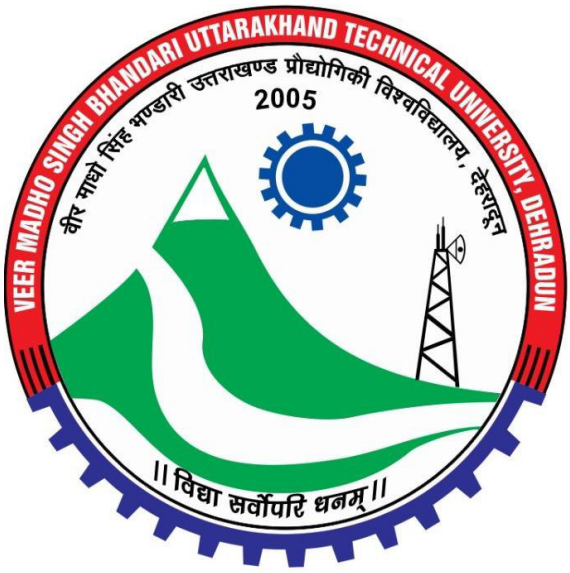
VEER MADHO SINGH BHANDARI UTTARAKHAND TECHNICAL UNIVERSITY

(Formerly Uttarakhand Technical University, Dehradun Established by Uttarakhand State Govt. wide Act no. 415 of 2005)

Suddhowala, PO-Chandanwadi, Premnagar, Dehradun, Uttarakhand (Website- www.uktech.ac.in)



SYLLABUS

For

Master of Engineering Programmes

(M.TECH-CSE)

(For admission in 2022-23 and onwards)

M.Tech Evaluation Scheme

Computer Science and Engineering

Semester I

Sr. No.	Course Type/Code	Course Name	Teaching Scheme			Credits	Internal Marks	External Marks	Total Marks
			L	T	P				
1	AHT-301	Advanced Mathematics	3	1	0	4	50	100	150
2	CST - 301	Open-Source Software Technologies	3	1	0	4	50	100	150
3	CST - 302	Advanced Data Structures and Algorithms	3	1	0	4	50	100	150
4	CST – 3XX	Program Elective - I	3	0	0	3	50	100	150
5	CST – 3XX	Program Elective - II	3	0	0	3	50	100	150
6	AHT - 302	Research Methodology and IPR	2	0	0	2	50	50	100
7	AHT-303	Audit-I	2	0	0	0	50	0	0
8	CSP- 301	Open-Source Software Technologies Lab	0	0	2	1	25	25	50
9	CSP - 302	Advanced-Data Structures and Algorithms Lab	0	0	2	1	25	25	50
Total			**22**	**3**	**8**	**22**	**400**	**600**	**950**

Program Elective-I

1. **CST-303** Machine Learning
2. **CST-304** Wireless Sensor Networks
3. **CST-305** Smart Sensors and IoT
4. **CST-306** Data Centre Management

Program Elective-II

1. **CST-307** Data Science
2. **CST-308** Distributed Systems
3. **CST-309** Advanced Wireless and Mobile Networks
4. **CST-310** Web analytics and development

Semester II									
Sr. No.	**Course Type/Code**	**Course Name**	**Teaching Scheme**			**Credits**	**Internal Marks**	**External Marks**	**Total Marks**
			L	**T**	**P**				
1	CST – 311	High Performance Scientific Computing	3	1	0	4	50	100	150
2	CST – 312	Soft Computing	3	1	0	4	50	100	150
3	CST – 3XX	Program Elective - III	3	1	0	4	50	100	150
4	CST – 3XX	Program Elective - IV	3	0	0	3	50	100	150
5	CST – 3XX	Open Elective-I	3	0	0	3	50	100	150
6	CSP - 303	High Performance Scientific Computing Lab	0	0	2	1	25	25	50
7	CSP - 304	Soft Computing Lab	0	0	2	1	25	25	50
		Total	**15**	**3**	**6**	**20**	**300**	**550**	**950**

Program Elective-III

1. **CST-313** Data Preparation and Analysis
2. **CST-314** Secure Software Design & Enterprise Computing
3. **CST-315** Computer Vision
4. **CST-316** Cyber Security

Program Elective-IV

1. **CST-317** Cloud Computing
2. **CST-318** GPU Computing
3. **CST-319** Digital Forensics
4. **CST-320** Big data analytics

Open Elective-I

1. **CST-321** Human Computer Interaction
2. **CST-322** Software Engineering
3. **CST-323** Python Programming

Semester III									
Sr. No.	**Course Type/Code**	**Course Name**	**Teaching Scheme**			**Credits**	**Internal Marks**	**External Marks**	**Total Marks**
			L	T	P				
1	CST-3XX	Open Elective - II	3	0	0	3	50	100	150
2	CSP- 305	Seminar	0	0	4	2	100		100
3	CSP- 306	Project	0	0	10	5	100	150	250
4	CSP- 307	Dissertation	0	0	12	6	300		300
		Total	**3**	**0**	**22**	**16**	**550**	**250**	**800**

Open Elective-II

1. **CST-324** Fuzzy Logic
2. **CST-325** Software Testing
3. **CST-326** Neural Networks
4. **CST-327** Web Technologies
5. **CST-328** Cyber Laws and Ethics
6. **CST-329** Mobile Application and Services

Semester IV									
Sr. No.	**Course Type/Code**	**Course Name**	**Teaching Scheme**			**Credits**	**Internal Marks**	**External Marks**	**Total marks**
			L	T	P				
1	CSP- 308	Dissertation	0	0	28	14	250	450	700
Total			**0**	**0**	**28**	**14**	**250**	**450**	**700**

Syllabus
Advanced Mathematics (AHT-301)

L:T:P:: 3:1:0 **Credits-4**

Course objectives:

From this course, students will be able to:

1. Learn distinct methods of solving simultaneous equations.
2. Well-versed with partial differential equations and their solutions and applications.
3. Acquire the knowledge of transformation to ease the complex problems.
4. Acquaintance with basics of random variables and their distribution for dealing with events by chance.
5. Study different mathematical domains to deal with real-time engineering problems.

Learning outcomes:

1. Comprehend with engineering problems in different mathematical realm.
2. Learn analytical and numerical methods to deal with mathematical problems.
3. Understand how to model the engineering problems and their solutions.
4. Implement the solutions to real-time complex engineering problems.
5. Apprehend with mathematical methodology.

Course content:

Unit I: Solution of linear simultaneous equations: (8 hours)

Consistency, Iterative method, Convergence, Cholesky's (Crout's) method, Gauss-Jordan method, Gauss-Seidel iteration and relaxation methods, Solution of Eigenvalue problems, Smallest, largest, and intermediate Eigen values

Computer based algorithm and programme for these methods (non-evaluative)

Unit II: Partial differential equation and its applications: (10 hours)

Introduction and classification of partial differential equation, Four standard forms of non-linear partial differential equations and their solutions, linear equations with constant coefficients. Applications of partial differential equationsone and two-dimensional wave equation, one and two-dimensional heat equation, Two-dimensional Laplace's equation.

Syllabus
Advanced Mathematics (AHT-301)

L:T:P:: 3:1:0 **Credits-4**

Unit III: Transform calculus-I: **(8 hours)**

Laplace transform, Properties of Laplace transform, Inverse Laplace transform,Applications of Laplace transform, Fourier integral theorem, Fourier transforms, Application of Fourier transform

Unit IV: Transform calculus-II: **(8 hours)**

Z-transform, Properties of Z-transform, Shifting theorems, Initial and final value theorem, Convolution theorems, Inverse Z-transform, Application of Z-transform

Unit V: Basic probability theory: **(8 hours)**

Concept and laws of probability,Discrete and continuous random variable and their distributions; Some special distributions such as Binomial, Poisson, Negative Binomial, Geometric, Continuous uniform, Normal, Exponential, Weibull, Moments,Moment generating functions, Expectation and variance

Practical demo with statistical software like R, SPSS, SAS, etc. (non-evaluative)

Text Books / References:

1. B.S. Grewal, Engineering Mathematics, Khanna Publications, 44th edition.
2. F.B. Hilderbrand, Method of Applied Mathematics, PHI Publications, 2nd edition.
3. M.D. Raisinghania, Ordinary and Partial Differential Equations, S. Chand Publication, 20th edition.
4. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathmematical Statistics, S. Chand Publication, 4th edition.
5. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons, 10th edition.
6. S. Ross, a First Course in Probability, Pearson Education, 8th edition.

VEER MADHO SINGH BHANDARI UTTARAKHAND TECHNICAL UNIVERSITY, DEHRADUN

CSP-301: Open-Source Software Technologies Lab

L: T: P: C: 0:0:2:1 **Credits-1**

Course Objectives:

1. To learn the setting up of GNU/Linux-based servers and workstation
2. To learn shell programming
3. To learn to configure application and server software
4. To learn to perform system administration tasks
5. To learn to use free and open-sourcecomponents.

Course Outcomes:

1. Install and run open-source operating systems.
2. Gather information about Free and Open Source Software projects from software releases and from sites on the internet.
3. Build and modify one or more Free and Open Source Software packages.
4. Apply the firewall configuration to webservers.
5. Contribute software to and interact with Free and Open Source Software development projects.

List of Programs:

S. No.	Name of the Program
1	Overview of FOSS & Basic Command interface on Linux **i.** Basic Linux Commands **ii.** File Utilities Commands
2	Learning Administrative Commands: **i.** Basic administrative commands **ii.** Vi Editor & its Modes **iii.** Network Related Commands
4	Learning Shell Scripting i. A Script to check for a file existence in the file system. ii. A Script to execute different command to demonstrate Switch cases statement. iii. A Script to handle command line argument and other Special symbols
7	Learn how to Compile, Debug & Execute C, C++ & Java Programming Code without IDEs i. Create a sample C program for demonstrating the use of control statements then compile and execute the code. ii. Create a sample C++ class based program for demonstrating the use of Object oriented Design Patterns, then compile and execute

CSP-301: Open-Source Software Technologies Lab

L: T: P: C: 0:0:2:1 **Credits-1**

	the code. iii. Creating a sample java Program for demonstrating the use of control statements then compile and execute the code.
10	Learning Basics of LAMP Server i. Installation And Configuration of LAMP Server on Linux (Ubuntu) ii. Creating simple Database in MySql Server performing queries
12	Learning A Deep Dive in MySql i. MySql Administrator: Monitoring the Server and User Administration ii. Mysql Admin, Backup and restore, User Account Rights Management
14	Basics of PHP Web Programming i. A PHP code to demonstrate the usage of Variable, String, Array and Control Structure
15	Some Deep Dive in PHP Programming i. A PHP Program to implement customized functions, Form Handling Strategies. ii. A PHP Program to demonstrate the use of PHP mail () Function
16	Learning Database Connectivity between PHP and MySql i. Create a login Control for a web page to demonstrate the use of Connectivity and Basic retrieval of data from database
17	Understanding build systems - constructing makefiles and using make, usingautoconf and autogen to automatically generate makefiles tailored for differentdevelopment environments, Using flex (lex) and bison (yacc) to design parsers.
18	Using the GNU Compiler Collection - getting acquainted with the the GNU compilertools - the C preprocessor (cpp), the C compiler (gcc) and the C++ compiler (g++),and the assembler (gas).
19	Setting up web servers - using Apache (for HTTP services), Setting up proxyservices, printer services, firewall.

TEXTBOOKS:

1. Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins, “Linux in a Nutshell”, Sixth Edition, OReilly Media, 2009.
2. Philosophy of GNU URL: http://www.gnu.org/philosophy/.
3. Linux Administration URL: http://www.tldp.org/LDP/lame/LAME/linux-admin-made-easy/.

CSP-301: Open-Source Software Technologies Lab

L: T: P: C: 0:0:2:1 **Credits-1**

4. Open Sources: Voices from the Open-Source Revolution, First Edition, January1999, ISBN: 1-56592-582-3. URL: http://www.oreilly.com/catalog/opensources/book/toc.html
5. The Linux Cookbook: Tips and Techniques for Everyday Use, First Edition,Michael Stutz, 2001. URL: http://dsl.org/cookbook/cookbook_toc.html
6. The Linux System Administrators' Guide, Lars Wirzenius, Joanna Oja, StephenStafford, and Alex Weeks, December 2003. URL: http://www.tldp.org/guides.html
7. Using GCC, Richard Stallman et al. URL: http://www.gnu.org/doc/using.html
8. An Introduction to GCC, Brian Gough. URL: http://www.networktheory.co.uk/docs/gccintro/
9. GNU Autoconf, Automake and Libtool, Gary V. Vaughan, Ben Elliston, Tom Tromey and Ian Lance Taylor. URL: http://sources.redhat.com/autobook/
10. Open Source Development with CVS, Third Edition, Karl Fogel and Moshe Bar. URL: http://cvsbook.red-bean.com/
11. Advanced Bash Scripting Guide, Mendel Cooper, June 2005.URL: http://www.tldp.org/guides.html

Advanced Data Structures and Algorithms Lab (CSP-301)

L: T: P: C: 0:0:2:1 **Credits-4**

Course Objectives:

1. To learn the implementation of linear data structures for concurrency.
2. To learn the implementation of advanced data structures such as search trees and heaps.
3. To explore the advanced concurrent data structures such as a hash table and Priority Queue.
4. To learn to apply principles of efficient algorithm design and learn various advanced algorithms.
5. Strengthen the ability of the students to identify and apply the suitable data structure for the given real-world problem.
6. Gain knowledge in practical applications of data structures.

Course Outcomes

1. Develop recursive algorithms as they apply to the Optimal binary search tree, AVL tree, and Red-Black trees.
2. Understand the concept of ADT using hashing techniques.
3. Understand Advanced Heap Structures suitable for solving Computational problems involving Optimization and analyzing these data structures using amortized analysis.
4. Implement advanced concurrent structures such as hash table & priority queue.
5. Come up with the analysis of efficiency and proof of correctness. Ability to select the data structures that efficiently model the information in a problem.

List of Programs:

1. Write a program for AVL Tree to implement all cases (LL, RR, RL, LR rotation).
2. Write a program to implement Red-Black trees with insertion and deletion operations for the given input data as Strings.
3. Write a program to implement insertion, deletion, display, and search operations in the m-way B tree (i.e., a non-leaf node can have almost m children) for the given data as integers (Test the program for m=3, 5, 7).
4. Write a program to store k keys into an array of size n at the location computed using a hash function, loc = key % n, where k<=n and k takes values from [1 to m], m>n. To handle the collisions, use the following collision resolution techniques,
   - **i.** Linear probing
   - **ii.** Quadratic probing
   - **iii.** Random probing
   - **iv.** Double hashing/rehashing
   - **v.** Chaining
5. Write a program to implement insertion, deletion, and display operation in Min-Max Heap for the given data as integers.

Advanced Data Structures and Algorithms Lab (CSP-301)

L: T: P: C: 0:0:2:1 **Credits-4**

6. Write a program to perform the following operations:
   - **i.** Insert an element into a binomial heap
   - **ii.** Delete an element from a binomial heap.
   - **iii.** Search for a key element in a binomial heap
7. Write a program to perform string matching using the Knuth-Morris-Pratt algorithm.
8. Write a program to perform string matching using the Boyer-Moore algorithm.
9. To create Markel tree structure in the blockchain.
10. Write a Python program to illustrate the following comprehensions:
    - **i.** List Comprehensions
    - **ii.** Dictionary Comprehensions
    - **iii.** Set Comprehensions
    - **iv.** Tuple Comprehensions

TEXTBOOKS:

1. Cormen T.H., Leiserson C.E, Rivest R.L. and Stein C, Introduction to Algorithms, Prentice Hall India, New Delhi, 2004.
2. Goldberg, Genetic Algorithms, Pearson Education India (1 December 2008), 1st Edition.
3. Sedgewick & Wayne, Algorithms, Addison-Wesley Professional (March 19, 2011), 4th Edition.
4. Yang, Xiaojing, Jinshan Liu, and Xiaohe Li. "Research and Analysis of Blockchain Data." Journal of Physics: Conference Series. Vol. 1237. No. 2. IOP Publishing, 2019.
5. Allen B Downey, Think Python, 2e: How to Think Like a Computer Scientist, O'Reilly; 2nd edition (15 December 2015).

REFERENCE BOOKS:

1. Kleinberg, Jon, and Eva Tardos. Algorithm design. Pearson Education India, 2006.
2. Aho A.V., Hopcroft J.E., and Ullman J.D., Data Structures and Algorithms, Pearson Education, New Delhi, 1983.
3. Sahni S., Data Structures, Algorithms, and Applications in C++, Mc Graw Hill, Singapore, 1998.
4. S.Sridhar, Design, and Analysis of Algorithms, First Edition, Oxford University Press. 2014.
5. Reema Thareja, Python Programming: Using Problem-Solving Approach, Oxford University Press; First edition (10 June 2017)

CSP-303: High Performance Computing Lab

L: T: P: C: 0:0:2:1 **Credits-1**

Course objective:

In this course you will learn how to write faster code that is highly optimized for modern multi-core processors and clusters, using modern software development tools, performance profilers, specialized algorithms, parallelization strategies, and advanced parallel programming constructs in OpenMP and MPI.

Course Outcomes:

1. **Identify** different GPU architecture basics in terms of functional units.
2. **Learn** how to install parallel programming platforms OpenMP, CUDA platform.
3. **Apply** GPU kernel implementation using CUDA/OpenMP.
4. **Evaluating** program performance using architecture specific details like GPU thread scheduling, shared memory
5. **Develop** process on GPU which uses huge datasets like image data, text data.

List of Programs:

The laboratory will require GPU based systems with OpenMP installed. The students be encouraged to configure the system. In the beginning, the students may undertake following preliminary exercises as below:

1. Analysis of Parallel Algorithms

2. Implementation using OpenMP

3. GPU kernel implementation for the given application.

4. Performance analysis using GPU memories.

5. Kernel reduction.

6. Profiling an application.

Once, the students are well versed with the environment, the exercises on the following may be taken. Datasets may be downloaded and used for the exercises:

1. Multiplication of Huge Matrices using CUDA

2. Sorting large data sets

CSP-303: High Performance Computing Lab

L: T: P: C: 0:0:2:1 **Credits-1**

3. Text Processing

4. Video Processing/ Image Analysis using CUDA

The list can be modified at the institute level depending upon the specialization of the faculty and availability of datasets.

CSP-304: Soft Computing Lab

L: T: P: C: 0:0:2:1 **Credits-1**

Course Objectives:

The course should enable the students to:

1. Understand Fuzzy concepts

2. Learn neural networks with back propagation and without preparation

3. Learn the operators of genetic algorithms

4.Design simple algorithms for pattern classification

Course Outcomes:

The course should enable the students to:

1. An understanding of fundamental concepts and methods of machine learning and its applications.
2. An ability to analyze and evaluate simple algorithms for pattern classification.
3. Able to design neural networks with back propagation and withoutpreparation.
4. Implement the operators of geneticalgorithms.
5. An ability to design simple algorithms for pattern classification, code them with Python programming language and test them with benchmark data sets.

List of Programs:

1. Introduction to Matlab/Python, Arrays and array operations, Functions and Files. Familiarization with a few ML Tools such as Excel, WEKA, R, Python and TensorFlow.
2. Study of neural network toolbox and fuzzy logic toolbox.
3. Simple implementation of Artificial Neural Network and Fuzzy Logic
4. Implementation of latest soft computing techniques using one of the above tools.
5. Regression (single and Multiple Variables) linear and non-liner;
6. Logistic regression
7. Classifiers: K-NN, Naïve Bayes Classifier, Perceptron, Multi Layer Perceptron
8. Clustering Algorithms: K-Means , DB-Scan
9. Applications of ANN and SVM using ML tools

Human Computer Interaction (CST-321)

L: T: P: C: 3:0:0:3 **Credits-3**

hrs. 40

Course objectives:

1.Provide an overview of the concepts relating to the design of human-computer interfaces

in ways making computer-based systems comprehensive, friendly and usable.

2.Understand the theoretical dimensions of human factors involved in the acceptance of computer interfaces

3.Understand the important aspects of implementation of human-computer interfaces

4.The course helps to learn basics concepts of field such as, design rules and guidelines, prototyping and design patterns for interactive systems

5. Introduction to the key areas, accessing and design developments in the field. The course aims, understanding and importance of UI its design and mistakes.

Course Outcomes:

After the completion of course the student will able to:

1. Design effective dialog for HCI
2. Design effective HCI for individuals and persons with disabilities.
3. Assess the importance of user feedback.
4. Explain the HCI implications for designing multimedia/ ecommerce/ e-learning Web sites.
5. Develop meaningful user interface.

UNIT – I **8Hrs**

FOUNDATIONS OF HCI : The Human: I/O channels – Memory – Reasoning and problem solving; The Computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms. – Case Studies.

UNIT – II **8hrs**

DESIGN & SOFTWARE PROCESS: Interactive Design: Basics – process – scenarios – navigation – screen design – Iteration and prototyping. HCI in software process: Software life cycle – usability engineering – Prototyping in practice – design rationale. Design rules: principles, standards, guidelines, rules. Evaluation Techniques – Universal Design.

Human Computer Interaction (CST-321)

L: T: P: C: 3:0:0:3 **Credits-3**

UNIT – III **8hrs**

MODELS AND THEORIES: HCI Models: Cognitive models: Socio-Organizational issues and stakeholder requirements –Communication and collaboration models-Hypertext, Multimedia and WWW.

UNIT – IV **8hrs**

MOBILE HCI: Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools. – Case Studies

UNIT – V **8Hrs**

WEB INTERFACE DESIGN: Designing Web Interfaces – Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow – Case Studies

TEXTBOOKS:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, ―Human Computer Interaction, 3rd Edition, Pearson Education, 2004.
2. Brian Fling, ―Mobile Design and Development, First Edition, O‘Reilly Media Inc., 2009.
3. Bill Scott and Theresa Neil, ―Designing Web Interfaces, First Edition, O‘Reilly, 2009.

REFERENCE BOOKS:

1. Human – Computer Interaction. ALAN DIX, JANET FINCAY, GRE GORYD, ABOWD, RUSSELL BEALG, PEARSON.
2. Interaction Design PRECE, ROGERS, SHARPS. Wiley Dreamtech.
3. User Interface Design, Soren Lauesen , Pearson Education.

Software Engineering (CST-322)

L: T: P: C: 3:0:0:3 **Credits-3**

Hrs
40

Course Objectives:

- The program will prepare our students to be successful professionals in the field with solid fundamental knowledge of software engineering.
- Be successful professionals in the field with solid fundamental knowledge of software engineering
- Utilize and exhibit strong communication and interpersonal skills, as well as professional and ethical principles when functioning as members and leaders of multi-disciplinary teams
- Apply their foundations in software engineering to adapt to readily changing environments using the appropriate theory, principles and processes

Course Outcomes:

After the completion of course the student will able to:

1. Explain various software characteristics and analyze different software Development Models
2. Demonstrate the contents of a SRS and apply basic software quality assurance practices to ensure that design, development meet or exceed applicable standards.
3. Compare and contrast various methods for software design
4. Formulate testing strategy for software systems, employ techniques such as unit testing, Test driven development and functional testing.
5. Manage software development process independently as well as in teams and make use of Various software management tools for development, maintenance and analysis.

UNIT – I **8Hrs**

Introduction: Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Software Quality Attributes. Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models.

Software Engineering (CST-322)

L: T: P: C: 3:0:0:3 **Credits-3**

UNIT – II **8Hrs**

Software Requirement Specifications (SRS): Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modelling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model.

UNIT – III **8Hrs**

Software Design: Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design. Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.

UNIT – IV **8Hrs**

Software Testing: Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, TopDown and BottomUp Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards.

UNIT – V **8Hrs**

Software Maintenance and Software Project Management: Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re- Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.

Software Engineering (CST-322)

L: T: P: C: 3:0:0:3 **Credits-3**

TEXTBOOKS:

1. RS Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
2. Pankaj Jalote, Software Engineering, Wiley.
3. Rajib Mall, Fundamentals of Software Engineering, PHI Publication.
4. KK Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.

REFERENCE BOOKS:

1. Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication.
2. Ian Sommerville, Software Engineering, Addison Wesley.
3. Kassem Saleh, “Software Engineering”, Cengage Learning.
4. P fleeger, Software Engineering, Macmillan Publication

Python Programming (CST-323)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives:

- To acquire programming skills in core Python.
- To acquire Object Oriented Skills in Python
- To develop the skill of designing Graphical user Interfaces in Python
- To develop the ability to write database applications in Python

Course Outcomes:

After the completion of course the student will able to:

1. Define the syntax and semantics of python programming language and Understand control flow statements, strings and functions.
2. Determine the methods to create and manipulate python programs by utilizing the data structures like lists, dictionaries, tuples and sets.
3. Annotate the concepts of functions, modules and packages in python.
4. Understand the concepts of files, exception handling and also apply the object-oriented programming concept by creating classes and objects.
5. Illustrate and use some of the libraries available with python.
6. Applying the problem-solving concepts to various applications using python.

UNIT - I **8 Hour**

BASICS OF PYTHON PROGRAMMING: Introduction to Python – Literals – Variables and Identifiers – Data Types – Input Operation – Comments – Reserved words – Indentation – Operators and Expressions - Conditionals: Boolean values and operators - conditional if - alternative if - chained conditional - Iteration - Illustrative programs: Evaluation of expressions - String Operations - Circulate the values of n variables - Square root (Newton's method) - GCD - Sum an Array of Numbers.

UNIT - II **8 Hour**

STRING, LISTS, TUPLES, DICTIONARIES, SETS: Strings: String slices - Immutability - String functions and methods - String module - Lists: List operations - List slices - List methods - List loop – MutabilityAliasing - Cloning lists - List parameters - Tuples: Tuple assignment - Tuple as return value.

Dictionaries: Operations and Methods - Advanced list processing - List comprehension - Sets: Creating Sets – Operations and methods – Set comprehension - Illustrative programs: Linear search - Binary search - Selection sort - Insertion sort - Merge sort.

Python Programming (CST-323)

L: T: P: C: 3:0:0:3 **Credits-3**

UNIT - III **8 Hour**

FUNCTIONS, MODULES, PACKAGES: Functions - definition and use - Flow of execution - Parameters and argumentsFruitful functions: Return values - Parameters - Local and global scope - Function composition - Recursion - Modules – from import statement – Name of Module – Making your own modules - Packages - Packages in Python – Standard Library Modules – Globals(), Locals() and Reload(); Illustrative programs: Fibonacci series using functions - Arithmetic operations using module - Area of different shapes using packages.

UNIT - IV **8 Hour**

FILES, EXCEPTIONS, CLASSES AND OBJECTS: Files and exception: Text files - Reading and writing files - Format operator - Command line arguments - Errors and exceptions - Handling exceptions - Classes and Objects: Defining classes - Creating Objects – Data abstraction – Class constructor – Class variables and Object variables – Public and Private data members – Private Methods; Illustrative programs: Word count - Copy file - Creating user defined exception - Creating student class and object.

UNIT - **8 Hour**

NUMPY, PANDAS, MATPLOTLIB: Introduction - Basics of NumPy - N-dimensional Array in NumPy - Methods and Properties - Basics of SciPy - Broadcasting in NumPy Array Operations - Array Indexing in NumPy, Pandas - Introduction - Series - DataFrame - Matplotlib - Basics - Figures and Axes - Method subplot() - Axis container

Illustrative Programs: Multiplying a Matrix by a Vector, Solving Linear System of Equations - Using Pandas to Open CSV files - Creating a Single plot.

TEXTBOOKS:

1. Reema Thareja, “Python Programming Using Problem Solving Approach”, Oxford University Press 2018.
2. Anurag Gupta, G.P. Biswas, “Python Programming: Problem Solving, Packages and Libraries”, McGrawHill, 2020.

Python Programming (CST-323)

L: T: P: C: 3:0:0:3 **Credits-3**

REFERENCE BOOKS:

1. Allen B. Downey, “Think Python: How to Think Like a Computer Scientist”, Second edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016.
2. Guido van Rossum and Fred L. Drake Jr, “An Introduction to Python – Revised and updated for Python 3.2", Network Theory Ltd., 2011.
3. John V Guttag, “Introduction to Computation and Programming Using Python", Revised and expanded Edition, MIT Press , 2013
4. Robert Sedgewick, Kevin Wayne, Robert Dondero, “Introduction to Programming in Python: An Inter-disciplinary Approach”, Pearson India Education Services Pvt. Ltd., 2016.
5. Timothy A. Budd, “Exploring Python”, Mc-Graw Hill Education (India) Private Ltd., 2015.
6. Kenneth A. Lambert, “Fundamentals of Python: First Programs”, CENGAGE Learning, 2012.
7. Charles Dierbach, “Introduction to Computer Science using Python: A Computational Problem-Solving Focus”, Wiley India Edition, 2013.
8. Paul Gries, Jennifer Campbell and Jason Montojo, “Practical Programming: An Introduction to Computer Science using Python 3”, Second edition, Pragmatic Programmers, LLC, 2013.

Data Preparation and Analysis (CST-313)

L: T: P: C: 3:1:0:4 **Credits-4**

Course objective: **40: Hours**

This course gives an overview of Big Data, i.e., storage, retrieval and processing of big data. In addition, it also focuses on the "technologies", i.e., the tools/algorithms that are available for storage, processing of Big Data. It also helps a student to perform a variety of "analytics" on different data sets and to arrive at positive conclusions.

1. To get hands on Descriptive Statistics, Probability Distributions, Inferential Statistics etc.
2. A brief introduction about big data analytics and its technologies like Hadoop Map reduce.
3. Processing big data.
4. Employing Hadoop Map Reduce.

Course Outcomes:

1. Students will develop relevant programming abilities.
2. Students will demonstrate proficiency with statistical analysis of data.
3. Students will develop the ability to build and assess data-based models.
4. Students will execute statistical analyses with professional statistical software. Students will demonstrate skill in data management.
5. Students will apply data science concepts and methods to solve problems in real- world contexts and will communicate these solutions effectively

UNIT - I **8 Hour**

DESCRIPTIVE STATISTICS: Probability Distributions, Inferential Statistics, Inferential Statistics through hypothesis tests Regression & ANOVA, Regression ANOVA (Analysis of Variance).

UNIT - II **8 Hour**

INTRODUCTION TO BIG DATA: Big Data and its Importance, Four V's of Big Data, Drivers for Big Data, Introduction to Big Data Analytics, Big Data Analytics applications. BIG DATA TECHNOLOGIES: Hadoop's Parallel World, Data discovery, Open source technology for Big Data Analytics, cloud and Big Data, Predictive Analytics, Mobile Business Intelligence and Big Data, Crowd Sourcing Analytics, Inter- and Trans-Firewall Analytics, Information Management.

UNIT - III **8 Hour**

Data Preparation and Analysis (CST-313)

L: T: P: C: 3:1:0:4 **Credits-4**

PROCESSING BIG DATA: Integrating disparate data stores, Mapping data to the programming framework, Connecting and extracting data from storage, transforming data for processing, subdividing data in preparation for Hadoop Map Reduce.

UNIT - IV **8 Hour**

HADOOP MAPREDUCE: Employing Hadoop Map Reduce, Creating the components of Hadoop Map Reduce jobs, distributing data processing across server farms, Executing Hadoop Map Reduce jobs, monitoring the progress of job flows, The Building Blocks of Hadoop Map Reduce Distinguishing Hadoop daemons, Investigating the Hadoop Distributed File System Selecting appropriate execution modes: local, pseudo-distributed, fully distributed.

UNIT - V **8 Hour**

BIG DATA TOOLS AND TECHNIQUES: Installing and Running Pig, Comparison with Databases, Pig Latin, User- Define Functions, Data Processing Operators, Installing and Running Hive, Hive QL, Querying Data, User-Defined Functions, Oracle Big Data.

TEXTBOOKS:

1. Michael Minelli, Michehe Chambers, “Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today’s Business”, 1st Edition, AmbigaDhiraj, Wiely CIO Series, 2013.
2. ArvindSathi, “Big Data Analytics: Disruptive Technologies for Changing the Game”, 1st Edition, IBM Corporation, 2012.1. Rajaraman, A., Ullman, J. D., Mining of Massive Datasets, Cambridge University Press, United Kingdom, 2012
3. Berman, J.J., Principles of Big Data: Preparing, Sharing and Analyzing Complex Information, Morgan Kaufmann, 2014

REFERENCE BOOKS:

1. Barlow, M., Real-Time Big Data Analytics: Emerging Architecture, O Reilly, 2013
2. Schonberger, V.M., Kenneth Cukier, K., Big Data, John Murray Publishers, 2013
3. Bill Franks, “Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics”, 1st Edition, Wiley and SAS Business Series, 2012.

Secure Software Design & Enterprise Computing(CST-314)

L: T: P: C: 3:1:0:4 **Credits-4**

Course Outcomes: **40: Hours**

1. Understand Software process vulnerabilities for an organization.
2. Design and develop multi-tier solution of a problem.
3. Able to administer Enterprise System.
4. Troubleshoot Enterprise Network.
5. Develop secure software which can defend against attackers

UNIT - I **8 Hour**

Secure Software Design Identify software vulnerabilities and perform software security analysis, Master security programming practices, Master fundamental software security design concepts, Perform security testing and quality assurance.

UNIT - II **8 Hour**

Enterprise Application Development Describe the nature and scope of enterprise software applications, Design distributed N-tier software application, Research technologies available for the presentation, business and data tiers of an enterprise software application, Design and build a database using an enterprise database system, Develop components at the different tiers in an enterprise system, Design and develop a multi-tier solution to a problem using technologies used in enterprise system, Present software solution.

UNIT - III **8 Hour**

Enterprise Systems Administration Design, implement and maintain a directory-based server infrastructure in a heterogeneous systems environment, Monitor server resource utilization for system reliability and availability, Install and administer network services (DNS/DHCP/Terminal Services/Clustering/Web/Email).

UNIT - IV **8 Hour**

Enterprise Network Obtain the ability to manage and troubleshoot a network running multiple services, Understand the requirements of an enterprise network.

Secure Software Design & Enterprise Computing (CST-314)

L: T: P: C: 3:1:0:4 **Credits-4**

UNIT - V **8 Hour**

Defending Applications Handle insecure exceptions and command/SQL injection, Defend web and mobile applications against attackers, software containing minimum vulnerabilities and flaws.

TEXTBOOKS:

1. Theodor Richardson, Charles N Thies, Secure Software Design, Jones & Bartlett
2. Kenneth R. van Wyk, Mark G. Graff, Dan S. Peters, Diana L. Burley, Enterprise SoftwareSecurity, Addison Wesley.
3. Principles of Secure Software Design: Dr. Raimundas Matuleviičius

REFERENCE BOOKS:

1. Architecting Applications for the Enterprise: Dino Esposito and Andrea Saltarello; Microsoft Press.
2. Enterprise Applications Administration: Jeremy Faircloth; Morgan Kaufmann publishers
3. RedHat Linux Networking & System Administration: Terry Collings and Kurt Wall; Wiley Publishing.
4. SQL Injection Attacks and Defense: Justin Clarke; Elsevier Publishing;

Computer Vision (CST-315)

L: T: P: C: 3:1:0:4 **Credits-4**

Course Outcomes:

1. Be familiar with both the theoretical and practical aspects of computing with images.
2. Have described the foundation of image formation, measurement, and analysis.
3. Have implemented common methods for robust image matching and alignment.
4. Understand the geometric relationships between 2D images and the 3D world.
5. Have gained exposure to object and scene recognition and categorization from images

UNIT - I **8-Hour**

Digital Image Formation and low-level processing Overview and State-of-the-art, Fundamentals of Image Formation, Transformation: Orthogonal, Euclidean, Affine, Projective, etc; Fourier Transform, Convolution and Filtering, Image Enhancement, Restoration, Histogram Processing.

UNIT - II **8-Hour**

Feature Extraction Edges – Canny, LOG, DOG; Line detectors (Hough Transform), Corners - Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, Scale-Space Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT.

UNIT - III **8-Hour**

Shape Representation, Segmentation and Object Recognition Shape Representation and Segmentation, Deformable curves and surfaces, Snakes and active contours, Level set representations, Fourier and Wavelet Descriptors, Medial Representations, Multiresolution analysis, Hough transforms and other simple object recognition Methods, Shape Correspondence and Shape Matching, Shape priors for recognition.

UNIT - IV **8-Hour**

Shape from X - Shape from shading, Photometric stereo, Texture, Occluding contour detection. Motion Analysis- Regularization theory, Optical Flow: brightness constancy equation, aperture problem, Horn-Shunck method, Lucas- Kanade method, Structure from motion.

UNIT - V **8-Hour**

Application: Photo album, Face detection, Face recognition, Eigen faces, Active appearance and 3D shape models of faces, In-vehicle vision system: locating roadway, road markings, identifying road signs, locating pedestrians

TEXTBOOKS:

Computer Vision (CST-315)

L: T: P: C: 3:1:0:4 **Credits-4**

1. Computer Vision: Algorithms and Applications by Richard Szeliski.
2. Computer Vision – A modern approach, by D.Forsyth and J.Ponce, Prentice Hall.
3. Robot Vision, by B. K. P. Horn, McGraw-Hill.
4. Deep Learning, by Goodfellow, Bengio, and Courville.
5. Dictionary of Computer Vision and Image Processing, by Fisher et al.
6. Three-Dimensional Computer Vision, by Olivier Faugeras, The MIT Press.

REFERENCE BOOKS:

1. Deep Learning, by Goodfellow, Bengio, and Courville.
2. Dictionary of Computer Vision and Image Processing, by Fisher et al.
3. Three-Dimensional Computer Vision, by Olivier Faugeras, The MIT Press.

Cyber Security (CST-306)

L: T: P: C: 3:1:0:4 **Credits-4**

Course Outcomes:

1. Critique and assess the strengths and weaknesses of general cyber security models, including the CIA triad.
2. Appraise the interrelationships among elements that comprise a modern security system, including hardware, software, policies, and people.
3. Assess how all domains of security interact to achieve effective system-wide security at the enterprise level.
4. Compare the interrelationships among security roles and responsibilities in a modern information-driven enterprise—to include interrelationships across security domains (IT, physical, classification, personnel, and so on).
5. Assess the role of strategy and policy in determining the success of information security.

UNIT - I **8-Hour**

The Security Environment: Threats, vulnerabilities, and consequences Advanced persistent threats, The state of security today, Why security matters to DoD? Principles of Cybersecurity-The interrelated components of the computing environment Cybersecurity models (the CIA triad, the star model, the Parkerian hexad) Variations on a theme: computer security, information security, and information assurance.

UNIT - II **8-Hour**

Cybersecurity Management Concepts: Security governance, Management models, roles, and functions, Enterprise Roles and Structures-Information security roles and positions Alternative enterprise structures and interfaces.

UNIT - III **8-Hour**

Strategy and Strategic Planning: Strategy, Strategic planning and security, strategy The information security lifecycle, Architecting the enterprise, Security Plans and Policies- Levels of planning, Planning misalignment, The System Security Plan (SSP), Policy development and implementation.

UNIT - IV **8-Hour**

Laws and Regulatory Requirements: Timeline of Indian laws related to information security, The Federal Information Security Management Act (FISMA), Security Standards and Controls -Security standards and controls, Certification and accreditation (C&A).

Cyber Security (CST-306)

L: T: P: C: 3:1:0:4 **Credits-4**

UNIT - V **8-Hour**

Risk Management: Principles of risk, Types of risk, Risk strategies, The Risk Management Framework (RMF), Security Metrics and Key Performance Indicators (KPIs)- The challenge of security metrics, What makes a good metric? Approaches to security metrics, Metrics and FISMA

TEXTBOOKS:

1. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Nina Godbole and Sunil Belapure, Wiley INDIA.
2. Matt Bishop, "Computer Security: Art and Science", 1st edition, Addison-Wesley Professional, 2015.
3. Bill Nelson, Amelia Phillips, F.Enfinger and Christopher Stuart, "Guide to Computer Forensics and Investigations, 4th ed., Thomson Course Technology, 2010.
4. Rhodes-Ousley, Mark. Information Security: The Complete Reference, Second Edition, . Information Security Management: Concepts and Practice. New York, McGraw-Hill, 2013.

REFERENCE BOOKS:

1. Introduction to Cyber Security, Chwan-Hwa(john) Wu, J. David Irwin. CRC Press T & F Group.
2. Michael E. Whitman and Herbert J Mattord, "Principles of Information Security", 6th edition, Vikas Publishing House, 2017.
3. William Stallings, "Cryptography and Network Security: Principles and Practice", 6th Edition, PHI, 2014.
4. Whitman, Michael E. and Herbert J. Mattord. Roadmap to Information Security for IT and Infosec Managers. Boston, MA: Course Technology, 2011.
5. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC

Machine Learning (CST-303)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40-Hours**

1. Underlying machine learning theories.
2. Machine learning problems corresponding to different applications.
3. Machine learning algorithms along with their strengths and weaknesses.
4. Machine learning algorithms to solve problems of moderate complexity.

Course Outcomes

1. Apply machine learning: data, model selection, model complexity, etc.
2. Apply design and analyze the popular machine learning approaches.
3. Demonstrate various machine learning algorithms in a wide range of real-world applications.
4. Appreciate the underlying mathematical relationships within and across Machine Learning algorithms and the paradigms of supervised and un-supervised learning.
5. Be able to design and implement various machine learning algorithms in a range of real- world applications.

UNIT - I **4-Hour**

Introduction: Learning Problems, Perspectives, and Issues, Concept Learning, Version Spaces, and Candidate Eliminations, Inductive bias, Decision Tree learning, Representation, Algorithm, Heuristic Space Search.

UNIT - II

Neural Networks and Genetic Algorithms: Neural Network Representation, Problems, Perceptrons, Multilayer Networks, and Back Propagation Algorithms, Advanced Topics, Genetic Algorithms, Hypothesis Space Search, Genetic Programming Models of Evaluation and Learning.

UNIT - III **4-Hour**

Bayesian and Computational Learning: Bayes Theorem, Concept Learning, Maximum Likelihood, Minimum Description Length, Principle Bayes Optimal Classifier, Gibbs Algorithm, Naïve Bayes Classifier, Bayesian Belief Network, EM Algorithm, Probability Learning, Sample Complexity, Finite and Infinite Hypothesis Spaces, Mistake Bound Model.

UNIT - IV **4-Hour**

Instant Based Learning: K- Nearest Neighbor Learning, Locally weighted Regression, Radial Bases Functions, Case Based Learning.

Machine Learning (CST-303)

L: T: P: C: 3:0:0:3 **Credits-3**

Advanced Learning: Learning Sets of Rules, Sequential Covering Algorithm, Learning Rule Set, First Order Rules, Sets of First Order Rules, Induction on Inverted Deduction, Inverting Resolution, Analytical Learning, Perfect Domain Theories, Explanation Base Learning, FOCL Algorithm, Reinforcement Learning Task, Q-Learning, Temporal Difference Learning.

UNIT - V **4-Hour**

Analytical Learning and Reinforced Learning: Perfect Domain Theories, Explanation Based Learning, Inductive Analytical Approaches, FOCL (First Order Combined Learner) Algorithm, Reinforcement Learning, Task, Q-Learning, Temporal Difference Learning.

TEXTBOOKS:

1. Tom M. Mitchell,―Machine Learning‖, McGraw Hill Education, 2017.
2. Stephen Marsland, ―Machine Learning: An Algorithmic Perspective‖, Second Edition, Chapman and Hall/CRC, 2014.

REFERENCE BOOKS:

1. Ethem Alpaydin, ”Introduction to Machine Learning 3e (Adaptive Computation and Machine Learning Series)”, Third Edition, MIT Press,2014.
2. Jason Bell, “Machine learning – Hands on for Developers and Technical Professionals”, First Edition, Wiley, 2014.
3. Peter Flach, “Machine Learning: The Art and Science of Algorithms that Make Sense of Data”, First Edition, Cambridge University Press, 2012.
4. Stephen Marsland, “Machine Learning – An Algorithmic Perspective”, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series,2014.

Wireless Sensor Networks (CST-304)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40-Hours**

1. To understand the basics of Sensor Networks.
2. To learn various fundamental and emerging protocols in the Wireless Sensor Network (WSN).
3. To study about the issues pertaining to establishment and management of sensor networks.
4. To understand the platforms and tools of sensor networks.
5. To understand various security practices and protocols of Sensor networks.

Course Outcomes

1. Identify the functions of various components in sensor.
2. Work with MAC and network layer protocols.
3. Illustrate the issues pertaining to the establishment and management of sensor networks.
4. Design and deploy a Sensor network environment for different type of applications.
5. Implement various security techniques in WSN.

UNIT - I **4-Hour**

Introduction to Wireless Sensor Networks: Motivations, Applications, History and Design factors, Performance Metrics, Anatomy of Sensor Node.

Sensor Network Architecture: Layered, Clustered, OSI Based, Cross Layer Architecture.

UNIT - II **4-Hour**

Sensing Techniques: Types of Sensors, Sensing Coverage, High-Level Sensors, Human as a Sensor, Actuators, sensor calibration, Detecting Errors.

Designing and Deploying WSN Applications: Early WSN Deployments, General Problems, General Testing & Validation, Requirements Analysis, Top-Down Design Process, Bottom-up Implementation Process.

UNIT - III **4-Hour**

Wireless Sensor Networks (CST-304)

L: T: P: C: 3:0:0:3 **Credits-3**

Medium Access Control Protocols for WSN: Introduction, Fundamentals, Performance Requirements, Schedule-Based Protocols, Random Access-Based Protocols, Sensor-MAC Case Study.

UNIT - IV **4-Hour**

Scheduling and Data Management: Survey on Data Routing in Wireless Sensor Networks, Data Centric Protocols: SPIN, Directed Diffusion, REAR, Rumor Routing

Hierarchical Routing: LEACH, Energy Efficient Weight-Clustering Algorithm in WSN, Self-Organizing Protocol

Location-Based Protocols, QoS-Aware Protocols: SPEED, MSPEED, Real-Time Power-Aware Routing.

UNIT - V **4-Hour**

Security: Fundamentals of Network Security, Challenges of Security in Wireless Sensor Networks, Security Attacks in Sensor Networks, Protocols and Mechanisms for Security, IEEE 802.15.4, and ZigBee Security.

TEXTBOOKS:

1. Anna Forster, ―Introduction to Wireless Sensor Networks‖, Wiley-IEEE Press,2016.
2. Waltenegus Dargie, Christian Poellabauer, "Fundamentals of Wireless Sensor Networks Theory and Practice", John Wiley and Sons, 2010.
3. KazemSohraby, Daniel Minoli,TaiebZnati, "Wireless Sensor Networks: Technology, Protocols, and Applications" , John Wiley & Sons, Inc .2007.

REFERENCE BOOKS:

1. Feng Zhao and Leonidas Guibas, Wireless Sensor Networks: An Information Processing Approach, Morgan Kaufman Publishers,2004.
2. Holger Karl, Andreas willig, "Protocols and Architectures for Wireless Sensor Networks", John Wiley & Sons, Inc.2005.
3. Erdal Çayırcı , Chumming Rong, "Security in Wireless Ad Hoc and Sensor Networks", John Wiley and Sons, 2009.
4. Miguel A. Lopez-Gomez and Juan C. Tejero-Calado, Nonmember "A Lightweight and Energy-Efficient Architecture for Wireless Sensor Networks", IEEE Transactions on Consumer Electronics, Vol. 55, No. 3, August 2009.

Wireless Sensor Networks (CST-304)

L: T: P: C: 3:0:0:3 **Credits-3**

5. Yingying Chen, Jie Yang, Wade Trappe, and Richard P. Martin, "Detecting and Localizing Identity-Based Attacks in Wireless and Sensor Networks ", IEEE transactions on vehicular technology, Vol. 59, No. 5,June 2010.
6. KealanMcCusker and Noel E. O'Connor, "Low-Energy Symmetric Key Distribution in Wireless Sensor Networks", IEEE Transactions On Dependable and Secure Computing, Vol. 8, No. 3, May/June 2011.

Smart Sensor and Internet of Things (CST-305)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40-Hours**

1. To provide knowledge on Sensor Principles.
2. To provide familiarity with different sensors and their application in real life.
3. To understand Basics of IoT and enabling technologies.
4. To design IoT applications using Arduino and Raspberry pi..

Course Outcomes

1. Demonstrate knowledge on the characteristics of sensors and principles of IoT.
2. Select appropriate sensors for the given application development.
3. Design basic IoT Applications using Arduino.
4. Design IoT Applications using Raspberry Pi.
5. Perform Data Acquisition and analysis using Cloud and Tkinter.

UNIT - I **8-Hour**

Introduction to Sensors: Sensors, Criteria to choose a Sensor, Generation of Sensors.

Optical Sources and Detectors: Electronic and Optical properties of semiconductor as sensors, LED, Semiconductor lasers, Fiber optic sensors, Thermal detectors, Photo multipliers, photoconductive detectors.

Strain, Force, Torque and Pressure sensors: Strain gages, strain gage beam force sensor, piezoelectric force sensor, load cell, torque sensor, Piezo-resistive and capacitive pressure sensor, optoelectronic pressure sensors, vacuum sensors.

UNIT - II **8-Hour**

Position, Direction, Displacement, Level sensors Velocity and Acceleration sensors. Temperature sensors: thermoresistive, thermoelectric, semiconductor and optical. Piezoelectric temperature sensor.

Wearable Sensors: From fibers to textile sensors - Interlaced network -Textile sensors for physiological state monitoring - Biomechanical sensing –Non-invasive sweat monitoring by textile sensors and other applications. FBG sensor in Intelligent Clothing and Biomechanics.

UNIT - III **8-Hour**

Introduction to Internet of Things: Characteristics of IoT, Design principles of IoT, IoT Architecture and Protocols, Enabling Technologies for IoT, IoT levels and IoT vs M2M.

Smart Sensor and Internet of Things(CST-305)

L: T: P: C: 3:0:0:3 **Credits-3**

IoT Design Methodology: Design methodology, Challenges in IoT Design, IoT System Management, IoT Servers.

Basics of Arduino: Introduction to Arduino, Arduino IDE, Basic Commands for Arduino, Connecting LEDs with Arduino, Connecting LCD with Arduino.

UNIT - IV **8-Hour**

Basics of Raspberry Pi: Introduction to Raspberry pi, Installation of NOOBS on SD Card, Installation of Raspbian on SD Card, Terminal Commands, Installation of Libraries on Raspberry Pi, Getting the static IP address of Raspberry Pi, Run a Program on Raspberry Pi, Installing the Remote Desktop Server, Pi Camera, Face Recognition using Raspberry Pi, Installation of I2C driver on Raspberry Pi, SPI (serial peripheral interface) with Raspberry Pi, Programming a Raspberry Pi, Play with LED and Raspberry Pi, Reading the digital input, Reading an edge triggered input, Interfacing of Relay with Raspberry Pi, Interfacing of Relay with Raspberry Pi, Interfacing of LCD with Raspberry Pi, Interfacing LCD with Raspberry Pi in I2C mode, Interfacing of DHT11 sensor with Raspberry Pi, Interfacing of ultrasonic sensor with Raspberry Pi, Interfacing of camera with Raspberry pi.

UNIT - V **8-Hour**

Data Acquisition with Python and Tkinter: Basics-CSV file, Storing Arduino data with CSV file, plotting random numbers using matplotlib, plotting real-time from Arduino, Integrating the plots in the Tkinter window.

Connecting to the Cloud: Smart IoT Systems, DHT11 Data Logger with ThingSpeak Server, Ultrasonic Sensor Data Logger with ThingSpeak Server, Air Quality Monitoring System and Data Logger with ThingSpeak Server, Landslide Detection and Disaster Management System, Smart Motion Detector and Upload Image to gmail.com**.**

TEXTBOOKS:

1. J. Fraden, Handbook of Modern Sensors: Physical, Designs, and Applications, AIP Press, Springer, Fourth Edition, 2010.
2. Rajesh Singh, Anita Gehlot, Lovi Raj Gupta, Bhupendra Singh, Mahendra Swain, Internet of Things with Raspberry Pi and Arduino, CRC Press, 2019.

REFERENCE BOOKS:

1. D. Patranabis, Sensors and Transducers, PHI Publication, New Delhi,2003.

Smart Sensor and Internet of Things (CST-305)

L: T: P: C: 3:0:0:3 **Credits-3**

2. Jan Holler and Vlasios Tsiatsis, From Machine-to-Machine to the Internet of Things Introduction to a New Age of Intelligence, Elsevier Ltd., 2014.
3. David Hanes and Gonzalo Salgueiro, IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, Cisco Press, 2017

Data Centre Management (CST-306)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40-Hours**

1. To deliver foundation knowledge about data centre facilities infrastructure.
2. To become an authority of expertise for service providers.
3. To provide knowledge on data centers network infrastructures, designs, and security provisions.

Course Outcomes

1. Conduct data centre space planning and energy calculations.
2. Evaluate contrasting data centre design techniques, and security provisions.
3. Design data centre networking infrastructure to provide Load Balancing, Segmentation, scalability, monitoring and lights-out management.
4. Describe security requirements and solutions and identify parameters for managing and monitoring storage infrastructure in Data centres.
5. Incorporate professional standards into Data Centre Infrastructure Management (DCIM).

UNIT - I **8-Hour**

Overview of data center: Data center defined, Goals, Facilities, Role of Data Centers in Enterprise and in service provider environment, Application Architecture Model, Client Server Model, n-tier model, multitier architecture, application environment.

Data Centre Architecture: Data Centre requirements, Data center Prerequisite, Budget construction, Selection of geographical location, Selection of building, Architecture Aggregation layer, Access layer, Storage layer, Data center Transport Layer, data center services, IP infrastructure services, Application Services, Security Services, Storage services, business continuance infrastructure services.

UNIT - II **8-Hour**

Data Centre Design Overview: Characteristics of an outstanding design, Guidelines for planning Data Centre, Data center structure, Raised Floor Design and development, Design and plan against Vandalism, Types of server farm and data center, internet server farm, intranet server farm, extranet server farm, corporate data center, data center topology.

UNIT - III **8-Hour**

Network Infrastructure in a data center: Modular cabling design, Point of distribution, ISP network Infrastructure, ISP WAN links.

Data Centre Management (CST-306)

L: T: P: C: 3:0:0:3 **Credits-3**

Load balancing: load balancing terminologies, Advantage of load balancing, types of load balancing, implementing network with load balancing switches.

UNIT - IV **8-Hour**

Data Centre Security Overview: The need of data center security, Vulnerabilities and common attacks, threats, vulnerabilities, common attack, network security infrastructure-ACLs, Firewalls, IDs, Security fundamental, Cryptography, Virtual Private network, Data Centre Security framework, Security Policies, Lifecycle, and secure management framework.

UNIT - V **8-Hour**

Data center maintenance: Network Operation Centre, Network Monitoring, Data center physical security, data center logical security.

Server Administration: Best practices for system administration, system administration work automation.

Case Study: IBM Data Centre Technology, Google Data Centre, Blade Server in Data Centre.

TEXTBOOKS:

1. Kailash Jayaswal, Administering Data Centres: Servers, Storage and Voice over IP, Wiley.
2. Mauricio Arregoces, Mauizio Portolani, Data Centre fundamental, Cisco Press.
3. Data Centre Blade Server Integration Guide, Cisco Systems.
4. James Hannan, A Practical Guide to Data Centre Operations Management, Aurabach Publications.

REFERENCE BOOKS:

1. Hwaiyu Geng, Data Centre Handbook, Wiley.
2. Jim Harrison. (2012), KS18 Data Centres: An Introduction to Concepts and Design, Chartered Institution of Building Services Engineers, [ISBN: 9781906846244].

CST-317Cloud Computing

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives **30-Hours**

1. The fundamental ideas behind Cloud Computing, the evolution of the paradigm, its applicability; benefits, as well as current and future challenges;
2. the basic ideas and principles in data centre design; cloud management techniques and cloud software deployment considerations;
3. Different CPU, memory and I/O virtualization techniques that serve in offering software, computation and storage services on the cloud;
4. cloud storage technologies and relevant distributed file systems,
5. the variety of programming models and develop working experience in several of them.

Course Outcomes:

1. Identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud.
2. Describe the core issues of cloud computing such as security, privacy, and interoperability to choose the appropriate technologies, algorithms, and approaches for the identified problems.
3. Analyze various cloud computing solutions and Evaluate cloud Storage systems and Cloud security, the risks involved, its impact.
4. Apply knowledge for solving real life cloud computing problem scenario and illustrate solutions.
5. Develop appropriate cloud computing solutions and recommendations according to the applications used.

UNIT - I **6- Hour**

INTRODUCTION: Introduction to Cloud Computing – Definition of Cloud – Evolution of Cloud Computing – Underlying Principles of Parallel and Distributed Computing -Cloud Characteristics – Elasticity in Cloud – On-demand Provisioning**.**

UNIT - II **6- Hour**

CLOUD ENABLING TECHNOLOGIES: Service Oriented Architecture – REST and Systems of Systems – Web Services – Publish-Subscribe Model – Basics of Virtualization – Types of Virtualization – Implementation Levels of Virtualization – Virtualization Structures – Tools and Mechanisms – Virtualization of CPU – Memory – I/O Devices -Virtualization Support and Disaster Recovery.

CST-317Cloud Computing

L: T: P: C: 3:0:0:3 **Credits-3**

UNIT - III **6- Hour**

CLOUD ARCHITECTURE, SERVICES AND STORAGE: Layered Cloud Architecture Design – NIST Cloud Computing Reference Architecture – Public, Private and Hybrid Clouds – laaS – PaaS – SaaS – Architectural Design Challenges – Cloud Storage – Storageas-a-Service – Advantages of Cloud Storage – Cloud Storage Providers – S3.

UNIT - IV **6- Hour**

RESOURCE MANAGEMENT AND SECURITY IN CLOUD: Inter Cloud Resource Management – Resource Provisioning and Resource Provisioning Methods – Global Exchange of Cloud Resources – Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Virtual Machine Security – IAM -Security Standards.

UNIT - V **6- Hour**

CLOUD TECHNOLOGIES AND ADVANCEMENTS: Hadoop – MapReduce – Virtual Box Google App Engine – Programming Environment for Google App Engine Open Stack – Federation in the Cloud – Four Levels of Federation – Federated Services and Applications – Future of Federation.

TEXTBOOKS:

1. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012.
2. Rittinghouse, John W., and James F. Ransome, Cloud Computing: Implementation, Management and Security, CRC Press, 2017.
3. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, Mastering Cloud Computing, Tata Megraw Hill, 2013.
4. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing – A Practical Approach, Tata Megraw Hill, 2009.
5. George Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice), OReilly, 2009.

REFERENCE BOOKS:

1. Cloud Computing (2nd Edition) by Dr. Kumar Saurabh, Wiley India.

2. Cloud Computing for Dummies by Judith Hurwitz, R. Bloor, M. Kanfman, F. Halper (Wiley India Edition).

Cloud Computing CST-317

L: T: P: C: 3:0:0:3 **Credits-3**

3. Enterprise Cloud Computing by Gautam Shroff, Cambridge.
4. Cloud Security by Ronald Krutz and Russell Dean Vines, Wiley-India

GPU Computing (CST-318)

L: T: P: C: 3:0:0:3 **Credits-3**

Objective:

1. To learn parallel programming with graphics processing units (GPUs).
2. Students would learn concepts in parallel programming.
3. Students would learn to implementation of programs on GPUs, debugging and profiling parallel programs.

Course Outcomes:

1. Define terminology commonly used in parallel computing, such as efficiency and speedup.
2. Describe common GPU architectures and programming models.
3. Implement efficient algorithms for common application kernels, such as matrix multiplication.
4. Given a problem, develop an efficient parallel algorithm to solve it.
5. Given a problem, implement an efficient and correct code to solve it, analyze its performance, and give convincing written and oral presentations explaining the achievements.

UNIT – I **8Hrs.**

Introduction: History, GPU Architecture, Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel Programming, CUDA OpenCL / OpenACC, Kernels Launch parameters, Thread hierarchy, Warps/Wavefronts, Threadblocks/Workgroups, Streaming multiprocessors, 1D/2D/3D thread mapping, Device properties, Simple Programs**.**

UNIT – II **8Hrs.**

Memory: Memory hierarchy, DRAM / global, local / shared, private / local, textures, Constant Memory, Pointers, Parameter Passing, Arrays and dynamic Memory, Multi-dimensional Arrays, Memory Allocation, Memory copying across devices, Programs with matrices, Performance evaluation with different memories.

UNIT – III **8Hrs.**

Synchronization: Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU.

Functions: Device functions, Host functions, Kernels functions, Using libraries (such as Thrust), and developing libraries.

UNIT – IV **8Hrs.**

GPU Computing (CST-318)

L: T: P: C: 3:0:0:3 **Credits-3**

Support: Debugging GPU Programs. Profiling, Profile tools, Performance aspects.

Streams: Asynchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based- Synchronization - Overlapping data transfer and kernel execution, pitfalls.

UNIT – V **8Hrs.**

Advanced Topics: Dynamic parallelism, Unified Virtual Memory, Multi-GPU, processing, Peer access, Heterogeneous processing.

Case Studies: Image Processing, Graph algorithms, Simulations, Deep Learning.

TEXTBOOKS:

1. Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-mei Hwu; Morgan Kaufman.
2. CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman.
3. GPU Computing and Applications: Yiyu Cai, Simon See; Springer;

REFERENCE BOOKS:

1. Hands-On GPU Programming with Python and CUDA: Explore high-performance parallel computing with CUDA, Brian Tuomanen, Packt Publishing.
2. Gerassimos Barlas, Multicore and GPU Programming: An Integrated Approach, Morgan Kaufmann; 2nd edition (June 16, 2022)

Digital Forensics (CST-319)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives:

1. To study the fundamentals of Computer Forensics
2. To learn, analyze and validate Forensics Data
3. To study the tools and tactics associated with Cyber Forensics

Course Outcomes:

4. Describe digital forensics and relate it to an investigative process.
5. Explain the legal issues of preparing for and performing digital forensic analysis based on the investigator's position and duty.
6. Perform basic digital forensics and Demonstrate use of digital forensics tools.
7. Guide a digital forensics exercise.
8. Recognize the state of the practice and the gaps in technology, policy, and legal issues.

UNIT – I **8Hrs.**

Computer forensics fundamentals, Benefits of forensics, computer crimes, computer forensics evidence and courts, legal concerns and private issues.

UNIT – II **8Hrs.**

Understanding Computing Investigations – Procedure for corporate High-Tech investigations, understanding data recovery work station and software, conducting and investigations.

UNIT – III **8Hrs.**

Data acquisition- understanding storage formats and digital evidence, determining the best acquisition method, acquisition tools, validating data acquisitions, performing RAID data acquisitions, remote network acquisition tools, other forensics acquisitions tools.

UNIT – IV **8Hrs.**

Processing crimes and incident scenes, securing a computer incident or crime, seizing digital evidence at scene, storing digital evidence, obtaining digital hash, reviewing case.

UNIT – V **8Hrs.**

Digital Forensics (CST-319)

L: T: P: C: 3:0:0:3 **Credits-3**

Current computer forensics tools- software, hardware tools, validating and testing forensic software, addressing data-hiding techniques, performing remote acquisitions, E-Mail investigations- investigating email crime and violations, understanding E-Mail servers, specialized E-Mail forensics tool.

TEXTBOOKS:

1. Warren G. Kruse II and Jay G. Heiser, "Computer Forensics: Incident Response Essentials", Addison Wesley, 2002.
2. Nelson, B, Phillips, A, Enfinger, F, Stuart, C., "Guide to Computer Forensics and Investigations, 2nd ed., Thomson Course Technology, 2006, ISBN: 0-619-21706-5.

REFERENCE BOOKS:

1. File System Forensic Analysis, by Brian Carrier, Addison-Wesley
2. Handbook of Digital Forensics and Investigation, by Eoghan Casey, Academic Press
3. Guide to Computer Forensics and Investigations 5th Edition, Nelson, Phillips, Steuart, Cengage Learning, 2015
4. The Basics of Digital Forensics, John Sammons, Elsevier
5. Computer Forensics: Computer Crime Scene Investigation, John Vacca, Laxmi Publications
6. Digital Forensic Course Materials from http://mgt2.buffalo.edu/departments/mss/djmurray/mgs610/syllabus.htm.

Big Data Analytics (CST-311)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives

- Understand the Big Data Platform and its Use cases
- Provide an overview of Apache Hadoop
- Provide HDFS Concepts and Interfacing with HDFS
- Understand Map Reduce Jobs
- Provide hands on Hodoop Eco System
- Apply analytics on Structured, Unstructured Data. Exposure to Data Analytics

Course Outcomes:

- Work with big data platform and explore the big data analytics techniques business applications.
- Design efficient algorithms for mining the data from large volumes.
- Analyze the HADOOP and Map Reduce technologies associated with big data analytics.
- Understand the fundamentals of various big data analytics techniques.
- Build a complete business data analytics solution.

UNIT – I **8Hrs.**

Introduction to big data : Introduction to Big Data Platform – Challenges ofConventional Systems - Intelligent data analysis – Nature of Data - Analytic Processesand Tools - Analysis vs Reporting.

UNIT – II **8Hrs.**

Mining data streams : Introduction To Streams Concepts – Stream Data Model andArchitecture - Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness ina Window – Decaying Window - Real time Analytics Platform(RTAP) Applications – CaseStudies - Real Time Sentiment Analysis- Stock Market Predictions.

UNIT – III **8Hrs.**

Hadoop: History of Hadoop- the Hadoop Distributed File System – Components ofHadoop Analysing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design ofHDFS-Java interfaces to HDFS Basics- Developing a Map Reduce Application-How MapReduce Works-Anatomy of a Map Reduce Job run-Failures-Job Scheduling-Shuffle andSort – Task execution - Map Reduce Types and Formats- Map Reduce Features-Hadoop environment.

UNIT – IV **8Hrs.**

Big Data Analytics (CST-311)

L: T: P: C: 3:0:0:3 **Credits-3**

Frameworks: Applications on Big Data Using Pig and Hive – Data processing operators in Pig – Hive services – HiveQL – Querying Data in Hive - fundamentals of HBase and ZooKeeper - IBM InfoSphere BigInsights and Streams.

UNIT – V **8Hrs.**

Predictive Analytics- Simple linear regression- Multiple linear regression- Interpretationof regression coefficients. Visualizations - Visual data analysis techniques- interactiontechniques - Systems and applications.

TEXTBOOKS:

1. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.
2. Tom White "Hadoop: The Definitive Guide" Third Edition, O'reilly Media, 2012.
3. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data", McGrawHill Publishing, 2012.
4. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", CUP, 2012.
5. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley& sons, 2012.

REFERENCE BOOKS:

1. Michael Minelli, Michele Chambers, and Ambiga Dhiraj, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley,2013.
2. Frank J. Ohlhorst, Big Data Analytics: Turning Big Data into Big Money, Wiley, 2012.
3. Arvind Sathi, Big Data Analytics: Disruptive Technologies for Changing the Game, MC Press, 2012.
4. Glenn J. Myatt, "Making Sense of Data", John Wiley & Sons, 2007.
5. Pete Warden, "Big Data Glossary", O'Reilly, 2011.
6. Jiawei Han, Micheline Kamber "Data Mining Concepts and Techniques", 2nd Edition, Elsevier, Reprinted 2008.
7. Da Ruan, Guoquing Chen, Etienne E.Kerre, Geert Wets, "Intelligent Data Mining", Springer, 2007.

8. Paul Zikopoulos, Dirkde Roos, Krishnan Parasuraman, Thomas Deutsch, James Giles , David Corrigan, "Harness the Power of Big Data The IBM Big Data Platform", Tata McGraw Hill Publications, 2012.

Big Data Analytics (CST-311)

L: T: P: C: 3:0:0:3 **Credits-3**

9. Arshdeep Bahga, Vijay Madisetti, "Big Data Science & Analytics: A Hands- On Approach ",VPT, 2016
10. Bart Baesens "Analytics in a Big Data World: The Essential Guide to Data Science and its Applications (WILEY Big Data Series)", John Wiley & Sons,2014.

Data Science (CST-307)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40-Hours**

1. Generalizable extraction of knowledge from data.
2. Engineering effective solutions.
3. Basic machine learning algorithms.
4. Building recommendation systems.
5. Considerate concepts on Graphs.

Course Outcomes:

1. Describe what Data Science is and the skill sets needed to be a data scientist, the Data Science Process and how its components interact.
2. Explain in basic terms what Statistical Inference means.
3. Identify probability distributions commonly used as foundations for statistical modeling. Fit a model to data.
4. Use R to carry out basic statistical modeling and analysis and APIs and other tools to scrap the web and collect data.
5. Apply basic tools (plots, graphs, summary statistics) to carry out EDA and apply EDA and the Data Science process in a case study.

UNIT - I **8- Hour**

Introduction: What is Data Science? Big Data and Data Science hype, getting past the hype now? Data fication current landscape of perspectives, Skill sets needed, Statistical Inference, Populations and samples, Statistical modeling, probability, distributions, fitting a model, Introduction to R.

UNIT - II **8- Hour**

Exploratory Data Analysis and the Data Science Process, Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, The Data Science Process, **Case Study:** Real Direct (online real estate firm).

Three Basic Machine Learning Algorithms: Linear Regression, k-Nearest Neighbors (k-NN) K-means.

UNIT - III **8- Hour**

Spam Filters, Naive Bayes, and Wrangling, Thought Experiment: Learning by Example, Naive Bayes, **Fancy It Up:** Laplace Smoothing, Comparing Naive Bayes to k-NN, Sample Code in bash, **Scraping the Web:** APIs and Other Tools, Jake‘s Exercise: Naive Bayes for Article Classification.

Data Science (CST-307)

L: T: P: C: 3:0:0:3 **Credits-3**

Logistic Regression: Thought Experiments, Classifiers, M6D Logistic Regression Case Study, Media 6 Degrees Exercise.

UNIT - IV **8- Hour**

Recommendation Systems: Building a User-Facing Data Product, Algorithmic ingredients of a Recommendation Engine, Dimensionality Reduction, Singular Value Decomposition, Principal Component Analysis, Exercise: build your own recommendation system.

UNIT - V **8- Hour**

Mining Social-Network Graphs, Social networks as graphs, Clustering of graphs, direct discovery of communities in graphs, Partitioning of graphs, Neighborhood properties in graphs, Data Visualization, Basic principles, ideas and tools for data visualization.

Data Science and Ethical Issues: Discussions on privacy, security, ethics, A look back at Data Science.

TEXTBOOKS:

1. Cathy O‘Neil and Rachel Schutt. ―Doing Data Science, Straight Talk from the Frontline‖, O‘Reilly Media, 2013.
2. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman, ―Mining of Massive Datasets‖, Second Edition, Dreamtech Press. 2016.

REFERENCE BOOKS:

1. Kevin P. Murphy, ―Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning Series‖, MIT Press, 2012.
2. Foster Provost and Tom Fawcet,―Data Science for Business: What You Need to Know about Data Mining and Data-analytic Thinking‖, O‘Reilly Media, 2013.

Distributed Systems (CST-308)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40-Hours**

1. To understand distributed system, and the desired properties of such systems are.
2. To list the principles underlying the functioning of distributed systems, describe the problems and challenges associated with these principles, and evaluate the effectiveness and shortcomings of their solutions.
3. To recognize how the principles are applied in contemporary distributed systems, explain how they affect the software design, and be able to identify features and design decisions that may cause problems.
4. To design a distributed system that fulfills requirements with regards to key distributed systems properties (such as scalability, transparency, etc.), be able to recognize when this is not possible, and explain why.
5. To build distributed system software using basic OS mechanisms as well as higher-level middleware and languages.
6. To understand the middleware technologies.

Course Outcomes

1. Explain what a distributed system is, why you would design a system as a distributed system, and what the desired properties of such systems are.
2. Specify the properties of distributed algorithms, so called liveness and safety Properties.
3. Models of distributed systems, including failure and timing model.
4. Master basic algorithms for failure detection, leader elections, broadcast and multicast, basic shared memory in distributed systems, agreement protocols, and group communication.
5. Practice in design and implementation of selected distributed algorithms in middleware designed for group communication.

UNIT - I **8- Hour**

Characterization of Distributed Systems- Introduction, Examples of Distributed systems, Resource sharing and web, challenges, System models- Introduction, Architectural and Fundamental models, Networking and Internetworking, Interprocess Communication. Distributed objects and Remote Invocation-Introduction, Communication between distributed objects, RPC, Events and notifications, Case study-Java RMI**.**

UNIT - II **8- Hour**

Operating System Support- Introduction, OS layer, Protection, Processes and Threads, Communication and Invocation, Operating system architecture, Distributed File Systems-Introduction, File Service architecture, case study- SUN network file systems. Name Services-

Distributed Systems (CST-308)

L: T: P: C: 3:0:0:3 **Credits-3**

Introduction, Name Services and the Domain Name System, Case study of the Global Name Service, Case study of the X.500 Directory Service.

UNIT - III **8- Hour**

Peer to Peer Systems-Introduction, Napster and its legacy, Peer to Peer middleware, Routing overlays, Overlay case studies-Pastry, Tapestry, Application case studies-Squirrel, Ocean Store. Time and Global States-Introduction, Clocks, events and Process states, Synchronizing physical clocks, logical time and logical clocks, global states, distributed debugging. Coordination and Agreement - Introduction, Distributed mutual exclusion, Elections, Multicast communication, consensus, and related problems.

UNIT - IV **8- Hour**

Transactions and Concurrency control - Introduction, Transactions, Nested Transactions, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency Controls. Distributed Transactions - Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery, Replication-Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data.

UNIT - V **8- Hour**

Security - Introduction, Overview of Security techniques, Cryptographic algorithms, Digital signatures, Case studies-Kerberos, TLS, 802.11 WiFi. Distributed shared memory, Design and Implementation issues, Sequential consistency and Ivy case study, Release consistency and Munin case study, other consistency models, CORBA case study- Introduction, CORBA RMI, CORBA Services.

TEXTBOOKS:

1. G Coulouris, J Dollimore and T Kindberg “Distributed Systems Concepts and Design”, Fourth Edition, Pearson Education. [ISBN:0-13-101621-0]
2. S. Ghosh, Chapman & Hall/CRC, Taylor & Francis Group “Distributed Systems, 2010. [ISBN-13:978-1-58488-564-1].

REFERENCE BOOKS:

1. S.Mahajan and S.Shah, “Distributed Computing”, Oxford University Press.[ISBN-13:978- 0198061861]
2. Pradeep K.Sinha,” Distributed Operating Systems Concepts and Design”, PHI.[ISBN:7- 302-02411-1]

Advanced Wireless and Mobile Networks (CST-309)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40Hours**

1. To study about advanced wireless network, LTE, 4G and Evolutions from LTE to LTE.
2. To study about wireless IP architecture, Packet Data Protocol and LTE network architecture.
3. To study about adaptive link layer, hybrid ARQ and graphs routing protocol.
4. To study about mobility management, cellular network, and micro cellular networks

Course Outcomes:

1. Familiar with the latest 4G networks and LTE.
2. Understand about the wireless IP architecture and LTE network architecture.
3. Familiar with the adaptive link layer and network layer graphs and protocol.
4. Understand about the mobility management and cellular network.
5. Understand about the wireless sensor network architecture and its concept..

UNIT - I **8- Hour**

INTRODUCTION: Introduction to 1G/2G/3G/4G Terminology. Evolution of Public Mobile Services Motivation for IP Based Wireless Networks -Requirements and Targets for Long Term Evolution (LTE) - Technologies for LTE- 4G Advanced Features and Roadmap Evolutions from LTE to LTEA - Wireless Standards. Network Model-Network Connectivity-Wireless Network Design with Small World Properties.

UNIT – II **8- Hour**

WIRELESS IP NETWORK ARCHITECTURES: 3GPP Packet Data Networks - Network Architecture - Packet Data Protocol (PDP) Context -Configuring PDP Addresses on Mobile Stations - Accessing IP Networks through PS Domain – LTE network Architecture - Roaming Architecture- Protocol Architecture- Bearer Establishment Procedure -Inter-Working with other RATs.

UNIT – III **8- Hour**

ADAPTIVE LINK AND NETWORK LAYER: Link Layer Capacity of Adaptive Air Interfaces-Adaptive Transmission in Ad Hoc Networks' Adaptive Hybrid ARQ Schemes for Wireless Links-Stochastic Learning Link Layer Protocol Infrared Link Access Protocol-Graphs and Routing Protocols-Graph Theory-Routing with Topology Aggregation-Network and Aggregation Models.

Advanced Wireless and Mobile Networks (CST-309)

L: T: P: C: 3:0:0:3 **Credits-3**

UNIT – IV **8- Hour**

MOBILITY MANAGEMENT: Cellular Networks-Cellular Systems with Prioritized Handoff-Cell Residing Time Distribution Mobility Prediction in Pico- and Micro-Cellular Networks.

UNIT – V **8- Hour**

QUALITY OF SERVICE: QoS Challenges in Wireless IP Networks - QoS in 3GPP - QoS Architecture, Management and Classes -QoS Attributes - Management of End-to-End IP QoS - EPS Bearers and QoS in LTE networks.

TEXTBOOKS:

1. Ayman ElNashar, Mohamed El-saidny, Mahmoud Sherif, "Design, Deployment and Performance of 4G-LTE Networks: A Practical Approach", John Wiley & Sons, 2014.
2. Crosspoint Boulevard, "Wireless and Mobile All-IP Networks", Wiley Publication, 2005.
3. Savo Glisic," advanced wireless networks-technology and business models", Third Edition, John Wiley & Sons, Ltd, 2016.
4. Savo Glisic,"Advanced Wireless Networks-4G Technologies", John Wiley & Sons, Ltd,2006.

REFERENCE BOOKS:

1. StefaniaSesia, IssamToufik and Matthew Baker, "LTE – The UMTS Long Term Evolution From Theory to Practice", John Wiley & Sons, Inc. Publication, Second Edition, 2011.
2. Minoru Etoh, "Next Generation Mobile Systems3G and Beyond," Wiley Publications,2005.
3. Jyh-Cheng Chen and Tao Zhang, "IP-Based Next-Generation Wireless Networks Systems, Architectures, and Protocols", John Wiley & Sons, Inc. Publication,2006.

Web Analytics and Development (CST-310)

L: T: P: C: 3:0:0:3 **Credits-3**

Course Objectives: **40-Hours**

1. To Learn the fundamentals of web Analytics
2. To understand the web analytics tools
3. To learn the concepts of web data collection.
4. To acquire knowledge web analytics strategy.
5. To study about connection and robustness.

Course Outcomes

1. Update the knowledge about web analytics.
2. Demonstrate various web analytics tools.
3. Acquire the knowledge of web data collection.
4. Understand web analytics strategy.
5. Familiar about making connection and robustness social involvements.

UNIT - I **8- Hour**

INTRODUCTION TO WEB ANALYTICS: A Brief history of Web Analytics – Web Analytics Terminology – Traditional Web Analytics – Web Analytics 2.0 – Capturing Data - Tools Selection – Quality Aspects – Implementing Best Practices. Social network - Web data and methods- Graph and Matrices - Basic measures for individuals and networks - Information Visualization.

UNIT - II **8- Hour**

WEB ANALYTICS TOOLS: Content organization tool – Process measurement tools - Visitor Segmentation tools- Campaign Analysis– Commerce Measurement Tools -Google Analytics – Piwik Web Analytics – Yahoo Web Analytics – Emerging Analytics: Social – Video - Mobile.

UNIT - III **8- Hour**

WEB DATA COLLECTION: Web Traffic Data – Web Transactional Data – Web Server Data – Page Weights – Usability Studies – User Submitted Information – Integrating Form based data – Web Data Sources – Server Log Files – Page Tags – Clickstream Data – Outcomes Data – Research Data – Competitive Data.

UNIT – IV

Web Analytics and Development (CST-310)

L: T: P: C: 3:0:0:3 **Credits-3**

WEB ANALYTICS STRATEGY: Component of Web Analytics Strategy – Customer Centric Focus – Business Problem Solving Focus – Reporting vs Analysis – IT and Business Strength – Clickstream vs Web 2.0 – Vendor Specific Options and Issues.

UNIT - V **8- Hour**

CONNECTION: Making Connection - Link Analysis - Random Graphs and Network evolution - Social Connects - Affiliation and identity Connection - Connection Search – Collapse - Robustness Social involvements and diffusion of innovation.

TEXTBOOKS:

1. Hansen, Derek, Ben Sheiderman, Marc Smith, "Analyzing Social Media Networks with NodeXL: Insights from a Connected World", Morgan Kaufmann, 2010.
2. Avinash Kaushik, "Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity", Sybex, First Edition, 2009.
3. Brian Clifton, "Advanced Web Metrics with Google Analytics", Sybex , Third Edition 2012.

REFERENCE BOOKS:

1. Avinash Kaushik, "Web Analytics 2.0: The Art of Online Accountability", Wiley Publishers, 2009.
2. Easley, D., Kleinberg, J, "Networks, Crowds, and Markets: Reasoning About a Highly Connected World. New York", Cambridge University Press, 2010.
3. Eric Peterson, "Web Analytics Demystified: A Marketer's Guide to Understanding How Your Web Site Affects Your Business", Celilo Group Media, First Edition, 2004.
4. Avinash Kaushik, "Web Analytics: An Hour a Day", Sybex, PAP/ CDR Edition, Sixth Edition, 2007.

CST-301: Open-Source Software Technologies

L: T: P: C: 3:1:0:4 **Credits-4**

Course Objectives:

1. Understand the difference between open-source software and commercial software.
2. Familiarity with Linux operating system.
3. Understanding and development of web applications using open-source web technologies like Apache, MySql and PHP (LAMP/XAMP).

Course Outcomes

1. Install and manage and understand the difference between open source software and commercial software.
2. Identify, install, and run Linux operating system.
3. Identify, install open source web technologies Apache, MySql, PHP.
4. Develop web applications using LAMP.
5. Write session control PHP code for a website.

UNIT - I

Open Source and Linux: Open Source Definition, The distribution terms of open source software, open source technology importance, Free and Open Source Software (FOSS), LAMP (Linux, Apache, MySQL, PHP, Python, and Perl.). Benefits, Perspectives of Open Source software Linux and Open Source. Development process and business model of companies involved in open source (e.g., Mozilla).

UNIT - II

Linux Administration: Linux Vs Windows, Installation of Linux interactively, perform user and group administration, Administer the Linux printing subsystem, automate tasks with at, cron, Install, update, query and remove software packages with RPM.

Linux Usage Basics: Logging into the system, changing users and editing text files. Running Commands and Getting Help, Browsing the File system, Users, Groups and Permissions.

UNIT - III

Open Source OS: Android, structure of Android, Basics of Android Studio. Accessing and Running Applications on Linux: cc compiler, gcc Compiler, Mozilla Firefox. Multimedia in Linux: Listening to Audio, playing video, Using Digital Camera, Recording music / video CDs. Publishing: Open office, Working with Graphics, Printing Documents, displaying documents with Ghostscript and Acrobat, Using Scanners driven by SANE.

CST-301: Open-Source Software Technologies

L: T: P: C: 3:1:0:4 **Credits-4**

UNIT - IV

Apache and PHP, Python: Introduction to Web server. Installing Apache on Linux: httpd service. PHP: Testing Installation. Basics of PHP scripts, Variables, Data types, Operators and Expressions, Constants, Flow control functions, If statement, Loops, Arrays, Strings, Dates and Times, Forms. Introduction to Python and Nginx.

UNIT - V

MySQL Server and Application: Configuring MySQL Server, working with MySQL Databases, MySQL Tables, SQL Commands – INSERT, SELECT, UPDATE, REPLACE, DELETE. Date and Time functions in MySQL. PHP – MySQL.

Application Development: Connecting to MySQL with PHP, inserting data with PHP, Retrieving data with PHP. Developing PHP scripts for dynamic web page like Feedback form, online admission form, online test.

TEXTBOOKS:

1. James Lee, Brent Ware, Open-Source Web Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP, Addison Wesley.
2. Jason Gerner, Morgan Owens, Elizabeth Naramore, Matt Warden, Professional LAMP: Linux, Apache, MySQL and PHP5 Web Development, Wrox Publication.
3. Julie C Meloni, PHP, MySQL and Apache, Pearson Education ISBN: 81-297-0443-9.

REFERENCE BOOKS:

1. Christopher Negus, Red Hat Linux Bible Wiley Publishing ISBN: 0-7645-4333-4.
2. The Complete Reference Linux Peterson Tata McGraw HILL, ISBN: 0-07-044489-7
3. Jack Dent, Tony Gaddis, UNIX using Linux, Pub: Course Technology (Thomson Learning), ISBN: 981-240- 218-7.

CST-302: Advanced Data Structures and Algorithms

L: T: P: C: 3:1:0:4 **Credits-4**

Course Objectives:

1. Introduces the elementary discrete mathematics for computer science and engineering.
2. Expose the student to the algorithm analysis techniques, the theory of reductions, and the classification of problems into complexity classes like NP.
3. To introduce various advanced concurrent structures such as hash table and Priority Queue.
4. Comprehend and select algorithm design approaches in a problem-specific manner.

Course Outcomes

1. Understand Advanced Tree Structures for the design of efficient algorithms
2. Understand Advanced Heap Structures suitable for solving Computational problems involving Optimization and analyzing these data structures using amortized analysis.
3. Implement advanced concurrent structures such as hash table & priority queue.
4. Come up with the analysis of efficiency and proof of correctness.
5. Understand the basic operation of python data structure and Blockchain along with the data structures used in it and the challenges in Blockchain data.

UNIT - I

ADVANCED TREES: Definitions Operations on Weight Balanced Trees (Huffman Trees), 2-3 Trees and Red- Black Trees, Splay Tree. Augmenting Red-Black Trees to Dynamic Order Statistics and Interval Tree Applications. Operations on Disjoint sets and their its union-find problem Implementing Sets. Dictionaries, Priority Queues, and Concatenable Queues using 2-3 Trees.

UNIT - II

MERGEABLE HEAPS: Mergeable Heap Operations, Binomial Trees Implementing Binomial Heaps and its operations, 2-3-4. Trees and 2-3-4 Heaps. Amortization analysis and Potential Function of Fibonacci Heap Implementing Fibonacci Heap. SORTING NETWORK: Comparison network, zero-one principle, bitonic sorting, and merging network sorter.

UNIT - III

ADVANCED CONCURRENT STRUCTURES: Hashing – closed-address and open-addressed hash sets — Lock-based closed-address Concurrent hash set, Cuckoo Hashing. Lock-based concurrent skip lists –An Unbounded Heap-Based Concurrent Priority Queue – skip list-based unbounded priority queues.

CST-302: Advanced Data Structures and Algorithms

L: T: P: C: 3:1:0:4 **Credits-4**

UNIT - IV

String Matching Algorithms: Suffix arrays, Suffix trees, tries, Rabin-Karp, Knuth-Morris-Pratt, Boyer Moore algorithm.
Approximation Algorithms: Need for approximation algorithms: Introduction to P, NP, NP-Hard and NP-Complete; Deterministic, non-Deterministic Polynomial-time algorithms; Knapsack, TSP, Set Cover, Open Problems.
Randomized Algorithms: Introduction, Type of Randomized Algorithms, Quick Sort, Min-Cut, 2-SAT; Game Theoretic Techniques, Random Walks.
Online Algorithms: Introduction, Online Paging Problem, Adversary Models, k-server Problem.
Genetic Algorithm: Introduction to GA, implementation in Python, problem-solving using GA such as subset problem, TSP, Knapsack.

UNIT - V

Blockchain Data Structure: - Blockchain Architecture, Blockchain Data Structures, and Data types, Contract Data, Problems to be solved in Blockchain data analysis
Advance Data Structure in Python: List, Tuple, Dictionary, Set, etc.

TEXTBOOKS:

1. Cormen T.H., Leiserson C.E, Rivest R.L. and Stein C, Introduction to Algorithms, Prentice Hall India, New Delhi, 2004.
2. Goldberg, Genetic Algorithms, Pearson Education India (1 December 2008), 1st Edition.
3. Sedgewick & Wayne, Algorithms, Addison-Wesley Professional (March 19, 2011), 4th Edition.
4. Yang, Xiaojing, Jinshan Liu, and Xiaohe Li. "Research and Analysis of Blockchain Data." Journal of Physics: Conference Series. Vol. 1237. No. 2. IOP Publishing, 2019.
5. Allen B Downey, Think Python, 2e: How to Think Like a Computer Scientist, O'Reilly; 2nd edition (15 December 2015).

REFERENCE BOOKS:

1. Kleinberg, Jon, and Eva Tardos. Algorithm design. Pearson Education India, 2006.
2. Aho A.V., Hopcroft J.E., and Ullman J.D., Data Structures and Algorithms, Pearson Education, New Delhi, 1983.
3. Sahni S., Data Structures, Algorithms, and Applications in C++, Mc Graw Hill, Singapore, 1998.
4. S.Sridhar, Design, and Analysis of Algorithms, First Edition, Oxford University Press. 2014.

CST-302: Advanced Data Structures and Algorithms

L: T: P: C: 3:1:0:4 **Credits-4**

5. Reema Thareja, Python Programming: Using Problem-Solving Approach, Oxford University Press; First edition (10 June 2017)

High Performance Scientific Computing (CST-311)

L: T: P: C: 3:1:0:4 **Credits-4**

Course Objective: **40-Hours**

1. Introduce the basic concepts related to HPC architecture and parallel computing;

2. To discuss various computational techniques for studying soft matter systems.

3. To apply these concepts to examine complex biomolecular/materials systems that generally require large-scale HPC platform with hybrid CPU-GPU architectures.

4.OpenMP on top of Fortran for parallel programming of sharedmemory computers

Course Outcomes:

1. Solve the challenges of High Speed Networks and its related performance.
2. Communicate effectively the principles used in High Performance computing.
3. Explain the basics of high speed networking technologies and to demonstrate the knowledge of network planning and optimization.
4. Describe the key components and technologies involved in building the state of art network design applications, concepts to optimize performance of high-speed networks.
5. Design and configure networks to support a specified set of applications.

UNIT - I **8- Hour**

Introduction: Single Processor Computing, Parallel Computing, Parallel System Organization

Numerical Linear Algebra: High Performance Linear Algebra, Numerical Treatment of Differential Equations

UNIT - II **8- Hour**

Applications: Molecular Dynamics, Sorting, Graph Analytics, N-body Problems, Monte Carlo Methods, Computation Biology.

UNIT - III **8- Hour**

Interactive Python using IPython, and the IPython Notebook, Python scripting and its uses in scientific computing, Subtleties of computer arithmetic that can affect program correctness.

High Performance Scientific Computing (CST-311)

L: T: P: C: 3:1:0:4 **Credits-4**

UNIT - IV **8- Hour**

Fortran 90, a Compiled language: That is widely used in scientific computing, Makefiles for building software and checking dependencies, Analyze the cost of data communication. Registers, cache, main memory, and how this memory hierarchy affects code performance.

UNIT - V **8- Hour**

OpenMP on Top of Fortran: OpenMP on top of Fortran for parallel programming of shared memory computers, such as a multicore laptop., MPI on top of Fortran for distributed memory parallel programming, such as on a cluster, Parallel computing in IPython, Debuggers, unit tests, regression tests, verification and validation of computer codes, Graphics and visualization of computational results using Python.

TEXTBOOKS:

1. A beginner's guide to GPU programming and parallel computing with CUDA 10.x and C/C++, Jaegeun Han, Bharatkumar Sharma
2. Parallel Computing for Real-time Signal Processing and Control, M. Osman Tokhi, M. Alamgir Hossain, M. Hasan Shaheed
3. New Frontiers in High Performance Computing and Big Data,G. Fox, V. Getov, L. Grandinetti, G.R. Joubert

REFERENCE BOOKS:

1. Introduction to High Performance Scientific Computing, Victor Eijkhout, Lulu .com.
2. Parallel and High-Performance Computing, Robert Robey, Manning.
3. Introduction to High Performance Scientific Computing, David Chopp, SIAM

Soft Computing (CST-312)

L: T: P: C: 3:1:0:4 **Credits-4**

Course Objectives: **40-Hours**

The main objective of the course is to expose the students to soft computing, various types of soft computing techniques, and applications of soft computing. Upon completion of this course, the student should be able to get an idea on:

1. Fuzzy Logic, Various fuzzy systems and their functions.
2. Neural Networks, architecture, functions and various algorithms involved.
3. Genetic algorithms, its applications and advances.
4. Simple implementation of Artificial Neural Network and Fuzzy Logic.

Course Outcomes:

1. Identify and describe soft computing techniques and their roles in building intelligent machines.
2. Apply fuzzy logic and reasoning to handle uncertainty and solve various engineering problems.
3. Apply genetic algorithms to combinatorial optimization problems.
4. Evaluate and compare solutions by various soft computing approaches for a given problem.
5. Use various tools to solve soft computing problems.

UNIT – I **8 Hrs**

Introduction to Soft Computing and Neural Networks: Evolution of Computing: Soft Computing Constituents, From Conventional AI to Computational Intelligence: Machine Learning Basics.

UNIT – II **8 Hrs**

Fuzzy Logic: Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions: Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making.

UNIT - III

Neural Networks: Machine Learning Using Neural Network, Adaptive Networks, Feed forward Networks, Supervised Learning Neural Networks, Radial Basis Function Networks : Reinforcement Learning, Unsupervised Learning Neural Networks, Adaptive Resonance architectures, Advances in Neural networks.

UNIT – IV **8 Hrs**

Genetic Algorithms: Goals of optimization, comparison with traditional methods, schemata, Terminology in GA – strings, structure, parameter string, data structures, operators, coding

Soft Computing (CST-312)

L: T: P: C: 3:1:0:4 **Credits-4**

Fitness function, algorithm, applications of GA in Machine Learning: Machine Learning Approach to Knowledge Acquisition.

UNIT – V **8 Hrs**

Matlab/Python Lib: Introduction to Matlab/Python, Arrays and array operations, Functions and Files, Study of neural network toolbox and fuzzy logic toolbox, Simple implementation of Artificial Neural Network and Fuzzy Logic. Recent Trends in various classifiers, neural networks and genetic algorithm.

TEXTBOOKS:

1. N.P.Padhy, S.P.Simon, "Soft Computing with MATLAB Programming", Oxford University Press, 2015.
2. S.N.Sivanandam , S.N.Deepa, "Principles of Soft Computing", Wiley India Pvt.Ltd., 2nd Edition, 2011.
3. S.Rajasekaran, G.A.Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm, Synthesis and Applications ", PHI Learning Pvt.Ltd., 2017.

REFERENCE BOOKS:

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, EijiMizutani, ―Neuro-Fuzzy and Soft Computing‖, Prentice-Hall of India, 2002.
2. Kwang H.Lee, ―First course on Fuzzy Theory and Applications‖, Springer, 2005.
3. George J. Klir and Bo Yuan, ―Fuzzy Sets and Fuzzy Logic-Theory and Applications‖, Prentice Hall, 1996.
4. James A. Freeman and David M. Skapura, ―Neural Networks Algorithms, Applications, and Programming Techniques‖, Addison Wesley, 2003

Syllabus
Research Methodology and IPR (AHT-302)

L:T:P:: 2:0:0 **Credits-2**

40-Hours

Course Objectives: Students will be able to:

1. To understand the fundaments of research in today's world controlled by technology, ideas, concept, and creativity.
2. To understand different methods of research designing and data collections.
3. To understand the methods of report writing and its different methods of interpretations.
4. To understand research ethics and methods of research publications
5. Understand that IPR protection provides an incentive to inventors for further research work and investment in R & D, which leads to creation of new and better products, and inturn brings about, economic growth and social benefits.

Course Outcomes:

1. To understand research problem formulation.
2. To study research design and method of data collections.
3. To study methods of report writing.
4. To follow research ethics.
5. To enhance student's competence to discover new inventions.

Syllabus Contents:

UNIT I: FUNDAMENTAL OF RESEARCH **8 Hour**

Meaning of research; objectives of research; basic steps of research; criteria of good research; Research methods vs. Methodology. Types of research –criteria of good research; Meaning of research problem; selection of research problem; Approaches of investigation of solutions for research problem, Errors in selecting a research problem, Scope and objectives of research problem, Review of related literature- Meaning, necessity and sources.

Unit 2: RESEARCH DESIGN AND DATA COLLECTION **8 Hour**

Research design: Types of research design- exploratory, descriptive, diagnostic and experimental;Variables- Meaning and types; Hypothesis- Meaning, function and types of hypothesis; Null/Alternative hypothesis; Sampling- Meaning and types of sampling; Probability and Non-Probability; Tools and techniques of data collection- questionnaire, schedule, interview, observation, case study, survey etc.

Unit 3:REPORT WRITING AND ITS INTERPRETATION **8 Hour**

Meaning of Interpretation, Technique of Interpretation, Precaution in Interpretation, Significance of Report Writing, Different Steps in Writing Report, Layout of the Research Report, Types of Reports, Oral Presentation, Mechanics of Writing a Research Report, Precautions for Writing Research Reports, Conclusions.

Syllabus
Research Methodology and IPR (AHT-302)

L:T:P:: 2:0:0 **Credits-2**

Unit 4:RESEARCH ETHICS AND SCHOLARY PUBLISHING **8 Hour**

Ethics-ethical issues, ethical committees (human & animal); scholarly publishing- IMRAD concept and design of research paper, citation and acknowledgement, plagiarism and its concept and importance for scholar.

Unit 5: INTELLECTUAL PROPERTY RIGHT (IPR **8 Hour**

IPR- intellectual property rights and patent law, commercialization, New developments in IPR; copy right, royalty, trade related aspects of intellectual property rights (TRIPS); Process of Patenting and Development; Procedure for grants of patents, Patenting under PCT;Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases.

Reference Books:

1. Stuart Melville and Wayne Goddard, "Research methodology: an introduction for science &engineeringstudents'"
2. WayneGoddardandStuartMelville,"ResearchMethodology:AnIntroduction"
3. RanjitKumar,2ndEdition,"ResearchMethodology:AStepbyStepGuideforbeginners"
4. Halbert,"ResistingIntellectualProperty",Taylor&FrancisLtd,2007.
5. Mayall,"IndustrialDesign",McGrawHill,1992.
6. Niebel,"ProductDesign",McGrawHill,1974.
7. Asimov,"IntroductiontoDesign",PrenticeHall,1962.
8. Robert P. Merges, Peter S. Menell, Mark A. Lemley, " Intellectual Property in New Technological Age",2016.
9. T.Ramappa,"IntellectualPropertyRightsUnderWTO",S.Chand,2008

Technical Writing and Presentation Skills (AHT-303)

L:T:P::2:0:0 Non-credits

Course Objectives: 28-Hours

- To develop effective writing and presentation skills in students.
- To develop textual, linguistic and presentation competencies instudents appropriate for their professional careers.

Course Outcomes:

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

CO1: Write clearly and fluently to produce effective technical documents.

CO2: Demonstrate an appropriate communication style to different types of audiences both orally and written as per demand of their professional careers.

CO3: Communicate in an ethically responsible manner.

Course Contents:

WRITING SKILLS

Unit-I **(4 hours)**

Technical Writing-Basic Principles: Words-Phrases-Sentences, Construction of Cohesive Paragraphs, Elements of Style.

Unit-II **(4 hours)**

Principles of Summarizing: Abstract, Summary, Synopsis

Unit-III **(6 hours)**

Technical Reports: Salient Features, Types of Reports, Structure of Reports, Data Collection, Use of Graphic Aids, Drafting and Writing

PRESENTATION SKILLS

Unit-IV **(6 hours)**

Speaking Skills: Accuracy vs. Fluency, The Audience, Pronunciation Guidelines, Voice Control.

Unit-V **(8 hours)**

Professional Presentations: Planning, Preparing, Presentation Strategies, Overcoming, Communication Barriers, Using Technology, Effective Presentations.

References:

1. Kumar, Sanjay & Pushp Lata, “Communication Skills”, Oxford UniversityPress,2011.
2. Quirk & Randolph, “A University Grammar of English”, Pearson,2006.
3. Rutherford, Andrea J., “Basic Communication Skills for Technology”, Pearson 2007.
4. Rizvi, M Ashraf, “Effective Technical Communication”, McGraw Hill,2009.
5. Leigh, Andrew & Maynard, Michael, “The Perfect Presentation”, Random House.
6. Barker, Larry L., “Communication”, Prentice-Hall.
7. Lesikar&Flatley, “Basic Business Communication-Skills for Empowering the Internet Generation”, Tata McGraw-Hill.