand Book, Van Nostrand Reinhold, New York.
- Kosivr, David: Understanding Electronic Commerce, Microsoft Press, Washington.
- Robert W. Sebesta, Programming the World Wide Web, Sixth Edition, Published by Addison-Wesley. Copyright © 2011 by Pearson Education, Inc.
- Mlnoli and Minol: Web Commerce Technology Handbook, Tata McGraw Hill, New Delhi.
- Schneider Gary P: Electronic Commerce, Course Technology, Delhi.

Learning Outcome: At the end of this course, students should be able to

- Identify and apply relevant problem solving methodologies.
- Identify advantages and disadvantages of technology choices such as merchant server software and electronic payment options
- Discuss electronic commerce and the stakeholders and their capabilities and limitations in the strategic convergence of technology and business.
- Analyse features of existing e-commerce businesses, and propose future directions or innovations for specific businesses.

CSC-334: OPEN SOURCE OPERATING SYSTEM		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week		Lab: 1
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	Operating system and basics of Linux.

Course Objectives:

- To introduce the concept of Free and Open-Source Software.
- To understand the role of Kernel and shell.
- To demonstrate different commands in Linux.

Course Outcomes: The student will be able to

- Demonstrate the understanding of different *NIX operating systems.
- Work on Linux using different commands and develop shell scripts.

Course Content:

UNIT- 1	Introduction: open Source, Free Software, Free Software vs. Open Source software, Public Domain Software, FOSS does not mean any cost. History : BSD, The Free Software Foundation and the GNU Project.	**(7 Hours)**
UNIT- 2	Open Source Development Model Licences and Patents: What Is A License, Important FOSS Licenses (Apache, BSD, GPL, LGPL).	**(7 Hours)**
UNIT- 3	Open source operating system: Overview of Open Source operating systems, Introduction to Linux and Unix, Flavours of *NIX Operating systems.	**(7 Hours)**
UNIT- 4	Introduction to Linux: File system, Shell and Kernel,vi editor, shell variables, command line programming.	**(8 Hours)**
UNIT- 5	Filters and commands: Pr, head, tail, cut, paste, sort, uniq, tr, join, etc, grep, egrep, fgrepetc., sed,awk etc.,granting and revoking rights.Any other relevant topic.	**(8 Hours)**
UNIT- 6	Shell Programming: Types of Shells, Shell Meta Characters $#, $*, $?, Shell Variables, Shell Scripts. Debugging scripts, echo, read, operators, keywords, Integer Arithmetic and String Manipulation, Decision Making: if-else-elif-fi, case-esac. Loop Control; while, for, until, break & continue Functions, I/O Redirection and Piping, Exception Handling. Creating shell programs for automating tasks, file handling,trapping signals etc	**(8 Hours)**

Internal Assessment:

	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	

Text Books:

- Brian W. Kernighan, The Practice of Programming, Pearson Education.
- Bach Maurice J, Design of the Unix Operating system, PHI.
- Daniel P. Bovet, Understanding the Linux Kernel, Oreilly.

Reference Books:

- Linux Administration Handbook, 2nd Edition, Evi Nemeth, PearsonPublications, 2007
- Shell Scripting: Expert Recipes for Linux, Bash and more‖, Steve Parker, Wrox Publication, 2011
- Linux Command Line and Shell Scripting Bible, 3rd Edition, Richard Blum, Christine Breshnahan, Wiley Publications, 2015

CSC-335: Discrete Structures

Course Outline:
Discrete mathematics is the backbone of computer sciences. It provides the base for algorithm development and semantics of programming languages. This course is designed to given an introductory idea of different discrete structures, including graph theory. In particular, this course introduces logic, proofs, sets, relations, functions, counting, and abstract structure, with an emphasis on applications in computer science.

UNIT I: Sets: Definition and types, Set operations, Partition of set, Cardinality (Inclusion-Exclusion & Addition Principles), Recursive definition of set. Functions, Relations, Properties of binary relations, closure, Partial Ordering Relations, The Pigeonhole & Generalized Pigeonhole Principles, Composition of Functions Concept, Mathematical induction

UNIT II: Graph Theory: Graphs – Directed, Undirected, Simple,. Adjacency & Incidence, Degre of Vertex, Subgraph, Complete graph, Cycle & Wheel Graph, Bipartite & Complete Bipartite Graph, Weighed Graph, Union of Simple Graphs. Complete Graphs. Isomorphic Graphs, Path, Cycles & Circuits Euclerian & Hamiltonian Graphs.

UNIT III: Planar Graph: Kuratowski's Two Graphs, Euler's Formula, Kuratowski's Theorem.
Trees: Spanning trees- Kruskal's Algo, Finding Spanning Tree using Depth First Search, Breadth First Search, Complexity of Graph, Minimal Spanning Tree, Graph Coloring.

UNIT IV: Language of Logic: Proposition, Compound Proposition, Conjunction, Disjunction, Implication, Converse, Inverse & Contrpositive, Bi-conditional Statements, tautology, Contradiction & Contingency, Logical Equivalences, Quantifiers, Arguments Groups, Ring, fields and Lattice

Text/Reference Books

1. C.L Liu and D.P. Mohapatra, Elements of Discrete Mathematics: A Computer Oriented Approach, TMH, 3rd Edition
2. Rosen, Discrete Mathematics and its applications, 6th Edition
3. Schaum's Outlines of Discrete Mathematics, Seymour Lipschutz & Marc Lipson, 2nd Edition
4. Narsingh & Deo, Graph Theory with Applications to Engineering and Computer Science, PHI 2004 Publication

Learning Outcomes
This introductory course will allow students to learn the following:

- Fundamental mathematical concepts and terminology underlying a variety of discrete structures.
- Use of graph and its analysis for a variety of practical problems like shortest distance, and Binary search tree.
- Techniques of constructing mathematical proofs and use of propositional and predicate logic.

CSC-336: Information Security

Course Outline: - The designed course covers basic security aspects to protect network from various kind of attacks. Various kinds of encryption and decryption techniques to protect data from malicious users. Course covers DES and RSA logarithms which are basic symmetric and asymmetric algorithms. Course also covers various authentication mechanisms.

Objectives:

- To understand security fundamentals.
- To learn Cryptography Principles such as key exchange, digital signature and certificates.
- Demonstration of various encryption decryption schemes by taking plain text.
- Generation Message Authentication codes for different kinds of messages.

UNIT I: Introduction to security, attacks, computer criminals, security services.
Cryptography: Substitution ciphers, transposition cipher, confusion, diffusion, symmetric and asymmetric encryption. DES, odes of DES. Hash function, key exchange, digital signatures and certificates.

UNIT II: Public Key Cryptosystems: Principles of Public Key Cryptosystems, Factorization, RSA Algorithm, security analysis of RSA, Exponentiation in Modular Arithmetic. Key Management in Public Key Cryptosystems: Distribution of Public Keys, Distribution of Secret keys using Public Key Cryptosystems. Discrete Logarithms, Diffie-Hellman Key Exchange.

UNIT III: Message Authentication & Hashing: Birthday Paradox and General case of Duplications, Basic functions of Message Authentication and Hashing, Introduction to Hash & MAC algorithms.
Digital Signatures: RSA Based, El Gamal Signatures, Undeniable Signatures.
Authentication: Model of Authentication Systems, Impersonation, Substitution and spoofing games, Authentication schemes for mutual authentication based on shared secret, two-way public key, one-way public key, Mediated Authentication, One way Authentication

UNIT IV: Security in networks: Threats in networks, network security controls, firewalls, intrusion detection system Administrating security: Security planning, risk analysis, physical security, Ethical issues in security

Text/Reference Books

1. Stalling Williams: Cryptography and Network Security: Principles and Practices, 4th Edition, Pearson Education, 2006.
2. Kaufman Charlie et.al; Network Security: Private Communication in a Public World, 2nd Ed., PHI/Pearson.
3. Pieprzyk Josef and et.al; Fundamentals of Computer Security, Springer-Verlag, 2008.
4. Trappe & Washington, Introduction to Cryptography, 2nd Ed. Pearson.

Outcomes:-

1. Students will able to understand the security features to improve the reliability in communication system.
2. Able to find the encrypt message by using various encryption mechanisms.
3. Various message authentication codes and message digest algorithms to improve the security aspects in data transmission.

CSC-337: Distributed Systems		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
Practical:	Term Work & OR:	Term Work & OR:
		Total: 4

Course Pre-requisites: The students should have knowledge of Operating System and Computer Networking

1	Basic concepts of Distributed System

Course Objectives:

	To develop the knowledge of Distributed Systems and how they are utilized in contemporary computing

Course Outcomes: The student will be able to

1	Elaborate the types of Distributed System and their components
2	Understand the underlying architecture of Distributed System
3	Distinguish between different distributed file system
4	Understand and apply the Distributed clock synchronization using various algorithms
5	Understand the concept of Concurrency Control , Recovery, and security mechanism in Distributed environment
6	Study the case study of Distributed object based systems

Course Content:

UNIT- 1	**Introduction to Distributed System:** Concepts, Architectures, transparency Self-management in Distributed system ,Thread, Virtualization, Client, server, code migration Semantics, Remote Procedure Calls, Communication, Naming,	**(10 Hours)**
UNIT- 2	Monolithic kernel, layered systems, virtual machines. Process based models and client server model. The micro-kernel based client-server approach. Inter-process communication and Remote Procedure Call. Tasks and Threads. Examples from LINUX, Solaris 2 and Windows NT.	**(10 Hours)**
UNIT- 3	**File System:** Flat naming, Structure naming and Attribute based naming, Security, Concurrency control and recovery, local area network, distributed languages and communication primitives, case studies of distributed systems.	**(10 Hours)**
UNIT- 4	**Distributed Synchronization:** Clocks and Election algorithm, Consistency Model, Consistency Protocol, Resilience, Reliable communication, Distributed Commit, recovery in Distributed systems. Security issues in distributed systems. Deadlock in distributed systems.	**(10 Hours)**
UNIT- 5	Distributed Operating Systems, Distributed File System, Sun NFS, and the Coda files system. NTFS, UNIX ext2 and ext3.	**(10 Hours)**
UNIT- 6	**Case studies** of Distributed object based systems (CORBA), Distributed web based Systems.	**(10 Hours)**

Internal Assessment:

Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4, 5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s from the list below	
	1. Types of Distributed System and their components	
	2. Understanding the underlying architecture of Distributed System	
	3. To undergo the case studies of distributed systems	

	4. Distributed Transaction Management and Distributed Deadlock Managing	
	5. Case study of Disturbed File System	
Text Books: Tanenbaum, A. S. and Van Steen, M. "Distributed Systems Principles and" (ISBN 0-13-088893-1), Prentice Hall 2002.		
Reference Books:		
P. K. Sinha, "Distributed Operating Systems," PHI. Paradigms.		
Bacon, J., "Concurrent Systems", 2nd Edition, (ISBN 0-201-177-676), Addison Wesley 1998.		
Silberschatz, A., Galvin, P. and Gagne, G., "Applied Operating Systems Concepts", 1st Edition," (ISBN 0-471-36508-4), Wiley 2000.		
Coulouris, G. et al, "Distributed Systems: Concepts and Design, 3rd Edition," (ISBN 0-201-61918-0), Addison Wesley 2001		

CSC-338: Programming in Java		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week		Lab: 1
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	Basic knowledge of C and C++ programming.

Course Objectives:

- To introduce concepts of core java systematically.
- To demonstrate the principles underlying the design of high-level programming languages.

Course Outcomes:

- The students will have competence in the use of the Java Programming language.
- The development of small to medium-sized application programs that demonstrate professionally acceptable coding.

Course Content:

UNIT- 1	Fundamentals of object-oriented programming, Java evolution, Features of Java, Java Development Kit (JDK), Java basics, keywords, constants, variables, and data types.	**(7 Hours)**
UNIT- 2	Operators and expressions, compiling and executing java Programs, operators, control flow Statements, Switch Statement, Looping statements, break and continue.	**(7 Hours)**
UNIT- 3	Classes, objects, and methods, declaring methods in java, constructors, static members, nesting of methods, inheritance, abstract methods and classes, array, String, and vectors.	**(8 Hours)**
UNIT- 4	Interfaces, defining interfaces, extending interfaces, implementing interfaces, Packages, creating packages, managing errors, and exception. Multithreaded programming.	**(7 Hours)**
UNIT- 5	Applets programming, Applet class, Applet and HTML, the Life cycle of an applet.	**(8 Hours)**
UNIT- 6	Graphics programming using AWT, colors, font, Event handling, Components of an event, event classes, event listener.	**(8 Hours)**

Internal Assessment:

	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	

Text Books:

- E. Balagurusamy, Programming with JAVA, a Primer, 4th Edition, 2010, Tata McGraw-Hill Publishing Company Limited, Delhi.
- Dr.Rao, Nageswara. Core Java: An Integrated Approach, New Edition Kongent Solutions Inc, 2009.

Reference Books:

- Schildt Herbert, Java: The Complete Reference, 8th Edition, Tata McGraw- Hill,2011.

CSC-339: Foundation of Vedic Mathematics

About the subject: The course is mainly based upon the book Lilavati, originally, authored by the mathematician Bhaskaracarya in vedic perid. The Lilavati is a book on arithmetic written in the twelfth century. It has been used as a textbook for 800 years in India.

Objective: The objective of the course is to introduce the methods used for arithmetic in vedic period. After studying this course the students will be able to use vedic methods in arithmetic which are easy to use and/ or whose computer algorithms are of less complexity.

Pre-requisite: None

Credit: 3

UNIT I

Brief introdction to Bhascarararyacarya and his works [02 Lectures]

Units of measurement. Indo-Arabic numericals, Place value system, Arithmetic operations of addition, subtarction, multiplication and division, Methods of finding squares, Square root, Methods to find cube, Cube roots. [12 Lectures]

Fractions - Operations, Addition and subtraction, Multiplication, Division, Squares, Cubes, Square roots, Cube roots. [5 Lectures]

UNIT II

Rules concerning zero, Reverse process, To find an unknown quantity, Method of transition, square transition. Linear and quadratic equations, The rule of three, Inverse proportion, The rule of five, Rules for Barter, Simple interest, Combinations. [13 Lecture]

UNIT III

Progression: Arithmetic and geometric progressions, and series, Mensuration, [15 Lectures]

References

1. Krishnaji Shankara Patwardhan, Somashekhara Amrita Naimpally and Shyam Lal Singh, Lilavati of Bhaskaracarya: A Treatise of Mathematics of Vedic Tradition, Motilal Banarsidaas Pub. Pvt. Ltd., Delhi, 2021, 2001.
2. Bhakaracarya, Leelavati, Srivenkateshwar Steem Press, Bombay, 1979.
3. A B Padmnabha Rao, Bhascaracarya's Leelavati, Chinmay International Foundation Shodha Sansthan Adi Sankara Nilyam, Veliyanad, 2014.
4. K Ram Subramaniam, Ganitanand, Indian Society of History of Mathematics, 2015.
5. N. H. Fadke, लीलावती पुनर्दर्शन, Sarvahak Prakashan, Pune, 1971.
6. Pandit Ganpatideva Shastri, गणितकौमुदि, Chaukhambha SansKrit Series, Varansi, 1969.

CSC-431: WEB TECHNOLOGY		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week		Lab: 1
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	• Basics of Internet • Operating System.

Course Objectives:

- This Subject is useful for Making own Web page and how to host own web site on internet.

Course Outcomes:

- After Studying that subject students would have capability to make own web site and host their own web site on internet. Also students would have enough knowledge about what are the technologies used in internet.

Course Content:

UNIT- 1	Introduction to WWW : Protocols and programs, secure connections, application and development tools, the web browser, What is server, choices, setting up UNIX and Linux web servers, Logging users, dynamic IP Web Design: Web site design principles, planning the site and navigation	**(7 Hours)**
UNIT- 2	Introduction to HTML : The development process, Html tags and simple HTML forms, web site structure Introduction to XHTML : XML, Move to XHTML, Meta tags, Character entities, frames and frame sets, inside browser.	**(7 Hours)**
UNIT- 3	Style sheets : Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2	**(7 Hours)**
UNIT- 4	Javascript : Client side scripting, What is Javascript, How to develop Javascript, simple Javascript, variables, functions, conditions, loops and repetition	**(8 Hours)**
UNIT- 5	DHTML : Combining HTML, CSS and Javascript, events and buttons, controlling your browser, Ajax: Introduction, advantages & disadvantages ,Purpose of it ,ajax based web application, alternatives of ajax	**(8 Hours)**
UNIT- 6	PHP : Starting to script on server side, Arrays, function and forms, advance PHP Databases : Basic command with PHP examples,	**(8 Hours)**

	Connection to server, creating database, selecting a database, listing database, listing table names creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP myadmin and database bugs	
Internal Assessment:		
	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	
Text Books:		
• Burdman, “Collaborative Web Development”, Addison Wesley. • Sharma &Sharma, “Developing E-Commerce Sites”, Addison Wesley • Ivan Bayross, “Web Technologies Part II”, BPB Publications.		
Reference Books:		
• Steven Holzner,”HTML Black Book”, Dremtech press. • Web Technologies, Black Book, Dreamtech Press • Web Applications : Concepts and Real World Design, Knuckles, Wiley-India • Internet and World Wide Web How to program, P.J. Deitel & H.M. Deitel Pearson.		

CSC-432: CLOUD COMPUTING		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
		Lab:
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	• Computer Networks. • Operating System.

Course Objectives:

- Demonstrate the various Distributed technologies to perform the complex task in highly distributed environment.
- Demonstrate the service oriented architecture to provide on-demand services to Internet users.
- Design service level agreements (SLA) to meet the guaranty services in Cloud Environment.
- Design Energy efficient Scheduling techniques to balance the Workload in a distributed environment.
- Design Energy Efficient model for sustainable cloud platform for next decade various novel service integration paradigm.

Course Outcomes:

- Student will able to understand basic concepts required to develop cloud computing applications.
- Student will able to develop applications for cloud computing to provide on-demand services required for users.
- Student will able to understand the service oriented architecture such as IaaS, PaaS and SaaS.
- Student will able to design and implement a novel cloud computing application in simulation environment.
- Student will able to do comparative study and analysis of different economic cloud computing models with existing conventional software developing methodologies.

Course Content:

UNIT- 1	Introduction Introduction to Cloud Computing, Roots of Cloud Computing: Fundamental concepts of Distributed Systems, Cluster Computing, Grid Computing, and Mobile Computing.	**(10 Hours)**
UNIT- 2	Cloud Models Basics of Cloud Computing Concepts, Characteristics of Cloud Computing, Need for Cloud, Cloud Deployment models: private, public, hybrid and community cloud, Cloud Services: Resource-as-a-Service (RaaS), Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS), Examples of each service.	**(10 Hours)**
UNIT- 3	Cloud Services RaaS: Usage of Physical resources like servers, networks, data center etc, IaaS: Virtualization,. PaaS: Integrated lifecycle platform: Google App Engine, Microsoft Azure, Anchored life cycle platform: Salesforce platform, SaaS: Characterizing SaaS, Salesforce's software environment.	**(10 Hours)**

UNIT- 4	Resource Scheduling for Cloud Computing: - Introduction, Virtual Machine provisioning and Migration Services, Scheduling techniques of Virtual machines for resource reservation, Cloud Service Scheduling hierarchy	**(10 Hours)**
UNIT- 5	Economic models for Resource-allocation scheduling , Heuristic Models for task –execution scheduling : Static Strategies , Dynamic Strategies , Heuristic Schedulers.	**(10 Hours)**
UNIT- 6	Cloud Applications Cloud Applications, Cloud challenges, Cloud Security and privacy issues, Mobile Cloud Computing, Integration of Cloud with Wireless Sensor Network and its application.	**(10 Hours)**
Internal Assessment:		
	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	

Text Books:

- Cloud Computing Bible by Barrie Sosinsky, Wiley Publication, 2011.
- Cloud Computing: A Practical Approach by Anthony T. Velte Toby J. Velte, Robert Elsenpeter, The McGraw-Hill Publication, 2010.
- Cloud Computing: Concepts, Technology and Architecture by Thomas Erl, Zaigham Mahmood, and Ricardo Puttini, 1st Edition, Prentice Hall.
- Cloud Computing: Data-Intensive Computing and Scheduling by Frederic Magoules , Jie Pan, and Fei Teng. CRC Press. Taylors & Francis Group.

Reference Books:

- Cloud Computing for Dummies, Judith Hurwitz, Robin Bloor, Marcia Kaufman and Fern Halper, Wiley Publication.
- New frontiers in information and software as a service, Divyakant Agrawal, K. SelcukCandan, WenSyan Li (Eds.), Springer Proceedings.
- Cloud Computing Theory and Practice Danc. Marinercus, Elsevier, 2013.

CSC-433: PARALLEL PROCESSING

Objective: After completion of this course students will be able to understand architectural design that provides the parallel computational power to the computer.

Unit-I: Pipeline and Vector Processing: Nonlinear and linear pipelining, Multiprocessor, Multicomputer, Super computer. Array Processors. Scope and Application of Parallel approach.

Unit-II: Paradigms of parallel computing: SIMD, Systolic; Asynchronous - MIMD, reduction paradigm. Hardware taxonomy: Flynn's classifications, Handler's classifications. PRAM model and its variants: EREW, ERCW, CRCW, PRAM algorithms, Sorting network, Interconnection RAMs. Parallelism approaches - data parallelism, control parallelism.

Unit-III: Parallel Processors: Taxonomy and topology - shared memory mutliprocessors, distributed memory networks.

Unit-IV: Processor organization - Static and dynamic interconnections. Embeddings and simulations.

Unit-V: Performance Metrices: Laws governing performance measurements. Metrices - speedup, efficiency, utilization, cost, communication overheads, single/multiple program performances, bench marks.

Unit-VI: Scheduling and Parallelization: Scheduling parallel programs. Loop scheduling. Parallelization of sequential programs. Parallel programming support environments.

BOOKS:

1. M. J. Quinn.Parallel Computing: Theory and Practice, McGraw Hill, New York, 1994.
2. T. G. Lewis and H. El-Rewini. Introduction to Parallel Computing, Prentice Hall, New Jersey, 1992.
3. T. G. Lewis.Parallel Programming: A Machine-Independent Approach, IEEE Computer Society Press, Los Alamitos.
4. Sima and Fountain, Advanced Computer Architectures, Pearson Education.
5. Mehdi R. Zargham, Computer Architectures single and parallel systems, PHI.
6. Ghosh, Moona and Gupta, Foundations of parallel processing, Narosa publishing.
7. Ed. Afonso Ferreira and Jose' D. P. Rolin, Parallel Algorithms for irregular problems - State of the art, Kluwer Academic Publishers.
8. Selim G. Akl, The Design and Analysis of Parallel Algorithms, PH International.

Learning Outcomes:

At the end of this course, the student will be able to:

- Understand uniprocessor computer architecture
- Understand the computer architecture (i.e., pipelining and superscalar processor design and memory hierarchy)
- Understand parallel hardware and parallel software
- Understand shares-memory management
- Understand distributed-memory with MPI
- Understand general-purpose GPU

CSC-434: Adhoc & Wireless Network		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
Practical:	Term Work & OR:	Term Work & OR:
		Total: 4

Course Pre-requisites: The students should have knowledge of Computer Network

1	Basic concepts of Computer Network

Course Objectives:

	To develop the knowledge of Adhoc and Wireless Sensor Network To understand the basics of Ad-hoc & Sensor Networks. To learn various fundamental and emerging protocols of all layers. To study about the issues pertaining to major obstacles in establishment and efficient management of Ad-hoc and sensor networks. To understand various security practices and protocols of Ad-hoc and Sensor Networks.

Course Outcomes: The student will be able to

1	Identify different issues in wireless ad hoc and sensor networks.
2	To analyze protocols developed for ad hoc and sensor networks.
3	To identify and address the security threats in ad hoc and sensor networks.
4	Establish a Sensor network environment for different type of applications.

Course Content:

UNIT- 1	Introduction: What is an Ad Hoc Network?, Types of Ad hoc Mobile Communications , Types of Mobile Host Movements, Challenges Facing Ad hoc Mobile Networks, Ad hoc wireless Internet, Issues in Designing a Routing Protocol for Ad Hoc Wireless Networks, Classifications of Routing Protocols: Table–Driven Routing Protocols, Destination Sequenced Distance Vector (DSDV), Wireless Routing Protocol (WRP), Cluster Switch Gateway Routing (CSGR), Source–Initiated On–Demand Approaches, Ad hoc On–Demand Distance Vector Routing (AODV), Dynamic Source Routing (DSR), Temporally Ordered Routing Algorithm (TORA), Signal Stability Routing (SSR), Location–Aided Routing (LAR), Power–Aware Routing (PAR), Zone Routing Protocol (ZRP).	**(10 Hours)**
UNIT- 2	Wireless Sensor Networks: Introduction to Wireless sensor networks, Single-sink single-hop WSN, Single-sink multi-hop WSN, Multi-sink multi-hop WSN, Advantages of ad-hoc/sensor networks, Node and Network Architectures,	**(10 Hours)**

	Wireless Sensor Device Architecture, Network Architectures, Main features of WSANs, Current and future research on WSANs	
UNIT- 3	Applications of WSNs: Positioning and animals tracking, Entertainment, Logistics, Transportation, Industrial Control and Monitoring, Home Automation and Consumer Electronics, Security and Military Sensing, Asset Tracking and Supply Chain Management, Intelligent Agriculture and Environmental monitoring, Health Monitoring.	**(10 Hours)**
UNIT- 4	Technologies for WSNs: ZigBee technology, Ultrawide bandwidth technology, Bluetooth technology, Comparison among technologies	**(10 Hours)**
UNIT- 5	The Physical Layer: Introduction, Wireless Propagation Models: The Free Space Propagation Model, The Two-Ray Ground Model, The Log-Distance Path Model, Energy Dissipation Model, Error Models: The Independent Error Model, The Two-State Markov Error Model, Sensing Models: The Binary Sensing Model, The Probabilistic Sensing Model	**(10 Hours)**
UNIT- 6	Communication protocols for WSNs MAC protocols: Scheduled protocols, LEACH protocol, Guo protocol, TRAMA protocol, Contention-based protocols, Zhong protocol, DMAC protocol, PAMAS protocol, SMAC protocol Routing protocols: Issues in designing routing protocols, Classification of routing protocols, Flat routing, Flooding and gossiping, SPIN protocol, Directed diffusion protocol, Rumour routing, Gradient-based routing, Hierarchical routing, LEACH protocol, PEGASIS protocol, TEEN protocol, MECN protocol, SPAN protocol, Location-based routing protocols, GAF protocol, GEAR protocol, GeRaF protocol, Rugin protocol	**(10 Hours)**
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4, 5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s	

Text Books:

1) Roberto Verdone, Davide Dardari, Gianluca Mazzini and Andrea Conti, “Wireless Sensor and Actuator Networks: Technologies, Analysis and Design”, Academic Press, 2008.
2) Miguel A. Labrador and Pedro M. Wightman, “Topology Control in Wireless Sensor Networks-with a companion simulation tool for teaching and research”, Springer Science, 2009.
3) Azzedine Boukerche, “Handbook of Algorithms for Wireless Networking and Mobile Computing”, Chapman & Hall/CRC, 2005.

CSC-435: High Performance Computing		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
Practical:		
		Total: 4

Course Pre-requisites: The students should have knowledge of

	Computer Networks. Operating Systems.

Course Objectives:

- Demonstrate the scientific application execution methodology in Highly Distributed Computing Environment.
- To study Processor Architecture and Memory Hierarchies which support for HPC.
- To learn Programming strategies for parallel computing to solve highly complex scientific problems.
- To understand the parallel computer concepts different types of parallel architecture, hard ware design and compilers principles.
- Illustration of well-known mathematical examples to understand the basic concepts of parallel computation which are highly required to solve scientific applications.
- Detail study of various kinds of mathematical examples where parallel computations are involved, for example in linear algebra solving system of equations and matrix decomposition, Fourier transforms.

Course Outcomes: The student will be able to

- Able to understand the difference between sequential architecture and parallel architecture to execute the scientific applications.
- Understand the way to develop parallel algorithm and way of execution on parallel computing environment.
- Analysis of time and space complexity for a particular mathematical problem in sequential as well as parallel.
- Writing programs for to solve Partial differential equations (PDE) and Matrix decomposition.
- Solve some Computational biology applications using Dynamic programming approaches.

Course Content:

UNIT- 1	Single-processor Computing, The Von Neumann architecture, Modern processors, Memory Hierarchies, Multi core architectures, Locality and data reuse, Programming strategies for high performance, Power consumption, Review questions.	**(10 Hours)**
UNIT- 2	Parallel Computing, Introduction, Quantifying parallelism, Parallel Computers Architectures, Different types of memory access, Granularity of parallelism, Parallel programming, Topologies, Multi-threaded architectures ,Co-	**(10 Hours)**

	processors, Remaining topics, Computer Arithmetic, Integers, Real numbers, Round-off error analysis, Compilers and round-off, More about floating point arithmetic, Conclusions.	
UNIT- 3	Numerical treatment of differential equations, Initial value problems, Boundary value problems, Initial boundary value problem, Numerical linear algebra, Elimination of unknowns, Linear algebra in computer arithmetic, LU factorization, Sparse matrices, Iterative methods, Further Reading.	**(10 Hours)**
UNIT- 4	High performance linear algebra, Collective operations, Parallel dense matrix-vector product, LU factorization in parallel, Matrix-matrix product, Sparse matrix-vector product, Parallelism in solving linear systems from Partial Differential Equations (PDEs), Computational aspects of iterative methods , Parallel preconditions ,Ordering strategies and parallelism, Operator splitting, Parallelism and implicit operations ,Grid updates ,Block algorithms on multi core architectures.	**(10 Hours)**
UNIT- 5	Applications, Molecular dynamics, Force Computation, Parallel Decompositions, Parallel Fast Fourier Transform, Integration for Molecular Dynamics, Sorting, Brief introduction to sorting Odd-even transposition sort, Quicksort, Bitonic sort, Graph analytics, Traditional graph algorithms, Real world' graphs, Hypertext algorithms, Large-scale computational graph theory,.	**(10 Hours)**
UNIT- 6	N-body problems, The Barnes-Hut algorithm, The Fast Multipole Method, Full computation, Implementation Monte Carlo Methods, Parallel Random Number Generation, Examples, Computational biology Dynamic programming approaches, Suffix tree.	**(10 Hours)**
Internal Assessment:		
	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	
Text Books:		
• Introduction to High Performance Scientific Computing Evolving Copy - open for comments Victor Eijkhout, Edmond Chow, Robert van de Geijn. • High Performance Computing (RISC Architectures, Optimization & Benchmarks), Charles Severance, Kevin Dowd, Oreilly.		
Reference Books:		
• High Performance Computing (RISC Architectures, Optimization & Benchmarks), Georg Hager, Gerhard Wellein, CRC Press. • Introduction to High-Performance Scientific Computing (Scientific and Engineering Computation), Lloyd D. Fosdick, Elizabeth R. Jessup		

CSC-436: INTERNET OF THINGS (IoT)

Course Outline:

Internet of Things (IoT) is presently an emerging technology worldwide. Government, academia, and industry are involved in different aspects of research, implementation, and business with IoT. IoT cuts across different application domain verticals ranging from civilian to defense sectors. These domains include agriculture, space, healthcare, manufacturing, construction, water, and mining, which are presently transitioning their legacy infrastructure to support IoT. Today it is possible to envision pervasive connectivity, storage, and computation, which, in turn, gives rise to building different IoT solutions. IoT-based applications such as innovative shopping system, infrastructure management in both urban and rural areas, remote health monitoring and emergency notification systems, and transportation systems, are gradually relying on IoT based systems.

Course Objectives

- To teach state of art of wireless sensor networks
- To discuss importance of communication protocols.
- To teach challenges in routing protocols and overview of protocols across different layers.
- To teach basics of Internet of Things.

Unit I: Introduction: Overview of Wireless Sensor Networks – Characteristics, Applications, Design objectives, challenges. Different types of sensors and applications of wireless sensor networks.

Unit II: Medium Access Control protocols for Wireless sensor networks: Functions of MAC layer, Fundamental MAC protocols, Objectives of MAC protocols, Energy efficiency in MAC design, Fixed assignment protocols, demand assignment protocols.

Unit III: Network and Transport Layer protocols for wireless sensor networks: Fundamentals and Challenges of Routing protocol, routing strategies in wireless sensor networks. Traditional transport protocols, Transport protocols for sensor networks.

Unit IV: Basics on Internet of Things: Introduction, Components of IoT, IoT communication technologies and protocols, developing basic IoT applications.

Unit V: Physical and Data link layer protocols for IoT like Zigbee and Z-Wave. Network layer protocols for IoT like RPL.

Unit VI: Transport layer protocols for IoT. Application layer protocols for IoT like MQTT. Emerging technologies in IoT.

Text/References:

1. Jun Zheng, Abbas, " Wireless sensor networks A networking perspective", WILEY, 2009.
2. Kazem Sohraby, Daniel Minoli, & Taieb Znati, ―Wireless Sensor Networks-Technology, Protocols, And Applications‖, John Wiley, 2007
3. Thomas Haenselmann, ―Wireless Sensor Networks: Design Principles for Scattered Systems‖, Oldenbourg Verlag, 2011
4. Vijay Madisetti and Arshdeep Bahga, "Internet of Things (A Hands-on-Approach)", 1st 9 Edition, VPT, 2014.
5. E. H. Callaway, Jr. E. H. Callaway, Wireless Sensor Networks Architecture and Protocols: CRC Press
6. F. Zhao and L. Guibas, Wireless Sensor Network: Information Processing Approach, Elsevier.
7. A. Hac, Wireless Sensor Network Designs, John Wiley & Sons 8. Francis daCosta, "Rethinking the Internet of Things: A Scalable Approach to Connecting Everything", 1st Edition, Apress Publications, 2013

Learning Outcomes

- At the end of the course students can be able to:
- Understand technological background of sensor networks.
- Able to design applications using Raspberry Pi.
- Design and apply various existing routing protocols of sensor networks.
- Design the architecture and reference model of IoT.

CSC-437: ADBMS		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
Practical:	Term Work & OR:	Term Work & OR:
		Total: 4

Course Pre-requisites: The students should have knowledge of Database System and Computer Networking

1	Basic concepts of Distributed Database Systems , Parallel Database Systems, Distributed Object Database Management systems

Course Objectives:

	To develop the understanding of various Advanced Database Systems and how they are utilized in present era of computing.

Course Outcomes: The student will be able to

1	Enrich their the knowledge in the area of various Advanced Database Systems
2	Understand the Distributed query processing.
3	Understand Distributed Transaction management and Distributed concurrency control.
4	Get the idea of Parallel Database Systems and Distributed Object Database Management systems and their architecture
5	Understand the concept of Concurrency Control , Recovery, and security mechanism in Distributed environment
6	Undergo the case study of Distributed Database System

Course Content:

UNIT- 1	**Introduction:** Distributed Data processing, Distributed Database Systems (DDBMSs), Promises of DDBMSs, Complicating factors and Problem areas in DDBMSs. Overview of relational database system. **Distributed DBMS Architecture:** DBMS Standardization, Architectural models for Distributed DBMS, Distributed DBMS Architecture.	**(10 Hours)**
UNIT- 2	**Distributed Database Design:** Alternative design Strategies, Distribution design issues, Fragmentation, Allocation. **Semantic Data Control:** View Management, Data security, Semantic Integrity Control.	**(10 Hours)**
UNIT- 3	**Overview of Query Processing:** Query processing problem, Objectives of Query Processing, Complexity of Relational Algebra operations, characterization of Query processors, Layers of Query Processing. **Introduction to Distributed Transaction Management:** Definition of Transaction, Properties of transaction, types of transaction.	**(10 Hours)**

UNIT- 4	**Distributed Concurrency Control:** Serializability theory, Taxonomy of concurrency control mechanisms, locking based concurrency control algorithms.	**(10 Hours)**
UNIT- 5	**Parallel Database Systems:** Database servers, Parallel architecture, Parallel DBMS techniques, Parallel execution problems, Parallel execution for hierarchical architecture. **Database Interoperability:** Database Integration, Query processing	**(10 Hours)**
UNIT- 6	**Distributed Object Database Management systems:** Fundamental Object concepts and Object models, Object distribution design. Architectural issues, Object management, Distributed object storage, Object query processing. Transaction management. **Introduction to other Advanced Database Systems**: Multimedia Databases, Spatial Databases, Deductive Databases, etc.	**(10 Hours)**
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4, 5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s from the list below	
	1. Distributed Query Processing	
	2. Distributed Transaction Management	
	3. Distributed Deadlock Management	
	4. Case study on DDBMS	
Term Work:		
Part – A		
Text Books: Principles of Distributed Database Systems, M.TamerOzsu, Patrick Valduriez, 2nd Edition, 1999.		
Reference Books:		
Distributed Databases principles and systems, Stefano Ceri, Giuseppe Pelagatti, TMH, 2008.		
Database System Concepts, 7th Edition, Abraham Silberschatz, Henry F. Korth, S. Sudarshan		
Fundamentals of Database Systems, Pearson, Elmasri Ramez, Navathe Shamkant		
Topics for Project based learning		

CSC-438: Software Project Management		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
Practical:	Term Work & OR:	Term Work & OR:
		Total: 4

Course Pre-requisites: The students should have knowledge of Software Engineering	
1	Basic concepts of Software Engineering
Course Objectives:	
	To develop the knowledge of Distributed Systems and how they are utilized in contemporary computing
Course Outcomes: The student will be able to	
1	Learn Conventional Software Management and Evolution of Software Economics
2	Understand Project Organizations and Responsibilities
3	Understand the evolution and applications of operations in various fields, mathematically formulate linear programming problems and solve them using different techniques
4	Construct a project network and apply program evaluation review technique and critical path method to find date of completion of project and other project related metrics

Course Content:

UNIT- 1	**Introduction** Project Management (PM) Fundamentals, People, Process, and Product, Technology Classic mistakes, PMI Processes, Software project phases, Organizational structures, Project charter Statement of Work (SOW)	**(10 Hours)**
UNIT- 2	**Planning Phase** Development lifecycle models, Matching lifecycles to projects, Project plans Work Breakdown Structures (WBS)	**(10 Hours)**
UNIT- 3	**Estimation and Budgeting** Estimation, Budgeting, Project selection, NPV, ROI, Payback models, Scheduling: Project network diagram fundamentals, PERT techniques, Gantt charts, Critical chain scheduling	**(10 Hours)**
UNIT- 4	**Risk and Change Management** Risk management, Change control, More MS-Project	**(10 Hours)**
UNIT- 5	**Development Management** Team models, Requirements process, Configuration management, Software metrics, Programming languages & tools, managing conflict and motivating, MS-Project: Assigning Resources	**(10 Hours)**
UNIT- 6	**System Test Process** Test specifications, Black box and white box testing, Test scripts, Unit and integration testing, Acceptance test specifications, Test tools, Final Phases & Other Issues: Project Recovery, Documentation,	**(10 Hours)**

	Cutover/Migration, Post Project Reviews, Closing	
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4, 5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s	
Text Books: Kathy Schwalbe, “Information Technology Project Management”, Cengage Learning, 7/e, 2013.		
Reference Books:		
M. Cottrell and B. Hughes, "Software Project Management", McGraw-Hill, 5/e, 2009.		
QuantumPM, “Microsoft Office Project Server 2003 Unleashed”, Pearson Education India, 2005.		
Robert T. Futrell, Donald F. Shafer and Linda Isabell Shafer , “Quality Software Project”, Pearson India, 2002.		
D. J. Henry, “Software Project Management – A Real-World Guide to Success”, Addison-Wesley, 2003.		

CSC-439: COMPUTING & VEDIC MATHEMATICS

About the subject: The course is mainly based upon the book Lilavati, originally, authored by the mathematician Bhaskaracarya in vedic perid. The Lilavati is a book on arithmetic written in the twelfth century. It has been used as a textbook for 800 years in India.

Objective: The objective of the course is to introduce the methods used for arithmetic in vedic period. After studying this course the students will be able to use vedic methods in arithmetic which are easy to use and/ or whose computer algorithms are of less complexity.

Pre-requisite: None

Credit: 4

UNIT I : Brief introdction to Bhascarararyacarya and his works [02 Lectures]

Units of measurement. Indo-Arabic numericals, Place value system, Arithmetic operations of addition, subtarction, multiplication and division, Methods of finding squares, Square root, Methods to find cube, Cube roots. [12 Lectures]

Fractions - Operations, Addition and subtraction, Multiplication, Division, Squares, Cubes, Square roots, Cube roots. [5 Lectures]

UNIT II: Rules concerning zero, Reverse process, To find an unknown quantity, Method of transition, square transition. Linear and quadratic equations, The rule of three, Inverse proportion, The rule of five, Rules for Barter, Simple interest, Combinations. [13 Lecture]

UNIT III

Progression: Arithmetic and geometric progressions, and series, Mensuration, [15 Lectures]

UNIT IV: Volume, Wood Cutting, Volume of a heap of grain, Shadows, Pulverization, Concatenation (Permnutations, Partitions etc.) [7 Lecture]

UNIT V: Pingal's binary number system, Different types of Meru Prastar (including Pascal triangle). [8 Lectures]

UNIT VI: Computer programmming based on methods given in Unit I - IV and comparing complexity with the respective modern methods. Square root as numerical approach as prescribed in Sulbasutras. [10–15 Lab sessions]

References

1. Krishnaji Shankara Patwardhan, Somashekhara Amrita Naimpally and Shyam Lal Singh, Lilavati of Bhaskaracarya: A Treatise of Mathematics of Vedic Tradition, Motilal Banarsidaas Pub. Pvt. Ltd., Delhi, 2021, 2001.
2. Kapil Dev Dwivedi, Shyam Lal Singh, The Prosody of Pingala: A Treatise of Vedic and Sanskrit Metrics with applications of Vedic Mathematics (with Hindi and English Translation), Vishwavidyalaya Prakashan, Varanasi, 2008.
3. Bibhutibhusan Datta, The Science of The Sulba, University of Calcutta, 1991.
4. Bhakaracarya, Leelavati, Srivenkateshwar Steem Press, Bombay, 1979.
5. Pt Kedar Nath, छन्द: शास्त्रम, Chaukhambha Publisher, Varanasi, 2002.
6. A B Padmnabha Rao, Bhascaracarya's Leelavati, Chinmay International Foundation Shodha Sansthan Adi Sankara Nilyam, Veliyanad, 2014.
7. K Ram Subramaniam, Ganitanand, Indian Society of History of Mathematics, 2015.
8. Pandit Yudhishtir Mimansa, वैदिक-छन्दोमीमांस, Ramlal Kapoor Trust, Haryana, 2006.
9. N. H. Fadke, लीलावती पुनर्दर्शन, Sarvahak Prakashan, Pune, 1971.
10. Sisheel Trivedi, छंद का आधुनिक रचना विधान, Rashtra Prakashan, Delhi, 1880.
11. Pandit Ganpatideva Shastri, गणितकौमुदि, Chaukhambha SansKrit Series, Varansi, 1969.

The first book covers syllabus from Unit I – III of the course. Second book covers last half of the Unit III. Third book covers some part of Unit IV

CSC-531: Data Science Algorithms		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week	Term Work & OR:	Term Work & OR: 1
		Total: 4

Course Pre-requisites: The students should have basic knowledge of Database System, Statistics and Machine Learning.

1	Basic concepts of Data Science, Data Analytics, and Data Science Algorithms, Applications of Data Science Algorithms.

Course Objectives:

	To develop the basic understanding of various Data Science Algorithms, Applications of Data Science Algorithms.

Course Outcomes: The student will be able to

1	Enrich their knowledge in the area of Data Science and its allied areas.
2	Understand the various Data Science Algorithms
3	Understand the applications of various Data Science Algorithms
4	Case study based on any one of the Data Science Algorithm

Course Content:

UNIT- 1	**Introduction Data Science:** The Art of Data Science, Volume, Velocity, Variety, Machine Learning, Supervised and Unsupervised Learning, Predictions and Forecasts, Innovation and Experimentation , The Dark Side-Big Errors, Privacy. Theories, Models, Intuition, Causality, Prediction, Correlation	**(10 Hours)**
UNIT- 2	**Classification**: K-nearest neighbors (K-NNs), Text classification using K-nearest neighbors, Naïve Bayes Theorem and Extended Naïve Bayes Theorem for Classification**.**	**(10 Hours)**
UNIT- 3	**Decision Trees (DTs):** Information theory, information entropy and information gain. Decision tree construction using ID3 algorithm and classification using DTs. Random Forest-construction and classification.	**(10 Hours)**
UNIT- 4	**Clustering:** Types of clustering, clustering using AGNES and DIANA, clustering using k-means, k-means vs. k-mediods. Density based clustering using DBSCAN. Clustering documents using clustering.	**(10 Hours)**
UNIT- 5	**Regression Analysis:** Linear regression, gradient descent algorithm for regression model. Non-linear regression model.	**(10 Hours)**

UNIT- 6	**Temporal Data Analysis:** Temporal data, Temporal Sequence analysis, Time Series Analysis**,** Analyzing data trends using regression, creating a time-dependent model.	**(10 Hours)**
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4,5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s from the list below	
	1. Implementation of Data Science Algorithms using Python	
	2. Case study on Time Series Data Analysis	
	3. Case study on Documents clustering	
	4. Case study on Text Classification	
Term Work:		
Part - A		
Text Books: Data Science Algorithms in a Week, David Natingga, Packt Publishing Ltd., Birminggham		
Reference Books:		
Data Science: Theories, Models, Algorithms, and Analytics, S.R. Das		
Algorithms for Data Science Book by B. C. H. Steele, John Chandler, and Swarna Reddy		
Data Science For Dummies Book by Lillian Pierson, A Willey Brand		
Topics for Project based learning		

CSC-532: Dot Net Technologies		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 3 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 3
Practical: 2 hours / Week		Lab: 1
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	Basic knowledge of C/ C++ or visual basic programming.

Course Objectives:

- Identify the Basics of .Net Framework, Architecture, and programming
- Build GUI applications using .NET Framework and Visual Basics/C#.

Course Outcomes: Students will be able to

- Design and Develop GUI based Applications using Vb.Net and C#.
- Integrate different components of the .net framework, including the database.

Course Content:

UNIT- 1	Introduction to .Net, .Net Framework Features & Architecture, CLR, Common Type System, MSIL, Types of Assemblies, Class Libraries. Event Drive Programming, Methods and Events, Related with Mouse and Keyboard.	**(7 Hours)**
UNIT- 2	Programming into Visual Studio, Toolbox, Properties Window, Form Designer, Form Layout, Immediate Window.	**(7 Hours)**
UNIT- 3	VB.Net language, Variables, Data Types, Scope & Lifetime of a Variable, Arrays, Types of Array, Control Array, Subroutine, Functions, Passing Argument to Functions, Optional Argument, Returning Value from Function.	**(8 Hours)**
UNIT- 4	Conditional and Loop Statement. Loading, Showing and Hiding Forms, Working with Multiple Forms, Controlling one Form within Another, Overview of C#, Structure of C# Program, C# in .Net.	**(7 Hours)**
UNIT- 5	GUI Programming, Windows Form and Properties, Methods, Events, Text Box Control, Label Control, Button Control, List box, Combo Box, Checked Box, Picture Box, Radio Button, Scroll Bar, Timer Control, Common Dialog Control, Designing Menus, MDI Forms.	**(8 Hours)**

UNIT- 6	ADO .Net Architecture, Create Connection, Accessing Data Using Data Adapters and Datasets, Using Command & Data Reader, Data Bind Controls, Displaying Data in Data Grid. Data Form Wizard, SQL queries, Database Using Ado.Net Object Model, Connection Object, Command Object, Add, Delete, Move & Update Records to Dataset, Executing Queries.	**8 Hours)**
Internal Assessment:		
	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	
Text Books:		
• Steven Holzner, Visual Basic .NET Programming Black Book, Dreamtech Publications • E. Balagurusamy, Programming in C# A primer, Tata McGraw-Hill Publishing Company Limited, Delhi.		
Reference Books:		
• Jeffrey R. Shapiro, Visual Basic.NET: The Complete Reference, McGraw Hill Education		

CSC-533: COMPILER DESIGN		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
		Total: 4

Course Pre-requisites: The students should have knowledge of

1	Theory of Computation

Course Objectives:

	To understand various models to produce tokens which are inputs syntax phase. Try to understand various parsing techniques such as top-down and bottom-up parsing techniques. Symbol Table generation and mechanisms to store information while scanning source code from various phases of the compiler. Semantic analysis to check the meaning of the sentences in a particular sentence.

Course Outcomes: The student will be able to

1	Students are able to understand the functionality of compiler design various phases.
2	Able to learn functionalities of various phases.
3	Able to design phases of compiler as a programming exercise.
4	Able Design various parsing techniques such as SLR, LALR and CLR.

Course Content:

UNIT- 1	**Overview of Compilation:** Phases of Compilation – Lexical Analysis, Regular Grammar and regular expression for common programming language features, pass and Phases of translation, interpretation, bootstrapping, data structures in compilation – LEX lexical analyzer generator.	**(10 Hours)**

UNIT- 2	**Top down Parsing:** Context-free grammars, Top down parsing – Backtracking, LL (1), recursive descent parsing, Predictive parsing, Preprocessing steps required for predictive parsing.	**(10 Hours)**
UNIT- 3	**Bottom up parsing:** Shift Reduce parsing, LR and LALR parsing, Error recovery in parsing, handling ambiguous grammar, YACC – automatic parser generator.	**(10 Hours)**
UNIT- 4	**Semantic analysis:** Intermediate forms of source Programs – abstract syntax tree, polish notation and three address codes. Attributed grammars, Syntax directed translation, Conversion of popular Programming languages language Constructs into Intermediate code forms, Type checker.	**(10 Hours)**
UNIT- 5	**Symbol Tables:** Symbol table format, organization for block structures languages, hashing, and tree structures representation of scope information. Block structures and non-block structure storage allocation: static, Runtime stack and heap storage allocation, storage allocation for arrays, strings and records.	**(10 Hours)**
UNIT- 6	**Code optimization:** Consideration for Optimization, Scope of Optimization, local optimization, loop optimization, frequency reduction, folding, DAG representation. **Data flow analysis:** Flow graph, data flow equation, global optimization, redundant sub expression elimination, Induction variable elements, Live variable analysis, Copy propagation. **Object code generation:** Object code forms, machine dependent code optimization, register allocation and assignment generic code generation algorithms, DAG for register allocation.	**(10 Hours)**

Assessment:

Internal	UNIT TEST- 1 :- UNIT- 1, 2, 3	
	UNIT TEST- 2 :- UNIT- 4, 5, 6	
External	EoSE: UNIT-1,2,3,4.5,6	

Text Books:

1. Principles of compiler design -A.V. Aho . J.D.Ullman; Pearson Education.
2. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press

Reference Books:

1. lex & yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
2. Modern Compiler Design- Dick Grune, Henry E. BAL, Cariel T. H. Jacobs, Wiley dreamtech.
3. Engineering a Compiler-Cooper & Linda, Elsevier.
4. Compiler Construction, Louden, Thomson.

CSC-534 SOFTWARE DEFINED NETWORKS

Prerequisites:

1) Computer Networks.
2) High Speed Networks.

Outline of the Course:-

The proposed course outline is to describe advanced technology in communication based on requirement and need for industry and academia. The designed course covers protocol framework which can support Software oriented networking protocol architecture which supports virtualization. Now-a-days network virtualization play key role in creating virtual local area networks (VLAN) to control the traffic generated by enterprise networks. The proposed course covers to design a state of art technology which can support Software Defined Networking.

Objectives: -

- To design protocol architecture which can meet the challenges of current user demands and needs data transmission.
- To demonstrate the performance of proposed SDN supportive protocols with Open Flow enabled networks.
- To learn simulator basics this can support SDN Functionalities.
- To design and detail study of security attacks which are going to occur in SDN supportive enterprise networks.

UNIT-I: Introduction, Centralized and Distributed Control and Data Planes, Introduction What Do They Do? Distributed Control Planes Centralized Control Planes Conclusions.

Open Flow: Introduction, Hybrid Approaches Conclusions SDN Controllers Introduction General Concepts Layer 3 Centric Plexxi Cisco One PK Conclusions.

UNIT-II: Network Programmability: Introduction, the Management Interface the Application-Network DivideModern Programmatic Interfaces, I2RSModern Orchestration Data Center Concepts and Constructs.

Introduction: The Multitenant Data Centerthe Virtualized Multitenant Data Center SDN Solutions for the Data Center Network VLANsEVPN, VxLan, NVGRE, Conclusions, Network Function Virtualization Introduction Virtualization and Data Plane I/O Services Engineered Path, Service Locations and Chaining, NFV at ETSI, Non-ETSI NFV Work, Conclusions.

UNIT-III: Network Topology and Topological Information Abstraction Introduction, Network Topology, Traditional Methods, LLDP, BGP-TE/LS,ALTO,I2RS Topology Building an SDN Framework, Introduction, Build Code First; Ask Questions Later, The Juniper SDN Framework, IETF SDN Framework(s), Open Daylight Controller/Framework, Policy, Conclusions.

UNIT-IV: Use Cases for Bandwidth Scheduling, Manipulation, and Calendaring, Introduction, Bandwidth Calendaring, Big Data and Application Hyper-Virtualization for Instant CSPF, Expanding Topology, Conclusion, Use Cases for Data Center Overlays, Big Data, and Network Function Virtualization, Introduction, Data Center Orchestration, Puppet (DevOps Solution), Network Function Virtualization (NFV), Optimized Big Data, Conclusions.

Text Books:

1. SDN: Software Defined Networks An Authoritative Review of Network Programmability Technologies By Thomas D. Nadeau, Ken Gray Publisher: O'Reilly Media Final Release Date: August 2013 Pages: 384.
2. Software Defined Networks: A Comprehensive Approach Paperback – Import, 30 Jun 2014by Paul Goransson (Author), Chuck Black (Author)
3. Software Defined Networking with Open Flow by SiamakAzodolmolky (Author).

Outcomes:

- Understanding between conventional networks and SDN Supportive networks to provide high throughput based on user needs.
- Understanding of Network Virtualization and requirements and changes in hardware design point of view.
- Virtual LAN supportive protocols and its operations to enhance the Quality of Service parameters.
- Understand and identify security vulnerabilities in open flow based networks. Understand prevention mechanism for well-known security attacks in conventional networks.
- Adaptive machine learning techniques to prevent security attacks in SDN.

CSC-535 MOBILE COMPUTING

Pre-requests to the Course:

- Computer Networks.
- Data Communication.

Course Outline: - The proposed course introduces the fundamentals of Wireless Communication, issues challenges in wireless communication. The course detail explains various generation of Wireless Networks generation those are 2G, 3G and 4G. The proposed course covers technical details layer-wise, which are Physical layer parameters such as modulation, demodulation and multiplexing techniques. MAC Layer issues such as various channel accessing schemes those are pure aloha, slotted aloha and p-persistent. The course covers in detail technical details such as packet formats of IEEE-802.11 standards for Medium accesses control to avoid collisions. Network Layer issues and challenges and details of various routing algorithms such as AODV, DSR and TORA protocols. Various TCP Enhancements for existing TCP Version which are TCP-RENO, Tahoe and SACK protocols for reliable and end-to-end communication for improving the performance.

Objectives:

- The objective is to understand various generations of Mobile Communication such as 2G, 3G and 4G.
- To study various issues and challenges in Physical layer such as analog to digital conversion and various modulation and demodulation techniques.
- Illustration of various physical layer issues like inter symbol interference, ISI Mitigation. Physical layer parameter such as refraction, reflection and signal to noise ratio to improve the quality.
- Demonstrate the Various MAC Layer challenges in Wireless Networks when compared to structured Networks.
- Study of various Routing Layer Protocols suitable for Wireless Ad-Hoc Networks and Protocol operations.
- Study of various TCP Layer issues and challenges for Wireless Networks.

UNIT-I

Introduction, Applications, A short history of wireless Communication, A market for Mobile Communications, Some open research topics, A Simple Reference Model. Overview, Wireless Transmission, Frequency for radio transmission, Regulations, Signals, Antennas, Signal Propagation, Path Loss of radio Signals, Additional signal Propagation effects, Multi-path Propagation. Multiplexing, Modulation, Spread Spectrum.

UNIT-II

Medium Accesses Control, Motivation for Specialization MAC, Hidden and exposed terminals, near and Far Terminals, SDMA, FDMA, TDMA, CDMA.

UNIT-III

Wireless LAN, IEEE 802.11: System Architecture, Protocol architecture, Physical Layer, MAC Control Layer, MAC Management, 802.11b, 80.11a,HIPERLAN: , Bluetooth : User Scenario, Architecture, Radio Layer, Link Manager Protocol, L2CAP, SDP, IEEE 802.15.

UNIT-IV

Mobile Network Layer, Mobile IP, Dynamic Host Configuration Protocol, Mobile Ad-Hoc Networks, Mobile Transport Layer, Classical TCP Improvements.

Text Books:

- Mobile Communications by JochenH.Schiller.
- Mobile Computing, Technology Applications and Service Creation by Asoke K Talukder and Roopa R Yavagal.

Reference Books:

- Stojmenovic and Cacute**,** "Handbook of Wireless Networks and Mobile Computing", *Wiley*, 2002, ISBN0471419028.
- Reza Behravanfar, "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", ISBN: 0521817331, Cambridge University Press, October 2004,
- Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden, Schwiebert, Loren, "Fundamentals of Mobile and Pervasive Computing", ISBN: 0071412379, McGraw-Hill Professional, 2005.

CSC-536 HUMAN COMPUTER INTERACTION

Human Computer Interaction deals with how humans interact with the Computer System. The course will uncover how designs are aesthetically done, details of ergonomics and evaluation techniques

Course Objectives

- Demonstrate how input-output channels work.
- To introduce the details of interaction and design.
- To discuss different evaluation techniques and cognitive methods.
- Laboratory exercises to be covered in Lab sessions.

The Human: input-output channels, Human memory, thinking, emotions, individual differences, psychology and the design of interactive systems.

The Computer: Text entry devices with focus on the design of key boards, positioning, pointing and drawing, display devices.

The Interaction: Models of interaction, ergonomics, interaction styles, elements of WIMP interfaces, interactivity, experience, engagement and fun. Paradigms for Interaction.

Design Process: The process of design, user focus, scenarios, navigation design screen design and layout, iteration & prototyping. Usability Engineering

Design rules: Principles to support usability, standards, guidelines, rules and heuristics, HCI patterns.

Evaluation Techniques: Definition and goals of evaluation, evaluation through expert analysis and user participation, choosing an evaluation method.

Cognitive methods: Goals and task hierarchies, linguistic models, challenges of display based systems, physical and device models, cognitive architectures.

Communications and collaborations models: Face to Face communication, conversations, Text based communication, group working.

BOOK:
Human Computer Interaction; Alan Dix et.al, 3rd ed., Pearson.

Outcomes:

At the end of this course, the student will be able to:

- Develop better interfaces that are more usable.
- Demonstrate understanding of design guidelines, principles and standards.

CSC-537 FRACTAL THEORY

Objective: After completion of this course students will be able to draw fractals and develop understanding of chaos.

Unit 1: The basic concepts of geometric iteration, principle of feedback processes Fundamentals of Fractals, Types of fractal (mathematical and nature), self-similarity, fractal dimension, multiple reduction copy machines, the chaos game, fractals in nature, and decoding fractals. Chaos wipes out every computer. Chaos in (nature and Math).

Unit 2: Standard mathematical fractals (Seirpinski carpet ,gasket, cantor dust , koch curve etc), limits and self similarity, Fractal dimension, Types of fractal dimension, implementation of standard fractal and calculating their dimensions. Affine transformation, Transformations, composing simple transformations, classical fractals by IFS, drawing the classical fractals using IFS.

Unit 3: Deterministic Chaos, analysis of chaos, periodic points, sensitivity, fixed points, logistic map, sensitivity dependence of initial condition, implementation and detailed analysis of logistic map (mathematically and in real life). L-systems, turtle graphics (graphical interpretation of L-Systems), Networked MRCMs, L-Systems tree and bushes, Growing classical fractals with L-Systems and their implementation.

Unit 4: Julia set (Fractal basin boundaries), complex numbers, escape and prisoners set, filled Julia set, Quaternion Julia set, exploring Julia sets by varying complex numbers.

Mandelbrot set, geometric features and properties , study structure of Mandelbrot set. Implementation of Julia set and Mandelbrot set.

Project: Students will complete a final creative project that involves researching an application to fractals and chaos. Students will create something to go along with the project, like artwork, a short story, or a computer generated image.

Learning Outcomes:

- Iterated Function System
- Escape-time Fractals
- Behavior of Chaotic logistic map
- L-system

CSC-538 SOFTWARE AGENTS AND SWARM INTELLGENCE

- Brief Introduction to S/W agent Technology
- Agent & AI
- Practical design of intelligent agent System
- Intelligent Agent application Area
- Biological Foundations of Swarm Intelligence
- Swarm Intelligence in Optimization
- Routing protocols for next-generation network Inspired by collective Behaviours of insects societies: An overview
- An Agent based approach to self-organised production
- Organic Computing and Swarm Intelligence

BOOKS:

1. Intelligent software agents: foundations and applications by Walter Brenner, Rudiger Zarnekow, Hartmut Witting Springer, 1998.

2. Swarm intelligence: introduction and applications By Christian Blum, Daniel Merkle., Springer 2008

CSC-539: Blockchain & Cyber Security		
TEACHING SCHEME:	**EXAMINATION SCHEME:**	**CREDIT ALLOTTED:**
Theory: 4 hours / Week	End Semester Examination: 60 Marks Internal Assessment: 40 Marks	Theory: 4
Practical:	Term Work & OR:	Term Work & OR:
		Total: 4

Course Pre-requisites: The students should have knowledge of Cyber Security

1	Basic concepts of Blockchain & Cyber Security

Course Objectives:

	To develop the Blockchain & Cyber Security

Course Outcomes: The student will be able to

1	Understand what and why of Blockchain
2	Explore the major components of Blockchain and Identify a use case for a Blockchain application
3	Create your own Blockchain network application

Course Content:

UNIT- 1	Introduction to Blockchain: Digital Trust, Asset, Transactions, Distributed Ledger Technology, Types of networks, Components of blockchain (cryptography, ledgers, consensus, smart contracts) Introduction to security, attacks, computer criminals, security services.	**(10 Hours)**
UNIT- 2	Cryptography: Substitution ciphers, transposition cipher, confusion, diffusion, symmetric and asymmetric encryption. DES, odes of DES. Hash function, key exchange, digital signatures and certificates.	**(10 Hours)**
UNIT- 3	PKI and Cryptography: Private keys, Public keys, Hashing, Digital Signature, Principles of Public Key Cryptosystems, Factorization, RSA Algorithm, security analysis of RSA, Exponentiation in Modular Arithmetic. Key Management in Public Key Cryptosystems: Distribution of Public Keys, Distribution of Secret keys using Public Key Cryptosystems. Discrete Logarithms, Diffie-Hellman Key Exchange.	**(10 Hours)**
UNIT- 4	Cryptocurrency: Bitcoin creation and economy, Limited Supply and Deflation, Hacks, Ethereum concept and Ethereum classic, Hacks Why it is so revolutionary.	**(10 Hours)**

UNIT- 5	Blockchain Applications: Building on the Blockchain, Ethereum Interaction - Smart Contract and Token (Fungible, non-fungible), Languages, Blockchain-as-a-service.	**(10 Hours)**
UNIT- 6	Blockchain Use Cases: Finance, Industry and Blockchain in Government Security and Research Aspects: Blockchain Security (DDos), Research Aspects in Blockchain, AI, Blockchain and Big Data.	**(10 Hours)**
Internal Assessment:		
Part- A	UNIT TEST- 1 :- UNIT- 1, 2,3	
	UNIT TEST- 2 :- UNIT- 4, 5,6	
PART- B	Assignments: Students should perform theoretical / experimental assignment/s	

Text Books:

1) Bahga, A., & Madisetti, V. (2017). Blockchain Applications: A Hands-On Approach. VPT.
2) Stalling Williams: Cryptography and Network Security: Principles and Practices, 4th Edition, Pearson Education, 2006.
3) Kaufman Charlie et.al; Network Security: Private Communication in a Public World, 2nd Ed., PHI/Pearson.

CSC-540 GAME THEORY

Course Outline:

This course provides an introduction to Game Theory. Game Theory is a mathematical framework that studies strategic interactions amongst self-interested decision makers. It has applications in a wide variety of areas, including statistical decision theory, artificial intelligence (online learning, multi-agent systems), economics and business (auctions, pricing, bargaining), political science (stability of government, military strategy), philosophy (ethics, morality and social norms) and biology (evolution, signaling behavior, fighting behavior).

Course Overview:

The novel concepts of game theory and how to find different equilibrium solutions to different types of games will be extensively covered in this course. These will be explained and elucidated with relevant examples.

This course provides a rigorous treatment of solution concepts for games with perfect and imperfect information including rationalxizability, Nash and subgame perfect Nash equilibria. It covers topics such as auction, VNM utility function, bargaining game etc. It also discusses cooperative game solution concepts-core, Shapley value and bayesian game with Cournot's duopoly.

UNIT 1- Games with Perfect Information-Strategic Games; Nash Equilibrium and Existence Properties; Some Games in Normal Form, Nash Equilibria in Zero-Sum Games, Bräss' Paradox, and more on Mixed Strategies, Games in Extensive Form, Market Equilibrium and Pricing.

UNIT 2- Electoral Competition: Median Voter Theorem; Auctions: Definitions and The role of Knowledge; Decision Making and Utility Theory; Mixed Strategy Equilibrium;

UNIT 3-The Paretian System Equilibrium, and Walrasian General Equilibrium Theory, Von Neumann and Morgenstern Utility Function, Theory of Risk Aversion, Equilibrium Theory.

UNIT 4- Sealed Bid Auctions, VCG Procedures, Generalized Vickrey Auctions, VCG Procedures, Cournot Competition and Stackelberg Equilibrium; Arrow's Impossibility Theorem, Gibbard-Satterthwaite Theorem, Bargaining Game with Alternating Offers; Bargaining Game with Alternating Offers (General Utilities); Nash Bargaining Solution; Stable Marriages; Multi-Item Auctions;

UNIT 5-Cooperative Game Theory: Cores; Stable Sets and Shapley Value.

UNIT 6- Strategic Games with Imperfect Information-Bayesian Games; Cournot's Duopoly with Imperfect Information; Radio Spectrum, With Arbitrary Distribution of Valuations

BOOKS:

1. "Fun and Games: A Text on Game Theory", Ken Binmore, A.I.T.B.S Publishers.
2. "A Course in Game Theory", Martin J. Osborne and Ariel Rubinstein, MIT Press.
3. Prajit Dutta, Strategies and Games, MIT Press

Learning Outcomes:

On successful completion of this course, students will be able to model competitive real world phenomena using concepts from game theory and identify optimal strategy and equilibrium solution for such models. They will be ready to explain the potential or proven relevance of game theory and its impact in various fields of human interaction which involve conflict of interest between two or more participants.