APPENDIX – 27(S)
UNIVERSITY OF MADRAS
M.Sc. COMPUTER SCIENCE
CHOICE BASED CREDIT SYSTEM

SYLLABUS

(w.e.f.2022-23)

Title of the Course/ Paper	Advanced Data Structure and Algorithms		
Core – 1	I Year - I Semester	Credit: 4	

Objectives:

- Define the basic concepts of algorithms and analyze the performance of algorithms.
- Discuss various algorithm design techniques for developing algorithms.
- Discuss various searching, sorting and graph traversal algorithms.
- Understand NP completeness and identify different NP complete problems.
- Discuss various advanced topics on algorithms.

Outcomes:

- Analyze programming problem statements.
- Comprehend and select algorithm design approaches in a problem specific manner.
- Choose appropriate data structures for a specific problem
- Utilize necessary mathematical abstractions to solve problems
- Come up with analysis of efficiency and proofs of correctness

UNIT I: Introduction: Algorithm, Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Probabilistic analysis, Amortized analysis.

UNIT II: Insertion and deletion and merging with 1) binary search tree, 2) AVL tree, 3) Red Black tree, 4) B tree, 5) B+ tree and Comparison of previous tree structures . Fibonacci Heap, Fibonacci Heap Operations: Find minimum, merge, insert, extract minimum, decrease key and delete, Complexity analysis of the above data structure operations.

UNIT III: Representations of Graphs, Minimum Spanning Trees: Growing a Minimum Spanning Tree – Kruskal and Prim- Single-Source Shortest Paths: The Bellman-Ford algorithm – Single-Source Shortest paths in Directed Acyclic Graphs – Dijkstra 's Algorithm, Divide and conquer: General method, applications - Quick sort, Merge sort, Strassen's matrix multiplication, External Sort: External merge sort, K-Way Merge sorting

UNIT IV: Greedy method: General method, applications-Job sequencing with deadlines, 0/1, knapsack problem, Huffman Codes, Dynamic Programming: General method, applications-Matrix chain multiplication, 0/1 knapsack problem, Traveling salesperson problem, Reliability design.

UNIT V: Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles. Branch and Bound: General method,

applications - Traveling salesperson problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution. NP-Hard and NP-Complete problems

Recommended Texts:

1. Peter Brass; Advanced Data Structures; CAMBRIDGE UNIVERSITY PRESS;2008
2. S. Dasgupta, C. Papadimitrou, U Vazirani; Algorithms; Mc Graw Hill;2022
3. J. Klienberg and E. Tardos, Algorithm Design, Pearson Education Limited;2013

Reference Books:

1. Sartaj Sahni, Data Structures Algorithms and Applications in C++, 2nd Edition, Universities Press, 2007.
2. Ellis Horowitz, Sartaj Sahni, Rajasekharan, Fundamentals of Algorithms, 2nd Edition, Universities Press, 2009.
3. Aho V Alfred, Hapcroft E John, Ullman D Jeffry, Data Structures and Algorithms, Pearson Education, 2001.
4. Adam Drozdek, Thomson, Data Structures and Algorithms in JAVA, 3rd Edition, Cengage Learning, 2008.
5. Horowitz, Sahni, Mehta, Fundamentals of Data Structures in C++, 2nd Edition, Universities Press, 2007.

Web References:

1. https://nptel.ac.in/courses/106102064

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	M	M	S	M	S	S	S	L	M
CO 2	S	L	S	M	S	L	M	M	S	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	M	M	S	L	S	M	S	S	S

S-Strong M-Medium L-Low

Title of the Course/ Paper	Advanced PYTHON Programming		
Core – 2	I Year - I Semester	Credit: 4	

Objectives:

- To Provide advanced programming knowledge in python environment
- To Make interactive Python programs.
- To develop GUI based applications
- To utilise libraries and APIs for rapid application development
- To use python as an analytical tool for different mathematical models

Outcomes:

- Be able to program decorators, closures, lambda, iterators and generators comprehensions with in OOP.
- Learn modern data structures to include collections, array, and queues
- Use platform independent file manipulation, file pattern matching using CSV, HTML, XML, JASON
- Be able to set up a client-server program and also multiprocess applications.
- Be able to use python as an analytical and presentation tool

UNIT I: Object Oriented Python – Class, Objects. member types (public. Private, protected) self ,__init method ,method overloading. Inheritance, method overriding, Polymorphism, Encapsulation, Assertion, Decorators, Generators, Iterators.

UNIT Threading in Python: Thread ,Starting a Thread, Daemon Threads, join() a Thread, Working With Many Threads, Using a Thread Pool Executor, Race Conditions, Basic Synchronization Using Lock, Deadlock, Producer-Consumer Threading, Producer-Consumer Using Lock, Producer-Consumer Using Queue, Threading Objects, Semaphore, Timer

UNIT III: Database programming using Python: Connecting to a database (sqlite, mysql) using Python, Sending DML and DDL queries and processing the result from a Python Program. Network programming using Python: An introduction to client-server programming, Basics of TCP and UDP protocols, Introduction to socket programming, Building an HTTP client and server

UNIT IV: GUI in Python: Introduction to GUI building libraries, Widgets: Button - Canvas - Check button -Entry - Frame -Label – List box – Menu button - Menu - Message –Radio button - Scale - Scrollbar - Text – Top level – Spin box- Paned Window – Label Frame – Message Box Basic image processing using Python: Introduction to digital image processing, Basic operations on an image: Crop - Scale - Rotate - Flip - Changing contrast, brightness and color - Edge detection, blur, sharpening

UNIT V:Basic numerical processing using Python: Introduction to numpy , Creation of vectors and matrices, Matrix manipulation Basic data analysis using Python: Introduction to Pandas ,Pandas data structures – Series and DataFrame , Data wrangling using pandas : Loading a dataset into a dataframe- Selecting Columns from a dataframe - Selecting Rows from a dataframe - Adding new data in a dataframe - Deleting data from a dataframe Basic data visualization using : Introduction to Matplotlib, Scatter plot , Line plot,Bar chart ,

Histogram , Box plot.

Recommended Texts:

1. John Hunt; Advanced Guide to Python 3 Programming; Springer Nature Switzerland AG; 2019

Reference Books:

1. Eric Matthes, Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 2nd Edition, No starch Press, 2019.
2. Mark Lutz; Learning Python, 5th Edition; O'Reilly Media, 2013
3. Mark Lutz, "Programming Python", 4th edition, O'Reilly Media, 2010.

Web References:

1. https://realpython.com/tutorials/advanced/

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	M	M	S	M	S	S	S	L	M
CO 2	S	L	S	M	S	L	M	M	S	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	M	M	S	L	S	M	S	S	S

S-Strong M-Medium L-Low

Title of the Course/ Paper	Artificial Intelligence		
Core – 3	I Year - I Semester	Credit: 4	

Objectives:

- To impart knowledge about Artificial Intelligence.
- To give understanding of the main abstractions and reasoning for intelligent systems.
- To enable the students to understand the basic principles of Artificial Intelligence in various applications.
- To identify the scope of Artificial Intelligence in real life applications
- To enable decoding of human thinking process and find the ways of making the machine decide intelligently in lieu of number crunching

Outcomes:

- Solve basic AI based problems.
- Define the concept of Artificial Intelligence.
- Apply AI techniques to real-world problems to develop intelligent systems.
- Select appropriately from a range of techniques when implementing intelligent systems.
- Possess the basic knowledge of different machine learning techniques.

Unit- I: AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

Unit -II Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A*, AO* Algorithms, Problem reduction, Game Playing- adversarial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

Unit -III Knowledge representation issues, predicate logic- logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules-based deduction systems. Reasoning under uncertainty, review of probability, Baye's probabilistic interferences and dumpster Shafer theory.

Unit - IV First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, learning from observation Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning.

Unit - V Expert systems:- Introduction, basic concepts, structure of expert systems, the human element in expert systems how expert systems works, problem areas addressed by expert systems, expert systems success factors, types of expert systems, expert systems and the internet interacts web, knowledge engineering, scope of knowledge, difficulties, in knowledge acquisition methods of knowledge acquisition, machine learning, intelligent agents, selecting an appropriate knowledge acquisition method, societal impacts reasoning in artificial intelligence, inference with rules, with frames: model based reasoning, case based reasoning, explanation & meta knowledge inference with uncertainty representing uncertainty.

Recommended Texts:

1. Elaine Rich, Kevin Knight and Shivshankar Nair; Artificial Intelligence ; McGraw Hill; Third Edition;2017
2. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", Third Edition, Pearson Education;2010.

Reference Books:

1. David Poole, Alan Mackworth, Randy Goebel, "Computational Intelligence : a logical approach", Oxford University Press, 1998.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem solving", Fourth Edition, Pearson Education,2001.
3. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers,1998.
4. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems by Pearson Education, 1995

Web References:

1. https://artint.info/index.html

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	S	S	M	M	L	S	L	L
CO2	S	M	S	M	S	S	M	L	M	L
CO3	M	S	S	S	M	S	L	M	L	M
CO4	S	L	M	M	S	L	L	M	M	S
CO5	S	S	M	S	L	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Practical - 1: Data Structure and Algorithms Lab		
Core – 4	I Year - I Semester	Credit: 2	

Objectives:

- To provide the foundations of the practical implementation and usage of Algorithms and Data Structures.
- To ensure that the student evolves into a competent programmer capable of designing and analyzing implementations of algorithms and data structures for different kinds of problems.
- To expose the student to the algorithm analysis techniques,
- To make the students understand the theory of reductions, and to the classification of problems.
- To make the students to be sure of complexity classes like NP.

Outcomes:

- Design and analyze programming problem statements.
- Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
- Be familiar with programming language constructs available for rapid application development,
- Understand the necessary programmatic abstraction to solve problems.
- Gain the capacity to solve real life problems by matching to the available algorithms.

Implement the following using Java

1. Write a program to perform the following operations on a heterogeneous singly linked list. i) Creation ii) Insertion iii) Deletion iv) Traversal.
2. Write a program to perform the following operations on a heterogeneous doubly linked list. i) Creation ii) Insertion iii) Deletion iv) Traversal in both ways
3. Write a program that implements using java generic class, the stack (its operations)
4. Write a program that implements using java generic class, the queue (its operations)
5. Write a program that implements the Quick sort method.
6. Write a program that implement the Merge sort method.
7. Write a program that implement the SHELL sort method.
8. Write a program to perform the following: i) Creating a Binary Tree of integers ii) Traversing the above binary tree in preorder, inorder and postorder.
9. Write a program to perform the following: i) Creating a AVL Tree ii) insertion iii)deletion iv) Traversing the above AVL tree in preorder, inorder and postorder.
10. Write a program that uses functions to perform the following: i) Creating a SplayTree ii) traverse
11. Write a program to perform the following: i) Creating a B-Tree of integers ii) insertion iii)deletion
12. Write a program that implements Kruskal's algorithm using a disjoint set data structure. The program takes as input a file (data.txt), in which each line either represents a vertex or an edge. For the edge lines, the first integer on that line representing the starting

vertex, the second the ending vertex, and the third the weight of the edge. Use this file to construct, line by line, the graph upon which Kruskal's algorithm will be run (do NOT hardcode this graph!).

13. Write a program to simulate various graph traversing algorithms.
14. Write a program to find the minimal spanning tree of a graph using the Prim's algorithm.
15. Write a program to find shortest path using Bellman Ford's Algorithm

Recommended Texts:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++:Pearson Education; 4th Edition,

Reference Books:

1. Sartaj Sahni, Data Structures Algorithms and Applications in C++, 2nd Edition, Universities Press, 2007.
2. Ellis Horowitz, Sartaj Sahni, Rajasekharan, Fundamentals of Algorithms, 2nd Edition, Universities Press, 2009.
3. Aho V Alfred, Hapcroft E John, Ullman D Jeffry, Data Structures and Algorithms, 1st Edition, Pearson Education, 2002.
4. Adam Drozdek, Thomson, Data Structures and Algorithms in JAVA, 3rd Edition, Cengage Learning, 2008.
5. Horowitz, Sahni, Mehta, Fundamentals of Data Structures in C++, 2nd Edition, Universities Press, 2007.

Web References:

1. https://nptel.ac.in/courses/106102064

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	M	M	S	M	S	L	S	L	M
CO 2	S	L	S	M	S	L	M	M	S	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	M	M	S	L	S	M	S	L	S

S-Strong M-Medium L-Low

Title of the Course/ Paper	Practical - 2: Advanced PYTHON Programming Lab		
Core – 5	I Year - I Semester	Credit: 2	

Objectives:

- Design, implement and test readable, efficient programs that take advantage of Python built-in capabilities and follow Python best practices.
- Understand implementation differences and performance tradeoffs associated with various Python data structures.
- Develop Python applications using the modules and packages available in the Python standard library.
- Develop Python applications using third party libraries.
- Design, implement and test Python programs that include a graphical user interface, data analysis and visualization, web data extraction and web applications.

Outcomes:

- Apply exception handling and user defined exception(s) Develop Module(s) and Package(s) in python
- Possess an ability to write database applications in Python
- Implement Object Oriented concepts in programming Apply Collection modules for the data types
- Possess the Object-oriented programming skills in Python. and the skill of to design graphical-user interfaces (GUI) in Python.
- Make use of Pandas and Numpy Libraries

1) Exception

a) Write a python program to catch following exception i) Value Error ii) Index Error iii) Name Error iv) Type Error v) DivideZero Error

b) Write a python program to create user defined exceptions.

c) Write a python program to understand the use of else and finally block with try block.

d) Write a python program that uses raise and exception class to throw an exception.

2) Numpy Library

a) Create a numpy array from list, tuple with float type

b) Python program to demonstrate slicing, integer and boolean array indexing

c) Write a python program to find min, max, sum, cumulative sum of array.

d) Write a python program to demonstrate use of ndim, shape, size, dtype.

3) Numpy Library: Linear Algebra

a) Write a python program to find rank, determinant, and trace of an array.

b) Write a python program to find eigenvalues of matrices

c) Write a python program to find matrix and vector products (dot, inner, outer, product), matrix exponentiation.

d) Write a python program to solve a linear matrix equation, or system of linear scalar equations.

e) Create a white image using NumPy in Python and

f) Convert a NumPy array to an image and Convert images to NumPy array?

g) Perform Sorting, Searching and Counting using Numpy methods.

h) Write a program to demonstrate the use of the reshape() method.

4) Pandas Library

a) Write a python program to implement Pandas Series with labels.

b) Create a Pandas Series from a dictionary.

c) Creating a Pandas DataFrame.

d) Write a program which make use of following Pandas methods i) describe() ii) head() iii) tail()

e) Write a program that converts Pandas DataFrame and Series into numpy.array.

f) Write a program that demonstrates the column selection, column addition, and column deletion.

g) Write a program that demonstrates the row selection, row addition, and row deletion.

h) Get n-largest and n-smallest values from a particular column in Pandas dataFrame

5) Visualization

a) Write a program which use pandas inbuilt visualization to plot following graphs: i. Bar plots ii. Histograms iii. Line plots iv. Scatter plots

b) Write a program to demonstrate use of groupby() method.

c) Write a program to demonstrate pandas Merging, Joining and Concatenating

d) Creating dataframes from csv and excel files.

6) Object Oriented Programming:

a) Write a Python class named Person with attributes name, age, weight (kgs), height (ft) and takes them through the constructor and exposes a method get_bmi_result() which returns one of "underweight", "healthy", "obese"

b) Write a python program to demonstrate various kinds of inheritance.

c) Write a python program to demonstrate operator overloading.

b) Write a python program to create abstract classes and abstract methods.

7) MULTITHREADING

a. Write a python program to create two threads to keep a count of number of even numbers entered by the user.

b. Write a JAVA program that creates threads by extending Thread class .First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds.

Recommended Texts:

1. Martin C. Brown (Author), “Python: The Complete Reference” McGraw Hill Education, Fourth edition , 2018

Reference Books:

1. R. Nageswara Rao , “Core Python Programming” Dreamtech Press India Pvt Ltd 2018.

Web References:

1. https://realpython.com/tutorials/advanced/
2. https://onlinecourses.nptel.ac.in/noc19_cs40/preview
3. https://onlinecourses.nptel.ac.in/noc19_cs41/preview

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	M	M	S	M	L	S	S	L	M
CO 2	S	L	S	L	S	L	M	M	M	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	M	M	S	L	S	M	S	S	S

S-Strong M-Medium L-Low

Title of the Course/ Paper	Theory of Computation		
Extra Disciplinary	I Year - I Semester	Credit: 3	

Objectives:

- To give an overview of the theoretical foundations of computer science from the perspective of formal languages
- To illustrate finite state machines to solve problems in computing
- To explain the hierarchy of problems arising in the computer sciences.
- To familiarize Regular grammars, context frees grammar.
- To use basic concepts of formal languages of finite automata techniques

Outcomes:

- Use the concepts and techniques of discrete mathematics for theoretical computer science
- Design Finite Automata for different Regular Expressions and Languages
- Identify and use different formal languages and their relationship.
- To solve various problems of applying normal form techniques, push down automata and Turing Machines
- Analyze various concepts of undecidability and Computable Function and Discuss nalytically and intuitively for problem-solving situation

Unit I: Review of Mathematical Theory

Combinatorics: Review of Permutation and Combination - Mathematical Induction - Pigeon hole principle - Principle of Inclusion and Exclusion - generating function - Recurrence relations. Statements – Connectives – Truth Tables – Normal forms – Predicate calculus – Inference – Theory for Statement Calculus and Predicate Calculus

Unit-II: Regular Languages and Finite Automata

Regular Expressions, Regular Languages, Application of Finite Automata, Automata with output - Moore machine & Mealy machine, Finite Automata, Memory requirement in a recognizer, Definitions, union- intersection and complement of regular languages, Non Deterministic Finite Automata, Conversion from NFA to FA, ??- Non Deterministic Finite Automata, Conversion of NFA- ? to NFA, Kleene's Theorem, Minimization of Finite automata, Regular And Non Regular Languages – pumping lemma.?

Unit-III: Context free grammar (CFG)

Definitions and Examples, Unions Concatenations And Kleene's of Context free language, Regular Grammar for Regular Language, Derivations and Ambiguity , Unambiguous CFG and Algebraic Expressions, Backus Naur Form (BNF), Normal Form – CNF.

Unit-IV: Pushdown Automata, CFL And NCFL

Definitions, Deterministic PDA, Equivalence of CFG and PDA & Conversion, Pumping lemma for CFL, Intersections and Complements of CFL, Non-CFL.

Unit-V: Turing Machine (TM)

TM Definition, Model Of Computation, Turing Machine as Language Acceptor, TM that Compute Partial Function, Church Turing Thesis, Combining TM, Variations Of TM, Non Deterministic TM, Universal TM, Recursively and Enumerable Languages, Context sensitive languages and Chomsky hierarchy.

Recommended Texts:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman ; Introduction to Automata Theory Languages andComputation; Pearson Education, India; 3rd edition;2008
2. KENNETH H. ROSEN ; Discrete Mathematics and Its Applications (SIE) 8th Edition ;2021

Reference Books:

1. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-Automata Languages and Computation, 2nd edition, Prentice Hall of India, India.

Web References:

1. https://www.youtube.com/playlist?list=PLbtzT1TYeoMjNOGEiaRmm_vMIwUAidnQz
2. https://nptel.ac.in/courses/106106049

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	M	M	S	M	L	S	S	L	M
CO 2	S	M	S	L	S	L	M	L	M	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	L	M	S	L	L	M	S	M	S

S-Strong M-Medium L-Low

Title of the Course/ Paper	Machine Learning		
Core – 6	I Year - II Semester	Credit: 4	

Objectives:

- To provide mathematical base for Machine learning
- To provide theoretical knowledge on setting hypothesis for pattern recognition.
- To impart Knowledge of machine learning techniques for data handling
- To provide the skill to evaluate the performance of algorithms and to provide solution for various real-world applications.
- To impart the knowledge of identifying similarities and differences in various patterns of data

Outcomes:

- Recognize the characteristics of machine learning strategies.
- Apply various supervised learning methods to appropriate problems.
- Identify and integrate more than one technique to enhance the performance of learning.
- Create probabilistic and unsupervised learning models for handling unknown pattern.
- Analyze the co-occurrence of data to find interesting frequent patterns.
- Preprocess the data before applying to any real-world problem and can evaluate its performance.

Unit I: BASIC MATHEMATICS FOR MACHINE LEARNING: Regression Correlation and Regression, types of correlation – Pearson's, Spearman's correlations –Ordinary Least Squares, Fitting a regression line, logistic regression, Rank Correlation Partial and Multiple correlation- Multiple regression, multicollinearity. Gradient descent methods, Newton method, interior point methods, active set, proximity methods, accelerated gradient methods, coordinate descent, cutting planes, stochastic gradient descent. Discriminant analysis, Principal component analysis, Factor analysis, k means.

Unit II: INTRODUCTION TO MACHINE LEARNING: Introduction, Examples of various Learning Paradigms, Perspectives and Issues, Version Spaces, Finite and Infinite Hypothesis Spaces, PAC Learning, VC Dimension.

Unit III: SUPERVISED LEARNING ALGORITHMS Learning a Class from Examples, Linear, Non-linear, Multi-class and Multi-label classification, Decision Trees: ID3, Classification and Regression Trees (CART), Regression: Linear Regression, Multiple Linear Regression, Logistic Regression. Neural Networks: Introduction, Perceptron, Multilayer Perceptron, Support vector machines: Linear and Nonlinear, Kernel Functions, K-Nearest Neighbors

Unit IV: ENSEMBLE LEARNING: Ensemble Learning Model Combination Schemes, Voting, Error-Correcting Output Codes, Bagging: RandomForest Trees, Boosting: Adaboost, Stacking: UNSUPERVISED LEARNING: Introduction to clustering, Hierarchical: AGNES, DIANA, Partitional: K-means clustering, K-Mode Clustering, Self-Organizing Map, Expectation Maximization, Gaussian Mixture Models, Principal

Component Analysis (PCA), Locally Linear Embedding (LLE), Factor Analysis
Unit V: PROBABILISTIC LEARNING: Bayesian Learning, Bayes Optimal Classifier, Naïve Bayes Classifier, Bayesian Belief Networks, Mining Frequent Patterns: MACHINE LEARNING IN PRACTICE: Design, Analysis and Evaluation of Machine Learning Experiments, Other Issues: Handling imbalanced data sets

Recommended Texts:

1. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, Prentice Hall of India, Third Edition 2014.
2. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar "Foundations of Machine Learning", MIT Press, 2012.

Reference Books:

1. Tom Mitchell, "Machine Learning", McGraw Hill, 3 rd Edition,1997.
2. Charu C. Aggarwal, "Data Classification Algorithms and Applications", CRC Press, 2014.
3. Stephen Marsland, "Machine Learning – An Algorithmic Perspective", 2 nd Edition, CRC Press, 2015.
4. Kevin P. Murphy "Machine Learning: A Probabilistic Perspective", The MIT Press, 2012
5. Jiawei Han and MichelineKambers and Jian Pei, "Data Mining –Concepts and Techniques", 3 rd Edition,Morgan Kaufman Publications, 2012.
6. Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, "Mathematics for Machine Learning", Cambridge University Press, 2019.

Web References:

1. https://www.youtube.com/watch?v=r4sgKrRL2Ys&list=PL1xHD4vteKYVpaIiy295pg6_SY5qznc77

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	S	M	S	M	L	M	S	L	L
CO 2	S	M	S	L	S	L	M	L	M	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	L	M	S	L	L	M	S	M	S

S-Strong M-Medium L-Low

Title of the Course/ Paper	Advanced Networks		
Core – 7	I Year - II Semester	Credit: 4	

Objectives:

- To uncover and understand the current directions of computer networks from literature readings.
- To expose students to the "full span" of the computer network's frontier.
- To encourage a performance perspective towards analysis of computer and communications networks.
- To "fill-in" gaps in students' networking knowledge.
- To plan the interworking of distributed application basing on Semantic Web technology

Outcomes:

- Differentiate between different LAN-based forwarding devices so that they can make thoughtful suggestions on how to build a network.
- Select appropriate transport protocol and quality of service mechanisms for a give computer network
- Write networking code that uses TCP and UDP in client-server applications.
- Design and implement networking protocols.
- Design and implement networking applications.

Unit I : Reliable Transmission – Multi access Network – Wireless network - Internetworking - Switching and Bridging- Basic Internetworking -Routing - Implementation -

Unit II : Advanced Internetworking - Global Internet – Multicast - Virtual Private Network - Multiprotocol Label Switching - Routing Among Mobile Devices

Unit III: End-to-End Protocols - Simple Demultiplexer (UDP) - Reliable Byte Stream (TCP) - Remote Procedure Call - Transport for Real-Time (RTP) - RTP Design - Control Protocol - HTTP is the New Narrow Waist

Unit IV: Congestion Control - Issues in Resource Allocation - Queuing Disciplines - TCP Congestion Control - Advanced Congestion Control - Approaches to QOS - Integrated services RSVP - Differentiated Services (EF & AF) - Equation based Congestion control.

Unit V: Network Security - Trust and Threats - Cryptographic Building Blocks - Key Predistribution - Authentication Protocols - Applications - Multimedia Applications - Infrastructure Applications - Name Service (DNS) - Overlay Networks - Peer to peer networks

Recommended Texts:

1. Larry L. Peterson , Bruce S. Davie; Computer Networks A Systems Approach; Morgan Kaufmann; 3rd Edition ;2022

Reference Books:

1. C. Kaufman, R. Perlman, M. Speciner , Network Security. Private Communication in a Public World, Prentice Hall PTR; Second Edition;2022
2. J. F. Kurose, K. W. Ross , Computer Networking. A Top-Down Approach;

Addison Wesley Longman; Fourth edition, 2008

Web References:

1. https://www.youtube.com/watch?v=O--rkQNKqls&list=PLEAYkSg4uSQ2NMmzNNsEK5RVbhxqx0BZF

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	S	M	S	M	L	M	S	L	L
CO 2	S	M	S	L	S	L	M	L	M	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	M	M	S	M	L	M	S	L	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Practical - 3: Machine Learning Lab		
Core – 8	I Year - II Semester	Credit: 2	

Objectives:

- Make use of Data sets in implementing the machine learning algorithms
- Implement the machine learning concepts and algorithms in any suitable language of choice.
- The programs can be implemented in either JAVA or Python.
- For Problems 1 to 6 and 10, programs are to be developed without using the builtin classes or APIs of Java/Python.
- Data sets can be taken from standard repositories (https://archive.ics.uci.edu/ml/datasets.html) or constructed by the students.

Outcomes:

- Understand the implementation procedures for the machine learning algorithms.
- Design Java/Python programs for various Learning algorithms.
- Apply appropriate data sets to the Machine Learning algorithms.
- Identify and apply Machine Learning algorithms to solve real world problems.
- be capable of confidently applying common Machine Learning algorithms in practice and implementing their own;

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file

2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.

5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

6. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.

8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

9. Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.

10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Recommended Texts:

1. Dr. Kamlesh Namdev, LAP LAMBERT ; Lab manual of Machine Learning: Machine Learning Practicals in Python; Academic Publishing; 2021

Reference Books:

1. Introduction to Machine Learning with Python by Andreas C. Müller, Sarah Guido Released October 2016 Publisher(s): O'Reilly Media, Inc. ISBN: 9781449369415

Web References:

1. https://www.youtube.com/watch?v=RnFGwxJwx-0

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	M	S	M	M	S	S	L	L	L	M
CO4	S	L	M	M	M	L	M	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Practical - 4: Full stack web development Lab (Elective II based Lab)		
Core – 9A	I Year - II Semester	Credit: 2	

Objectives:

- To ensure responsiveness of applications
- To work alongside graphic designers for web design features
- To manage a project from conception to finished product
- To meet both technical and consumer needs for a web development project
- To learn to research new methods of development in web applications and programming languages

Outcomes:

- Structure and implement HTML/CSS.
- Apply intermediate and advanced web development practices.
- Develop a fully functioning website and deploy on a web server.
- Create webpages that function using external data.
- Identify mobile strategies and design for multiple operating systems.
- Distinguishing trends in multi-device implementation.

1. Create a Simple Login form with validations and verification using java script and PHP respectively

2. Create a student mark entry form with validation and verification using java script and PHP respectively

3. Create a employee payroll form with validation and verification using java script and PHP respectively

4. create a simple for count the number of visiters and number of times a single user visited the page.

5. Build a navigation menu that highlights the selected entry using Angular's directives

Create a simple inline editor - clicking a paragraph will show a tooltip with a text field using angularJS

6. Switch between different layout modes (grid or list) with a click of a button.

7. Create a simple application for online shopping using AngularJS and PHP session.

8. Create a Simple Login form with validations and verification using AngularJS PHP respectively

9. Create a student mark entry form with validation and verification using AngularJS and PHP respectively

10. Create a employee payroll form with validation and verification using AngularJS and PHP respectively

Recommended Texts:

1. Full Stack AngularJS for Java Developers: Build a Full-Featured Web Application from Scratch Using AngularJS with Spring RESTful, Ravi Kant Soni,Apress; 1st ed. edition ;2017

Reference Books:

1. Learning AngularJS: A Guide to AngularJS Development,Ken Williamson,O' Reilly ISBN: 9789352130702, 9352130707

Web References:

1. https://www.youtube.com/watch?v=9b9pLgaSQuI

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	M	M	M	L	M	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Practical - 4: Natural Language Processing Lab (Elective II based Lab)		
Core – 9B	I Year - II Semester	Credit: 2	

Objectives:

- To understand the algorithms available for the processing of linguistic information and computational
- properties of natural languages.
- To conceive basic knowledge on various morphological, syntactic and semantic NLP tasks.
- To familiarize various NLP software libraries and datasets publicly available.
- To develop systems for various NLP problems with moderate complexity.
- To learn various strategies for NLP system evaluation and error analysis.

Outcomes:

- Describe the concepts of morphology, syntax, semantics, discourse & pragmatics of natural language.
- Demonstrate understanding of the relationship between NLP and statistics & machine learning.
- Discover various linguistic and statistical features relevant to the basic NLP task, namely, spelling
- correction, morphological analysis, parts-of-speech tagging, parsing and semantic analysis.
- Develop systems for various NLP problems with moderate complexity

LIST OF EXERCISES

1. How to tokenize a given text?
2. How to get the sentences of a text document?
3. How to tokenize text with stop words as delimiters?
4. How to remove stop words and punctuations in a text?
5. How to perform stemming?
6. How to lemmatize a given text?
7. How to extract usernames from emails?
8. How to find the most common words in the text excluding stop words?
9. How to do spell correction in a given text?
10. How to classify a text as positive/negative sentiment?
11. How to extract Noun and Verb phrases from a text?
12. How to find the ROOT word of any word in a sentence?
13. Write a Python program to load the iris data from a given csv file into a dataframe and print the shape of the data, type of the data and first 3 rows.
14. Write a Python NLTK program to find the sets of synonyms and antonyms of a given word.
15. Write a Python NLTK program to print the first 15 random combine labeled male and labeled female names from names corpus.

Recommended Texts:

1. Jurafsky Dan and Martin James H. "Speech and Language Processing" ,3rd Edition, 2018.

Reference Books:

1. Jurafsky D. and Martin J. H., "Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", 2nd Edition, Upper Saddle River, NJ: Prentice-Hall, 2008.
2. Goldberg Yoav "A Primer on Neural Network Models for Natural Language Processing".
3. Natural Language Processing with Python, Steven Bird, Ewan Klein, and Edward Loper

Web References:

1. https://www.youtube.com/watch?v=dIUTsFT2MeQ

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	M	M	M	L	M	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Practical - 4: Digital Image Processing Lab (Elective II based Lab)		
Core – 9C	I Year - II Semester	Credit: 2	

Objectives:

- To use Java or Python or Scilab for reading and writing images.
- To understand programming tools for image processing
- To manipulate images.
- To analyse image statistics
- To apply image processing algorithms

Outcomes:

- Use image processing tools
- Perform image manipulation operations
- Perform image enhancement techniques
- Perform edge detection operations
- Possess ability to perform object recognition methods

1) Basic Manipulations
    a) Reading
    b) Writing
    c) Quantisation
    d) Subsampling
2) Basic intensity transformations
3) Histogram processing
4) Filtering in spatial domain Low pass and High pass filters
5) Frequency domain image enhancement 2D FFT
6) Color image enhancement - spatial sharpening
7) Convert a color image into grayscale image

Recommended Texts:

1. Rohit M. Thanki , Ashish M. Kothari, Digital Image Processing using SCILAB-Springer ;2018
2. Sandipan Dey; Image Processing Masterclass with Python; BPB Publications;2021

Reference Books:

1. Hands-On Image Processing with Python: Expert techniques for advanced image analysis and effective interpretation of image data by Sandipan Dey, bpb

Web References:

1. https://www.youtube.com/watch?v=oXlwWbU8l2o

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	M	M	M	L	M	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Principles of Compiler Design		
Extra Disciplinary	I Year - II Semester	Credit: 3	

Objectives:

- To learn the various phases of compiler.
- To learn the various parsing techniques.
- To understand intermediate code generation and run-time environment.
- To learn to implement the front-end of the compiler.
- To learn to implement code generators.

Outcomes:

- Understand the different phases of the compiler.
- Design a lexical analyzer for a sample language.
- Apply different parsing algorithms to develop the parsers for a given grammar.
- Design and implement a scanner and a parser using LEX and YACC tools
- Learn to implement code optimization techniques and a simple code generator.

.

UNIT I: INTRODUCTION TO COMPILERS: Structure of a compiler – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA.
UNIT II: SYNTAX ANALYSIS: Role of Parser – Grammars – Error Handling – Context-free grammars – Writing a grammar – Top Down Parsing - General Strategies Recursive Descent Parser Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser- LR (0)Item Construction of SLR Parsing Table - Introduction to LALR Parser - Error Handling and Recovery in Syntax Analyzer-YACC.
UNIT III: INTERMEDIATE CODE GENERATION: Syntax Directed Definitions, Evaluation Orders for Syntax Directed Definitions, Intermediate Languages: Syntax Tree, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking.
UNIT IV: RUN-TIME ENVIRONMENT AND CODE GENERATION: Storage Organization, Stack Allocation Space, Access to Non-local Data on the Stack, Heap Management - Issues in Code Generation - Design of a simple Code Generator.
UNIT V: CODE OPTIMIZATION: Principal Sources of Optimization – Peep-hole optimization - DAG- Optimization of Basic Blocks- Global Data Flow Analysis - Efficient Data Flow Algorithm.

Recommended Texts:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers: Principles, Techniques and Tools; Pearson Education; Second Edition; 2013

Reference Books:

1. Randy Allen, Ken Kennedy, Optimizing Compilers for Modern Architectures: A Dependence based Approach, Morgan Kaufmann Publishers, 2002.
2. Steven S. Muchnick, Advanced Compiler Design and Implementation, Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint 2003.
3. Keith D Cooper and Linda Torczon, Engineering a Compiler, Morgan Kaufmann Publishers Elsevier Science, 2004.

4. V. Raghavan, Principles of Compiler Design, Tata McGraw Hill Education Publishers, 2010.
5. Allen I. Holub, Compiler Design in C, Prentice-Hall Software Series, 1993.

Web References:

1. https://www.youtube.com/watch?v=k4QXWFZZq1E&list=PLENQMW_c1dimxHUu6KjuBC2rOlAaoLozF

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Cloud Computing		
Elective -1	I Year - II Semester	Credit: 3	

Objectives:

- To introduce the cloud computing concepts and map reduce programming model.
- To provide skills and knowledge about operations and management in cloud technologies so as to implement large scale systems.
- To provide skills to design suitable cloud infrastructure that meets the business services and customer needs.
- To provide Knowledge of different CPU, memory and I/O virtualization techniques that serve in offering software, computation and storage services on the cloud; Software Defined Networks (SDN) and Software Defined Storage (SDS); cloud storage technologies and relevant distributed file systems, NoSQL databases and object storage;
- To introduce the variety of programming models and develop working experience in several of them.

Outcomes:

- Understand the evolution, principles, and benefits of Cloud Computing in order to assess existing cloud infrastructures to choose an appropriate architecture that meets business needs.
- Decide a suitable model to capture the business needs by interpreting different service delivery and deployment models.
- Understand virtualization foundations to cater the needs of elasticity, portability and resilience by cloud service providers.
- Infer architectural style, work flow of real-world applications and to implement the cloud applications using map reduce programming models.
- Compare operation and economic models of various trending cloud platforms prevailing in IT industry.

Unit I: Foundations of cloud: Inception and need for cloud computing: Motivations from distributed computing predecessors - Evolution - Characteristics - Business Benefits – Challenges in cloud computing - Exploring the Cloud Computing Stack - Fundamental Cloud Architectures – Advanced Cloud Architectures - Specialized Cloud Architectures

Unit II: Service Delivery and Deployment Models: Service Models (XaaS): Infrastructure as a Service (IaaS) - Platform as a Service (PaaS) - Software as a Service(SaaS) -

Deployment Models: Types of cloud - Public cloud - Private cloud - Hybrid cloud – Service level agreements - Types of SLA – Lifecycle of SLA- SLA Management

Unit III: Cloud Resource Virtualization: Virtualization as Foundation of Cloud – Understanding Hypervisors – Understanding Machine Image and Instances - Managing Instances – Virtual Machine Provisioning and Service Migrations Cloud Computing Applications and Paradigms: Existing Cloud Applications and Opportunities for New

Applications - Architectural Styles for Cloud Applications - Workflows: Coordination of Multiple Activities - Coordination Based on a State Machine Model: The ZooKeeper - The MapReduce Programming Model - A Case Study: The Grep The Web Application

Unit IV: Resource Management and Scheduling in Cloud: Policies and Mechanisms for Resource Management – Stability of a Two-Level Resource Allocation Architecture-Feedback Control Based on Dynamic Thresholds - Coordination of Specialized Autonomic Performance Managers - A Utility-Based Model for Cloud-Based Web Services - Resource Bundling: Combinatorial Auctions for Cloud Resources – Scheduling Algorithms for Computing Clouds - Resource Management and Dynamic Application Scaling

Unit V: Cloud Platforms and Application Development: Comparing Amazon web services, Google AppEngine, Microsoft Azure from the perspective of architecture (Compute, Storage Communication) services and cost models. Cloud application development using third party APIs, Working with EC2 API – Google App Engine API - Facebook API, Twitter API. Advances is Cloud: Media Clouds - Security Clouds - Computing Clouds - Mobile Clouds – Federated Clouds – Hybrid Clouds

Recommended Texts:

1. Rajkumar Buyya, James Broberg, Andrzej, M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 1st Edition, 2013.
2. Sosinsk, Barrie, Cloud Computing Bible, John Wiley & Sons, 1st Edition, 2011.

Reference Books:

1. Marinescu, Dan C. Cloud Computing: Theory and Practice. Morgan Kaufmann, 2017.
2. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing: A Practical Approach, Mc Graw Hill Education, 1st Edition, 2017.
3. Buyya, Rajkumar, Christian Vecchiola, and S. Thamarai Selvi. Mastering Cloud Computing: Foundations and Applications Programming, Tata Mcgraw Hill, 1st Edition, 2017.

Web References:

1. https://www.youtube.com/watch?v=-8O32k26RWA

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	M	S	L	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Internet of things		
Elective -1	I Year - II Semester	Credit: 3	

Objectives:

- To understand Smart Objects and IoT Architectures
- To learn about various IOT-related protocols
- To build simple IoT Systems using Arduino and Raspberry Pi.
- To understand data analytics and cloud in the context of IoT
- To develop IoT infrastructure for popular applications

Outcomes:

- Explain the concept of IoT.
- Analyze various protocols for IoT.
- Design a PoC of an IoT system using Raspberry Pi/Arduino
- Apply data analytics and use cloud offerings related to IoT.
- Analyze applications of IoT in real time scenario

Unit - I Introduction: Internet Layers - Protocols - Packets - Services - Performance parameters – Peer to peer networks - Sensor networks - Multimedia - IOT Definitions and Functional Requirements – Motivation – Architecture - Web 3.0 View of IoT– Ubiquitous IoT Applications – Four Pillars of IoT – DNA of IoT - The Toolkit Approach for End-user Participation in the Internet of Things. Middleware for IoT: Overview – Communication middleware for IoT –IoT Information Security.

Unit - II IoT protocols: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and

RFID Protocols – Issues with IoT Standardization – Unified Data Standards – Protocols – IEEE 802.15.4 – BACNet Protocol – point-to-point protocols - Ethernet protocols -

cellular Internet access protocol - Machine-to-machine protocol - Modbus – KNX – Zigbee Architecture – Network layer – APS layer – Security.

Unit - III Web of Things: Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals and Business Intelligence. Cloud of Things: Grid/SOA and Cloud Computing – Cloud Middleware – Cloud Standards – Cloud Providers and Systems – Mobile Cloud Computing – The Cloud of Things Architecture.

Unit - IV Integrating IOT: Integrated Billing Solutions in the Internet of Things Business Models for the Internet of Things - Network Dynamics: Population Models – Information Cascades - Network Effects - Network Dynamics: Structural Models - Cascading Behavior in Networks - The Small World Phenomenon.

Unit - V Applications: The Role of the Internet of Things for Increased Autonomy and Agility in Collaborative Production Environments - Resource Management in the Internet of Things: Clustering, Synchronisation and Software Agents. Applications - Smart Grid – Electrical Vehicle Charging - Case studies: Sensor body-area-network and Control of a smart home.

Recommended Texts:

1. Honbo Zhou;The Internet of Things in the Cloud:A Middleware Perspective-CRC Press 2012.
2. Architecting the Internet of Things - Dieter Uckelmann; Mark Harrison; Florian Michahelles- (Eds.) – Springer – 2011

Reference Books:

1. David Easley and Jon Kleinberg; Networks, Crowds, and Markets: Reasoning About a Highly Connected World ; Cambridge University Press - 2010.
2. Olivier Hersent, Omar Elloumi and David Boswarthick ; The Internet of Things: Applications to the Smart Grid and Building Automation ; Wiley 2012
3. Olivier Hersent, David Boswarthick, Omar Elloumi , "The Internet of Things – Key applications and Protocols", Wiley, 2012

Web References:

1. https://www.youtube.com/watch?v=b7GC4Zr74M0

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Data Analytics		
Elective -1	I Year - II Semester	Credit: 3	

Objectives:

- To gives an overview of Big Data, i.e. storage, retrieval and processing of big data.
- To focus on the "technologies", i.e., the tools/algorithms that are available for storage, processing of Big Data.
- To help a student to perform a variety of "analytics" on different data sets and to arrive at positive conclusions.
- To introduce the tools required to manage and analyze big data like Hadoop,NoSql MapReduce.
- To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability

Outcomes:

- Understand Big Data and its analytics in the real world
- Analyze the Big Data framework like Hadoop and NOSQL to efficiently store and process Big Data to generate analytics
- Design of Algorithms to solve Data Intensive Problems using Map Reduce Paradigm
- Design and Implementation of Big Data Analytics using pig and spark to solve data intensive problems and to generate analytics
- To have skills that will help them to solve complex real-world problems in for decision support.

UNIT – I: ESSENTIALS OF BIG DATA AND ANALYTICS: Data, Characteristics of data and Types of digital data, Sources of data, Working with unstructured data, Evolution and Definition of big data, Characteristics and Need of big data, Challenges of big data; Overview of business intelligence, Data science and Analytics, Meaning and Characteristics of big data analytics, Need of big data analytics, Classification of analytics, Challenges to big data analytics, Importance of big data analytics, Basic terminologies in big data environment.

UNIT –II: HADOOP: Introducing Hadoop, Need of Hadoop, limitations of RDBMS, RDBMS versus Hadoop, Distributed computing challenges, History of Hadoop, Hadoop overview, Use case of Hadoop, Hadoop distributors, HDFS (Hadoop Distributed File System) , Processing data with Hadoop, Managing resources and applications with Hadoop YARN (Yet another Resource Negotiator), Interacting with Hadoop Ecosystem.

UNIT – III: MAPREDUCE PROGRAMMING: Introduction, Mapper, Reducer, Combiner, Partitioner, Searching, Sorting, Compression, Real time applications using MapReduce, Data serialization and Working with common serialization formats, Big data serialization formats.

UNIT – IV: HIVE: Introduction to Hive, Hive architecture, Hive data types, Hive file format, Hive Query Language (HQL), User-Defined Function (UDF) in Hive;

UNIT – V: PIG: The anatomy of Pig , Pig on Hadoop, Pig Philosophy, Use case for Pig; ETL Processing , Pig Latin overview , Data types in Pig , Running Pig , Execution modes of Pig, HDFS commands, Relational operators, Piggy Bank , Word count example using Pig.

Recommended Texts:

1. Seema Acharya, Subhashini Chellappan, "Big Data Analytics", 2nd Edition, Wiley, 2019.

Reference Books:

1. Boris Lublinsky, Kevin T. Smith, Alexey Yakubovich, "Professional Hadoop Solutions",1st Edition, Wrox, 2013.
2. Chris Eaton, Dirk Deroos et. al., "Understanding Big data", Indian Edition, McGraw Hill, 2015.
3. Tom White, "HADOOP: The definitive Guide", 3rd Edition, O Reilly, 2012.
4. Vignesh Prajapati, "Big Data Analytics with R and Hadoop", 1st Edition, Packet Publishing Limited, 2013.

Web References:

1. https://www.youtube.com/watch?v=xvEKQefqQ7A

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Full Stack Web Development		
Elective -2	I Year - II Semester	Credit: 3	

Objectives:

- To provide knowledge and abilities to develop web sites for the internet
- To provide basic design principles to present ideas, information, products, and services on websites
- To induce basic programming principles to the construction of websites
- To make effectively manage website projects using available resources
- To inculcate full front end website architecture Knowledge

Outcomes:

- Design user interactions on web pages
- Develop back end website applications
- Create servers and databases for functionality
- Develop adaptive content for multiple devices (cell phone, tablets, etc.) Ensure cross-platform optimization for mobile phones
- Design and develop Application Programming Interfaces (APIs)

Unit I: Introduction to Dynamic Web Content-Three-tier architecture-architecture for client-server applications-Introduction to HTML5-Structural Elements-Paving the Way for Web Applications:HTML5 forms-The HTML5 Canvas-Audio and Video

Unit II: Introduction to CSS-Advanced CSS with CSS3-Accessing CSS from JavaScript-Exploring JavaScript-Expressions and Control Flow in JavaScript-JavaScript Functions, Objects, and Arrays.

Unit III: Introduction to PHP. -Expressions and Control Flow in PHP. -PHP Functions and Objects-PHP Arrays-Cookies, Sessions, and Authentication-Accessing MySQL Using PHP

Unit IV: Introduction to AngularJS, -AngularJS - Overview- Environment Setup- MVC Architecture- Directives- Expressions- Controllers – Filters - Tables

Unit V: HTML DOM – Modules – Forms – Includes – AJAX – Views – Scopes – Services -Dependency Injection - Custom Directives - Fetching Data From a PHP Server Running MySQL

Recommended Texts:

1. Robin Nixon; Learning PHP, MySQL, JavaScript, CSS & HTML5; O'Reilly; Fourth edition;2015
2. Ken Williamson; Learning AngularJS; O'Reilly Media;2015

Reference Books:

1. Full Stack AngularJS for Java Developers: Build a Full-Featured Web Application from Scratch Using AngularJS with Spring RESTful, Ravi Kant Soni, Apress; 1st ed. edition (4 December 2017)

Web References:

1. https://www.alvinisd.net/cms/lib03/TX01001897/Centricity/Domain/1077/beginning_html5_and_css3.pdf
2. https://www.tutorialspoint.com/angularjs/index.htm
3. https://www.youtube.com/watch?v=9b9pLgaSQuI

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Natural Language Processing		
Elective -2	I Year - II Semester	Credit: 3	

Objectives:

- To learn the fundamentals of natural language processing
- To understand the use of CFG and PCFG in NLP
- To understand the role of semantics of sentences and pragmatics
- To apply the NLP techniques to IR applications

Outcomes:

- To tag a given text with basic Language features
- To design an innovative application using NLP components
- To implement a rule-based system to tackle morphology/syntax of a language
- To design a tag set to be used for statistical processing for real-time applications
- To compare and contrast the use of different statistical approaches for different types of NLP applications.

UNIT I INTRODUCTION: Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM - Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance

UNIT II WORD LEVEL ANALYSIS: Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Part-of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models.

UNIT III SYNTACTIC ANALYSIS: Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing – Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs - Feature structures, Unification of feature structures.

UNIT IV SEMANTICS AND PRAGMATICS: Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selection restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.

UNIT V DISCOURSE ANALYSIS AND LEXICAL RESOURCES: Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill's Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC).

Recommended Texts:

1. Daniel Jurafsky, James H. Martin;Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech; Pearson Publication; 2014.
2. Steven Bird, Ewan Klein and Edward Loper, ―Natural Language Processing with Python , First Edition, OReilly Media, 2009.

Reference Books:

1. Breck Baldwin, ―Language Processing with Java and LingPipe Cookbook, Atlantic Publisher, 2015.
2. Richard M Reese, ―Natural Language Processing with Java , O_Reilly Media, 2015.
3. Nitin Indurkhya and Fred J. Damerau, ―Handbook of Natural Language Processing, Second Edition, Chapman and Hall/CRC Press, 2010.
4. Tanveer Siddiqui, U.S. Tiwary, ―Natural Language Processing and Information Retrieval, Oxford University Press, 2008.

Web References:

1. https://www.youtube.com/watch?v=oWsMIW-5xUc&list=PLLssT5z_DsK8HbD2sPcUIDfQ7zmBarMYv

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Digital Image Processing		
Elective -2	I Year - II Semester	Credit: 3	

Objectives:

- To become familiar with digital image fundamentals
- To get exposed to simple image enhancement techniques in Spatial and Frequency domain.
- To learn concepts of degradation function and restoration techniques.
- To study the image segmentation and representation techniques.
- To become familiar with image compression and recognition methods

Outcomes:

- Know and understand the basics and fundamentals of digital image processing, such as digitization, sampling, quantization, and 2D-transforms.
- Operate on images using the techniques of smoothing, sharpening and enhancement.
- Perform the restoration concepts and filtering techniques.
- Demonstrate the segmentation, features extraction, compression and recognition methods for color models.
- Compress images and use tools for image recognition.

UNIT I DIGITAL IMAGE FUNDAMENTALS: Steps in Digital Image Processing – Components – Elements of Visual Perception – Image Sensing and Acquisition – Image Sampling and Quantization – Relationships between pixels - Color image fundamentals - RGB, HSI models, Two-dimensional mathematical preliminaries, 2D transforms - DFT, DCT.

UNIT II IMAGE ENHANCEMENT: Spatial Domain: Gray level transformations – Histogram processing – Basics of Spatial Filtering– Smoothing and Sharpening Spatial Filtering, Frequency Domain: Introduction to Fourier Transform– Smoothing and Sharpening frequency domain filters – Ideal, Butterworth and Gaussian filters, Homomorphic filtering, Color image enhancement.

UNIT III IMAGE RESTORATION: Image Restoration - degradation model, Properties, Noise models – Mean Filters – Order Statistics – Adaptive filters – Band reject Filters – Band pass Filters – Notch Filters – Optimum Notch Filtering – Inverse Filtering – Wiener filtering

UNIT IV IMAGE SEGMENTATION: Edge detection, Edge linking via Hough transform – Thresholding - Region based segmentation – Region growing – Region splitting and merging – Morphological processing- erosion and dilation, Segmentation by morphological watersheds – basic concepts – Dam construction – Watershed segmentation algorithm.

UNIT V IMAGE COMPRESSION AND RECOGNITION: Need for data compression, Huffman, Run Length Encoding, Shift codes, Arithmetic coding, JPEG standard, MPEG. Boundary representation, Boundary description, Fourier Descriptor, Regional Descriptors – Topological feature, Texture - Patterns and Pattern classes - Recognition based on matching.

Recommended Texts:

1. Rafael C. Gonzalez, Richard E. Woods; Digital Image Processing; Pearson, Fourth Edition;2018.
2. Anil K. Jain; Fundamentals of Digital Image Processing; Pearson;2015.

Reference Books:

1. Kenneth R. Castleman, Digital Image Processing ‘, Pearson, 2006.
2. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,; Digital Image Processing using MATLAB ; Pearson Education, Inc., 2011.
3. D E. Dudgeon and RM. Mersereau; Multidimensional Digital Signal Processing; Prentice Hall; 1990.
4. William K. Pratt; Digital Image Processing ; John Wiley; 2002
5. Milan Sonka et al; Image processing, analysis and machine vision; Brookes/Cole, Vikas Publishing House; 2nd edition; 1999.

Web References:

1. https://www.youtube.com/watch?v=DSGHkvQBMbs&list=PLuv3GM6-gsE08DuaC6pFUvFaDZ7EnWGX8

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Parallel and Distributed Computing		
Core - 10	II Year - III Semester	Credit: 4	

Objectives:

- To analyze the current popular distributed systems such as peer-to-peer (P2P) systems will also be analyzed.
- To learn and apply knowledge of parallel and distributed computing techniques and methodologies
- To learn the architecture and parallel programming in graphics processing units (GPUs).
- To understand the memory hierarchy and cost-performance tradeoffs.
- To gain experience in the design, development, and performance analysis of parallel and distributed applications

Outcomes:

- Develop and apply knowledge of parallel and distributed computing techniques and methodologies.
- Apply design, development, and performance analysis of parallel and distributed applications.
- Use the application of fundamental Computer Science methods and algorithms in the development of parallel applications.
- Explain the design, testing, and performance analysis of a software system, and to be able to communicate that design to others.
- Understand the requirements for programming parallel systems and how they can be used to facilitate the programming of concurrent systems.

UNIT-I Introduction to Parallel Computing: The Idea of Parallelism, Power and potential of parallelism, examining sequential and parallel programs, Scope and issues of parallel and distributed computing, Goals of parallelism, Parallelism and concurrency using multiple instruction streams.

UNIT-II Parallel Architecture: Pipeline architecture, Array processor, Multi-processor architecture, Systolic architecture, Dataflow architecture, Architectural classification schemes, Memory access classification, Memory Issues: Shared vs. distributed, Symmetric multiprocessing (SMP), SIMD, Vector processing, GPU co-processing, Flynn's Taxonomy, Instruction Level support for parallel programming, Multiprocessor caches and Cache Coherence, Non-Uniform Memory Access (NUMA).

UNIT-III Parallel Algorithm Design Principles and Programming: Need for communication and coordination/synchronization, Scheduling and contention, Independence and partitioning, Task- Based Decomposition, Data Parallel Decomposition, Characteristics of task and interaction, Load balancing, Data Management, parallel algorithm models, Sources of overhead in parallel programs, Performance metrics for parallel algorithm implementations, Parallel algorithmic patterns like divide and conquer, Map and Reduce, Specific algorithms like parallel Merge Sort, Parallel graph Algorithms.

UNIT-IV : Architectures Of Distributed Systems - Architectural Styles - System

Architectures - Architectures Versus Middleware - Self-Management In Distributed Systems - Processes - Threads - Virtualization - Clients -Servers - Communication - Remote Procedure Call - Message-Oriented Communication - Stream-Oriented Communication - Multicast Communication

UNIT-V : Distributed Object Based Systems - Architecture - Processes - Communication - Naming - Synchronization - Fault Tolerance - Security - Distributed System Examples - File Systems And Web Based Systems

Recommended Texts:

1. Ananth Grama, Anshul Gupta, and George Karypis, Vipin Kumar; Introduction to Parallel Computing; Addition Wesley; 2nd Edition;2003
2. A.S. Tanenbaum; Distributed Operating Systems; Create Space Independent Publishing Platform; 3rd edition;2017

Reference Books:

1. Introduction To Parallel Programming, Steven Brawer, Academic Press
2. Introduction To Parallel Processing, M. Sasikumar, Dinesh Shikhare and P. Ravi Prakash,PHI
3. Randy Chow, T. Johnson, Distributed Operating Systems and Algorithms, Addison Wesley
4. Ian Foster: Designing and Building Parallel Programs – Concepts and tools for Parallel Software Engineering, Pearson Publisher, 1st Edition, 2019.
5. Parallel Programming in C with MPI and OpenMP Michael J. Quinn, McGrawHill Higher Education

Web References:

1. https://www.youtube.com/watch?v=qbQCQ0U6H0o&list=PLbMVogVj5nJQRvzENlvMKA9q70ScSRZBQ

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	S	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Deep Learning and Neural Networks		
Core - 11	II Year - III Semester	Credit: 4	

Objectives:

- To understand the theoretical foundations, algorithms and methodologies of Neural Network
- To design and develop an application using specific deep learning models
- To provide practical knowledge in handling and analyzing real world applications.
- To recognize the characteristics of deep learning models that are useful to solve real-world problems.
- To introduce Various paradigms of earning problems, Perspectives and Issues in deep learning framework, review of fundamental learning techniques.

Outcomes:

- Understand different methodologies to create applications using deep nets.
- Identify and apply appropriate deep learning algorithms for analyzing the data for a variety of problems.
- Implement different deep learning algorithms
- Design the test procedures to assess the efficacy of the developed model.
- Combine several models in to gain better results

Unit I: Basics of artificial neural networks (ANN): Artificial neurons, Computational models of neurons, Structure of neural networks, Functional units of ANN for pattern recognition tasks

Feedforward neural networks: Pattern classification using perceptron, Multilayer feedforward neural networks (MLFFNNs), Backpropagation learning, Empirical risk minimization, Regularization, Autoencoders

Unit II: Deep neural networks (DNNs): Difficulty of training DNNs, Greedy layer wise training, Optimization for training DNNs, Newer optimization methods for neural networks (AdaGrad, RMSProp, Adam), Second order methods for training, Regularization methods (dropout, drop connect, batch normalization)

Unit III: Convolution neural networks (CNNs): Introduction to CNNs – convolution, pooling, Deep CNNs, Different deep CNN architectures – LeNet, AlexNet, VGG, PlacesNet, training a CNNs: weights initialization, batch normalization, hyperparameter optimization, Understanding and visualizing CNNs.

Unit IV: Recurrent neural networks (RNNs): Sequence modeling using RNNs, Backpropagation through time, Long Short Term Memory (LSTM), Bidirectional LSTMs, Bidirectional RNNs, Gated RNN Architecture - Generative models: Restricted Boltzmann Machines (RBMs), Stacking RBMs, Belief nets.

Unit V: Learning sigmoid belief nets, Deep belief nets Under complete - Auto encoder, Regularized Auto encoder, stochastic Encoders and Decoders, Contractive Encoders.

Applications: Applications in vision, speech and natural language processing

Recommended Texts:

1. S. Haykin, Neural Networks and Learning Machines , Prentice Hall of India, 2016
2. Ian Goodfellow, Yoshua Bengio and Aaron Courville, " Deep Learning", MIT Press, 2017

Reference Books:

1. Satish Kumar, Neural Networks - A ClassRoom
2. B. Yegnanarayana, Artificial Neural Networks, Prentice- Hall of India, 1999
3. Giancarlo Zaccone, Md. RezaulKarim, Ahmed Menshawy "Deep Learning with TensorFlow: Explore neural networks with Python", Packt Publisher, 2017.
4. Antonio Gulli, Sujit Pal "Deep Learning with Keras", Packt Publishers, 2017.
5. Francois Chollet "Deep Learning with Python", Manning Publications, 2017.

Web References:

1. https://www.youtube.com/watch?v=aPfkYu_qiF4&list=PLEAYkSg4uSQ1r-2XrJ_GBzzS6I-f8yfRU

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	S	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	L	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Cryptography		
Core - 12	II Year - III Semester	Credit: 4	

Objectives:

- To learn the emerging concepts of cryptography and algorithms
- To defend the security attacks on information systems using secure algorithms and Authentication process
- To categorize and analyze the key concepts in network and wireless security
- To Infer the need of security to introduce strong cryptosystems.

Outcomes:

- Analyze the cryptographic algorithms for information security.
- Identify the authentication schemes for membership authorization.
- Identify computer and network security threats, classify the threats and develop a security model for detect and mitigate the attacks.
- Identify the requirements for secure communication and challenges related to the secure web services
- Ability to identify the need of ethical and professional practices, risk management Using emerging security solutions.

Unit I: Introduction and Symmetric Key Cryptographic Systems : Introduction to Cryptography, Types of Attacks, Symmetric Key Cryptography, Data Encryption Standard (DES), Differential and Linear cryptanalysis, Advanced Encryption Standard(AES), Modes of operation, Stream Ciphers: Feedback shift registers, Stream ciphers based on LFSRs.

Unit II: Asymmetric Key Cryptosystems: Applications of asymmetric Cryptosystems – RSA Rabin, Elgamal, Probabilistic Cryptosystems, Elliptic Curve Cryptography (ECC), Diffie-Hellman key exchange protocol, Chinese Remainder Theorem (CRT).

Unit III: Data Integrity and Authentication: Message Authentication Code (MAC), Hash function properties, General model for iterated hash functions -MD5, Secure Hash algorithms, HMAC, Attacks on hash functions.

Unit IV: Digital Signature algorithm, Public key infrastructure: X. 509 digital certificate, Kerberos, Zero-Knowledge Protocol.

Unit V: Advanced Cryptographic Techniques: Multiparty Computation and Secret Sharing, Introduction - Indistinguishability - Secret - Sharing Simulation - Based Security-Security against Active Corruption-BGW Protocol (Active, Honest Majority)- Homomorphic Encryption-Lattice Cryptography

Recommended Texts:

1. J. Katz and Y. Lindell, Introduction to Modern Cryptography. Chapman & Hall/CRC Press, 2014
2. W. Stallings, Cryptography and Network Security: Principles and Practice, 7th Ed. Pearson Publishers, 2017.

3. C. Paar and J Pelzl, Understanding Cryptography, Springer, 2010
4. Behrouz A. Forouzan, Cryptography and Network Security:6th Ed. McGraw-Hill,2017
5. Dan Boneh and Victor Shoup, A Graduate Course in Applied Cryptography, Jan 2020

Reference Books:

1. Kaufman, Perlman and Speciner. Network Security: Private Communication in a Public World., 2 nd edition,2002 , Pearson Publishers (ISBN No.:978-01-3-04601-96)
2. Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone,Handbook of Applied Cryptography,5th edition,2001,CRC Press,(ISBN No:0-8493-8523-7)
3. D. R. Stinson, Cryptography: Theory and Practice, 3 rd Ed. Boca Raton, FL: Chapman &Hall/CRC, 2005. (ISBN No.:978-1-58-488508-5)J. H. Silverman, A Friendly
4. Introduction to Number Theory, 4th Ed. Boston: Pearson, 2012. (ISBN No.:978-0-321- 81619-1)
5. Ronald Cramer, Ivan BjerreDamgård, JesperBuus Nielsen, "Secure MultipartyComputation and Secret Sharing", ISBN 9781107043053, Cambridge University Press, 2015
6. Philip N. Klein, "A Cryptography Primer-Secrets and Promises", ISBN 9781107603455, Cambridge University Press, 2014

Web References:

1. https://www.youtube.com/watch?v=iTVyKbDCJrA&list=PLgMDNELGJ1CbdGLyn7OrVAP-IKg-0q2U2

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	M	S	M	L	M	L	S
CO2	S	S	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	L	S	L	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Practical - 5: Deep learning Lab		
Core - 13	II Year - III Semester	Credit: 2	

Objectives:

- To provide the practical knowledge in handling and analysing real world applications.
- To recognize the characteristics of deep learning models that are useful to solve real-world problems.
- To introduce Various paradigms of earning problems, Perspectives and Issues in deep learning framework, review of fundamental learning techniques.

Outcomes:

- Understand different methodologies to create application using deep nets.
- Identify and apply appropriate deep learning algorithms for analyzing the data for variety of problems.
- Implement different deep learning algorithms
- Design the test procedures to assess the efficacy of the developed model.
- Combine several models in to gain better results

1. Write a program to generate XOR function using McCulloch-Pitts neuron and appropriate values for weights, bias and threshold.
2. Write a program for perceptron net for an AND function with bipolar inputs and targets.
3. Write a program to recognize the number from 0, 1, 2, 3, . . . , 9. A number is represented as a 5 × 3 matrix of 0 and 1. For any valid point it is taken as 1 and invalid point it is taken as 0. The net has to be trained to recognize all the numbers and when the test data is given. The file numbers. Mat has three components, as: input-data: The training data. Each column represents a number. output-data: A 10 × 10 matrix of desired outputs. test-data: Test data. Each column is a test vector.
4. Write a program (with a suitable example) to demonstrate how the hyperplane is changing in different iterations using the perceptron learning law with its decision regions. Give the output in graphical form.
5. Write a program to compress the data given in the data file alphabet. Mat using a multilayer feedforward neural network and back propagation. An alphabet is represented by a 9×7 matrix. Check the performance with different values of the learning rate parameter, and momentum factor. (Hint: The training input vector and the target output vectors are the same).

Recommended Texts:

1. Dr. S Lovelyn Rose, Dr. L Ashok Kumar, Dr. D Karthika Renuka;Deep Learning Using Python Wiley;Edition: 1, 2019

Reference Books:

1. Francois Chollet; Deep Learning with Python Paperback;Manning;2017

Web References:

1. https://www.youtube.com/watch?v=H-fzcFoBq44&list=PLkhxeo3AyR-xrggjA0DgWpbWZMTwBy3dV

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	S	M	L	M	L	S
CO2	S	M	S	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	M	M	S	L	M	L	M	S	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Project & Viva-Voce		
Core- 14	II Year & IV Semester	Credit: 20	

Objectives:

- To make the project an extended piece of individual work.
- To work on a topic that interests the student
- To have regular meetings with their supervisor and/or external project provider to discuss progress
- To produce dissertations that contain some element of original work.
- To encourage and reward individual inventiveness and application of effort

Outcomes:

- Construct a project from initial ideas;
- Plan, schedule, monitor and control their own work;
- Defend their ideas in discussions and presentations;
- Use libraries and other information resources;
- Apply tools and techniques from taught courses
- Communicate their findings through a written report.

Project: The project work is to be carried out either in a software industry or in an academic institution for the entire semester and the report of work done is to be submitted to the University.

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	S	M	L	M	L	S
CO2	S	M	S	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	S	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Cyber Security		
Elective - 3	II Year - III Semester	Credit: 3	

Objectives:

- To understand various types of cyber-attacks and cyber-crimes
- To learn threats and risks within context of the cyber security
- To have an overview of the cyber laws & concepts of cyber forensics
- To study the defensive techniques against these attacks
- To describe various legal responses to cybercrime

Outcomes:

- Analyze cyber-attacks, types of cybercrimes, cyber laws and also how to protect them self and ultimately the entire Internet community from such attacks.
- Interpret and forensically investigate security incidents
- Apply policies and procedures to manage Privacy issues
- Design and develop secure software modules
- Understand different forms of hacking techniques

UNIT -I Introduction to Cyber Security: Basic Cyber Security Concepts, layers of security, Vulnerability, threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, etc., Comprehensive Cyber Security Policy.

UNIT - II Cyberspace and the Law & Cyber Forensics: Introduction, Cyber Security Regulations, Roles of International Law. The INDIAN Cyberspace, National Cyber Security Policy. Introduction, Historical background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Forensics Investigation, Challenges in Computer Forensics

UNIT - III Cybercrime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Organizational security Policies and Measures in Mobile Computing Era, Laptops.

UNIT- IV Cyber Security: Organizational Implications: Introduction, cost of cybercrimes and IPR issues, web threats for organizations, security and privacy implications, social media marketing: security risks and perils for organizations, social computing and the associated challenges for organizations

UNIT - V Privacy Issues: Basic Data Privacy Concepts: Fundamental Concepts, Data Privacy Attacks, Datalinking and profiling, privacy policies and their specifications, privacy policy languages, privacy in different domains- medical, financial, etc Cybercrime: Case study on recent threats and attacks.

Recommended Texts:

1. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley

Reference Books:

1. B.B. Gupta, D.P. Agrawal, Haoxiang Wang, Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives, CRC Press, ISBN 9780815371335,2018.
2. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson,CRC Press.
3. Introduction to Cyber Security, Chwan-Hwa(john) Wu,J. David Irwin, CRC Press T&F Group.

Web References:

1. https://www.youtube.com/watch?v=6wi5DI6du-4&list=PL_uaeekrhGzJlB8XQBxU3z_hDwT95xlk

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	S	M	L	M	L	S
CO2	S	M	S	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	S	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Advanced Computer Architecture		
Elective - 3	II Year - III Semester	Credit: 3	

Objectives:

- To introduce the students to the recent trends in the field of Computer Architecture and identify performance related parameters.
- To learn the different multiprocessor issues.
- To expose the different types of multicore architectures.
- To understand the design of the memory hierarchy.

Outcomes:

- Identify the limitations of Instruction-level parallelism (ILP).
- Discuss the issues related to multiprocessing and suggest solutions
- Point out the salient features of different multicore architectures and how they exploit parallelism.
- Discuss the various techniques used for optimizing the cache performance
- Design hierarchical memory system
- Point out how data level parallelism is exploited in architectures

UNIT I FUNDAMENTALS OF COMPUTER DESIGN AND ILP

Fundamentals of Computer Design – Measuring and Reporting Performance – Instruction Level Parallelism and its Exploitation – Concepts and Challenges –Exposing ILP - Advanced Branch Prediction - Dynamic Scheduling - Hardware-Based Speculation - Exploiting ILP - Instruction Delivery and Speculation - Limitations of ILP - Multithreading

UNIT II MEMORY HIERARCHY DESIGN

Introduction – Optimizations of Cache Performance – Memory Technology and Optimizations – Protection: Virtual Memory and Virtual Machines – Design of Memory Hierarchies – Case Studies.

UNIT III MULTIPROCESSOR ISSUES

Introduction- Centralized, Symmetric and Distributed Shared Memory Architectures – Cache Coherence Issues – Performance Issues – Synchronization – Models of Memory Consistency – Case Study-Interconnection Networks – Buses, Crossbar and Multi-stage Interconnection Networks

UNIT IV MULTICORE ARCHITECTURES

Homogeneous and Heterogeneous Multi-core Architectures – Intel Multicore Architectures – SUN CMP architecture – IBM Cell Architecture. Introduction to Warehouse-scale computers Architectures- Physical Infrastructure and Costs- Cloud Computing –Case Study- Google Warehouse-Scale Computer.

UNIT V VECTOR, SIMD AND GPU ARCHITECTURES

Introduction-Vector Architecture – SIMD Extensions for Multimedia – Graphics Processing Units – Case Studies – GPGPU Computing – Detecting and Enhancing Loop Level Parallelism-Case Studies.

Recommended Texts:

1. Darryl Gove ;Multicore Application Programming: For Windows, Linux, and Oracle Solaris, Pearson, 2011

Reference Books:

1. David B. Kirk, Wen-mei W. Hwu, ―Programming Massively Parallel Processors, Morgan Kauffman, 2010
2. David E. Culler, Jaswinder Pal Singh, ―Parallel computing architecture : A hardware/software approach , Morgan Kaufmann /Elsevier Publishers, 1999
3. John L. Hennessey and David A. Patterson, ―Computer Architecture – A Quantitative Approach, Morgan Kaufmann / Elsevier, 5th edition, 2012.
4. Kai Hwang and Zhi.Wei Xu, ―Scalable Parallel Computing, Tata McGraw Hill, NewDelhi, 2003

Web References:

1. https://www.youtube.com/watch?v=v7iefsovo9M&list=PLwdnzlV3ogoWJhBxBYu-K4l-q-nNHd24D

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	S	M	L	M	L	S
CO2	S	M	S	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	S	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Distributed Database Systems		
Elective - 3	II Year - III Semester	Credit: 3	

Objectives:

- To introduction students to Distributed DBMS and associated problems.
- To make students understand various algorithms and techniques for managing distributed database.
- To understand theoretical and practical aspects of distributed database systems.
- To study and identify various issues related to the development of distributed database system.
- To make students understand Transaction Management & Compare various approaches to concurrency control in Distributed database

Outcomes:

- Apply various fragmentation techniques given a problem
- Analyse and calculate the cost of enforcing semantic integrity control
- Use the steps of query processing
- Apply optimization techniques are applies to Distributed Database
- Apply effectively Query Optimization Algorithms

Unit-I: Introduction: Distributed Data Processing, Distributed Database Systems, Promises of DDBSs, Complicating factors, Problem areas Overview of RDBMS: Concepts, Integrity, Normalization - Distributed DBMS Architecture: Autonomy, Distribution, Heterogeneity DDBMS Architecture – Client/Server, Peer to peer, MDBS

Unit-II: Data Distribution Alternatives: Design Alternatives – localized data, distributed data Fragmentation – Vertical, Horizontal (primary & derived), hybrid, general guidelines, correctness rules Distribution transparency – location, fragmentation, replication Impact of distribution on user queries – No Global Data Dictionary (GDD), GDD containing location information Example on fragmentation

Unit-III: Semantic Data Control: View Management, Authentication – database authentication, OS authentication, Access Rights, Semantic Integrity Control – Centralized & Distributed , Cost of enforcing semantic integrity - : Query Processing: Query Processing Problem, Layers of Query Processing Query Processing in Centralized Systems – Parsing & Translation, Optimization, Code generation, Example Query Processing in Distributed Systems – Mapping global query to local, Optimization,

Unit-IV: Optimization of Distributed Queries: Query Optimization, Centralized Query Optimization, Join Ordering Distributed Query Optimization Algorithms - Distributed Transaction Management & Concurrency Control: Transaction concept, ACID property, Objectives of transaction management, Types of transactions, Objectives of Distributed Concurrency Control, Concurrency Control anomalies, Methods of concurrency control, Serializability and recoverability, Distributed Serializability, Enhanced lock based and

timestamp based protocols, Multiple granularity, Multi version schemes, Optimistic Concurrency Control techniques

Unit-V: Distributed Deadlock & Recovery: Deadlock concept, Deadlock in Centralized systems, Deadlock in Distributed Systems – Detection, Prevention, Avoidance, Wait-Die Algorithm, Wound-Wait algorithm Recovery in DBMS - Types of Failure, Methods to control failure, Different techniques of recoverability, Write- Ahead logging Protocol, Advanced recovery techniques- Shadow Paging, Fuzzy checkpoint, ARIES, RAID levels, Two Phase and Three Phase commit protocols

Recommended Texts:

1. Ozsu; Principles of Distributed Database Systems; Springer; 4th edition;2020

Reference Books:

1. Rahimi & Haug; Distributed Database Management Systems;Wiley;2010
2. Distributed Database Systems, Chanda Ray, Pearson Publication
3. Sachin Deshpande; Distributed Databases; Dreamtech;2014

Web References:

1. https://www.youtube.com/watch?v=dlBVWMdGhqw&list=PLUJ7JmcrTifBROWODSG8wgyl20XgBuE-N

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	S	M	L	M	L	S
CO2	S	M	S	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	S	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Human Computer Interaction		
Elective - 4	II Year - III Semester	Credit: 3	

Objectives:

- To learn the foundations of Human Computer Interaction.
- To become familiar with the design technologies for individuals and persons with disabilities.
- To be aware of mobile HCI.
- To learn the guidelines for user interface.
- To encourage to design certain tools for blind or differently abled people

Outcomes:

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

UNIT I 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 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

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

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

Recommended Texts:

- Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale;Human Computer Interaction; Pearson Education; 3rd Edition; 2004
- Brian Fling;Mobile Design and Development;First Edition; O'Reilly Media Inc.;

2009

- Bill Scott and Theresa Neil;Designing Web Interfaces; First Edition; O'Reilly, 2009.

Reference Books:

- Designing the user interface. 3rd Edition Ben Shneidermann, Pearson Education Asia.
- Interaction Design Prece, Rogers, Sharps. Wiley Dreamtech.
- User Interface Design, Soren Lauesen , Pearson Education.
- Human –Computer Interaction, D. R. Olsen, Cengage Learning.
- Human –Computer Interaction, Smith - Atakan, Cengage Learning.

Web References:

- https://www.youtube.com/watch?v=q81KXc54Ozs&list=PLxtKZf9nLWO3d2a6M8l2BU8WTJKzHC4HJ

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	M	S	L	M	L	L
CO2	S	M	M	S	M	M	S	L	M	L
CO3	L	M	S	L	M	S	M	L	S	M
CO4	S	L	L	M	M	L	L	S	M	S
CO5	M	S	M	L	S	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Agile Software Engineering		
Elective - 4	II Year - III Semester	Credit: 3	

Objectives:

- To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software.
- To provide a good understanding of software design and a set of software technologies and APIs.
- To do a detailed examination and demonstration of Agile development and testing techniques.
- To understand the benefits and pitfalls of working in an Agile team.
- To understand Agile development and testing.

Outcomes:

- Upon completion of the course, the students will be able to:
- Realize the importance of interacting with business stakeholders in determining the requirements for a software system
- Perform iterative software development processes: how to plan them, how to execute them.
- Point out the impact of social aspects on software development success.
- Develop techniques and tools for improving team collaboration and software quality.
- Perform Software process improvement as an ongoing task for development teams.
- Show how agile approaches can be scaled up to the enterprise level.

UNIT I AGILE METHODOLOGY: Theories for Agile Management – Agile Software Development – Traditional Model vs. Agile Model - Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams - Agility in Design, Testing – Agile Documentations – Agile Drivers, Capabilities and Values

UNIT II AGILE PROCESSES: Lean Production - SCRUM, Crystal, Feature Driven Development- Adaptive Software Development - Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

UNIT III AGILITY AND KNOWLEDGE MANAGEMENT: Agile Information Systems – Agile Decision Making - Earl_ S Schools of KM – Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment, leveraging –

KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM).

UNIT IV AGILITY AND REQUIREMENTS ENGINEERING: Impact of Agile Processes in RE–Current Agile Practices – Variance – Overview of RE Using Agile – Managing Unstable Requirements – Requirements Elicitation – Agile Requirements Abstraction Model – Requirements Management in Agile Environment, Agile Requirements Prioritization – Agile Requirements Modeling and Generation – Concurrency in Agile Requirements Generation.

UNIT V AGILITY AND QUALITY ASSURANCE: Agile Product Development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile Approach to Quality Assurance - Test Driven Development – Agile Approach in Global Software Development.

Recommended Texts:

1. David J. Anderson and Eli Schragenheim; Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results; Prentice Hall; 2003.
2. Hazza and Dubinsky; Agile Software Engineering, Series: Undergraduate Topics in Computer Science ; Springer; 2009.

Reference Books:

1. Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.
2. Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Web References:

1. https://www.youtube.com/watch?v=x90kIAFGYKE&t=8s

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	M	S	L	M	L	L
CO2	S	M	M	S	M	M	S	L	M	L
CO3	L	M	S	L	M	S	M	L	S	M
CO4	S	L	L	M	M	L	L	S	M	S
CO5	M	S	M	L	S	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Computer vision		
Elective - 4	II Year - III Semester	Credit: 3	

Objectives:

- Understanding the Basics of Computer Vision.
- Acquiring skills to develop computer vision-based applications.To introduce students the fundamentals of image formation
- To introduce students the major ideas, methods, and techniques of computer vision and pattern recognition
- To develop an appreciation for various issues in the design of computer vision and object recognition systems
- To provide the student with programming experience from implementing computer vision and object recognition applications

Outcomes:

- Ability to understand the computer vision pipeline.
- Ability to build solutions using computer vision algorithms.
- Identify basic concepts, terminology, theories, models and methods in the field of computer vision
- Describe known principles of human visual system
- Describe basic methods of computer vision related to multi-scale representation, edge detection and detection of other primitives, stereo, motion and object recognition
- Suggest a design of a computer vision system for a specific problem

Unit I : Cameras - Pinhole Cameras - Cameras with Lenses - The Human Eye - Sensing Geometric Camera Models - Elements of Analytical Euclidean Geometry - Camera Parameters & Perspective projection - Affine Cameras and Affine Projection equations
Unit II : Geometric Camera Calibration - Least squares parameter estimation - A Linear Approach to Camera Calibration - Taking Radial Distortion into Account - Analytical Photogrammetry - Radiometry - Light in Space - Light at Surfaces -
Unit III : Sources, Shadows and shading - Qualitative Radiometry - Sources and Their Effects - Local Shading Model - Color- The Physics of Color - Human Color Perception - Representing Color - Surface Color from Image Color
Unit IV : Linear filters - Convolution - Shift Invariant Linear Systems - Spatial Frequency and Fourier Transforms- Sampling and Aliasing - Scale and Image Pyramids
Unit V : Edge detection - Noise - Detecting Edges - Texture - Representing Texture - Analysis (and Synthesis) Using Oriented Pyramids - Synthesizing Textures for Rendering - Shape from Texture for Planes

Recommended Texts:

1. D. Forsyth and J. Ponce; Computer Vision - A modern approach; Pearson India;2015

Reference Books:

1. Richard Szeliksy "Computer Vision: Algorithms and Applications" (http://szeliski.org/Book/)
2. Haralick& Shapiro, "Computer and Robot Vision", Vol II
3. G_erardMedioni and Sing Bing Kang "Emerging topics in computer vision"
4. Emanuele Trucco and AllessandroVerri "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998.
5. Olivier Faugeras, "Three-Dimensional Computer Vision", The MIT Press, 1993

Web References:

1. https://www.youtube.com/watch?v=3LaVxEX3F0o&list=PLwdnzlV3ogoVsma5GmBSsgJM6gHv1QoAo

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	M	S	M	S	L	M	L	L
CO2	S	M	M	S	M	M	S	L	M	L
CO3	L	M	S	L	M	S	M	L	S	M
CO4	S	L	L	M	M	L	L	S	M	S
CO5	M	S	M	L	S	M	M	L	M	L

S-Strong M-Medium L-Low

List of Soft Skill Courses: Syllabus

Title of the Course/Paper	Communication Skills for Software Engineers - I		
Soft Skill -	Year - Semester		Credit: 2

Objectives:

- Understand the need of current soft skills
- Generalize self development and implementation procedures
- Demonstrate narration skills
- Design simple comprehension with given requirements
- Develop implementations in latest technologies
- Demonstrate the applications with varied soft skills like debate, oration, tell about yourself etc.

Outcomes:

- Enumerate varied soft skills needed for employment
- Identify the lack in oneself and improve it
- Learn the current technical implementations
- Summarize the different requirements for employability
- Calculate self performance ,Generalize narration , oration and debate skills
- Conceptualize the representation of current technologies

1 Basics of Communication

1.1 Definition and process of communication

1.2 Types of communication - formal and informal, oral and written, verbal and non-verbal

1.3 Communications barriers and how to overcome them

1.4 Barriers to Communication, Tools of Communication

2 Application of Grammar

2.1 Parts of Speech (Noun, verb, adjective, adverb) and modals

2.2 Sentences and its types

2.3 Tenses

2.4 Active and Passive Voice

2.5 Punctuation

2.6 Direct and Indirect Speech

3 Reading Skill

Unseen passage for comprehension (one word substitution, prefixes, suffixes, antonyms, synonyms etc. based upon the passage to be covered under this topic)

4 Writing Skill

4.1 Picture composition

4.2 Writing paragraph

4.3 Notice writing

5 Listening and Speaking Exercises

1. Self and peer introduction
2. Newspaper reading

3. Just a minute session-Extempore
4. Greeting and starting a conversation
5. Leave taking
6. Thanking
7. Wishing well
8. Talking about likes and dislikes
9. Group Discussion
10. Listening Exercises.

● Student should be encouraged to participate in role play and other student centred activities in class room and actively participate in listening exercises
● Assignments and quiz/class tests, mid-semester and end-semester written tests − Actual practical work, exercises and viva-voce − Presentation and viva-voce

Recommended Texts:

1. Communicating Effectively in English, Book-I by RevathiSrinivas; Abhishek Publications, Chandigarh.
2. Communication Techniques and Skills by R. K. Chadha; DhanpatRai Publications, New Delhi.

Reference Books:

1. High School English Grammar and Composition by Wren & Martin; S. Chand & Company Ltd., Delhi.
2. Excellent General English-R.B.Varshnay, R.K. Bansal, Mittal Book Depot, Malhotra
3. The Functional aspects of Communication Skills – Dr. P. Prsad, S.K. Katria & Sons, New Delhi
4. Q. Skills for success – Level & Margaret Books, Oxford University Press.
5. e-books/e-tools/relevant software to be used as recommended by AICTE/ NITTTR, Chandigarh.

Web References:

1. http://www.mindtools.com
2. http://www.letstalk.com.in
3. http://www.englishlearning.com
4. http://learnenglish.britishcouncil.org/en/
5. http://swayam.gov.in

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	L	L	M	S	M	S	L	S	S	M
CO2	S	M	L	M	L	S	M	L	M	S
CO3	M	S	S	L	M	S	L	M	S	M
CO4	S	L	S	M	S	L	L	M	M	S
CO5	S	M	L	S	L	M	S	L	M	S

S-Strong M-Medium L-Low

Title of the Course/Paper	Communication Skills for Software Engineers - II		
Soft Skill -	Year - Semester		Credit: 2

Objectives:

- Knowledge of English Language plays an important role in career development.
- This subject aims at introducing basic concepts of communication besides laying emphasis on developing listening, speaking, reading and writing skills as parts of Communication Skill.

Outcomes:

- Frame correct sentences with illustrations
- Comprehend the language correctly and Interpret the language correctly
- Use given material in new situations.
- Correspond effectively using various types of writings like letters, memos etc.
- Communicate effectively in English with appropriate body language making use of correct and appropriate vocabulary and grammar in an organised set up and social context.

1. Functional Grammar

1.1 Prepositions
1.2 Framing Questions
1.3 Conjunctions
1.4 Tenses

2 Reading

2.1 Unseen Passage for Comprehension (Vocabulary enhancement - Prefixes, Suffixes, one word substitution, Synonym and Antonym) based upon the passage should be covered under this topic.

3 Writing Skill

3.1. Correspondence a) Business Letters- Floating Quotations, Placing Orders, Complaint Letters. b) Official Letters- Letters to Government and other Offices

3.2. Memos, Circular, Office Orders
3.3. Agenda & Minutes of Meeting
3.4. Report Writing

LIST OF PRACTICALS

Note: Teaching Learning Process should be focused on the use of the language in writing reports and making presentations. Topics such as Effective listening, effective note taking, group discussions and regular presentations by the students need to be taught in a project oriented manner where the learning happens as a byproduct.

4 Speaking and Listening Skills

1. Debate
2. Telephonic Conversation: general etiquette for making and receiving calls
3. Offering- Responding to offers.
4. Requesting – Responding to requests
5. Congratulating
6. Exploring sympathy and condolences

7. Asking Questions- Polite Responses
8. Apologizing, forgiving
9. Complaining
10. Warning
11. Asking and giving information
12. Getting and giving permission
13. Asking for and giving opinions

● Students should be encouraged to participate in role play and other student-centered activities in class rooms and actively participate in listening exercises

● Assignments and quiz/class tests, mid-semester and end-semester written tests - Actual practical work, exercises and viva-voce - Presentation and viva-voce

Recommended Texts:

1. Communicating Effectively in English, Book-I by RevathiSrinivas; Abhishek Publications, Chandigarh.
2. Communication Techniques and Skills by R. K. Chadha; Dhanpat Rai Publications, New Delhi.

Reference Books:

1. High School English Grammar and Composition by Wren & Martin; S. Chand & Company Ltd., Delhi.
2. e-books/e-tools/relevant software to be used as recommended by AICTE/NITTTR, Chandigarh.

Web References:

1. http://www.mindtools.com
2. http://www.letstalk.com.in
3. http://www.englishlearning.com
4. http://learnenglish.britishcouncil.org/en/
5. http://swayam.gov.in

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	L	L	M	S	M	S	L	S	S	M
CO2	S	M	L	M	L	S	M	L	M	S
CO3	M	S	S	L	M	S	L	M	S	M
CO4	S	L	S	M	S	L	L	M	M	S
CO5	S	M	L	S	L	M	S	L	M	S

S-Strong M-Medium L-Low

Title of the Course/Paper	Personality Development and other Soft skills for Software Engineers		
Soft Skill -	Year - Semester		Credit: 2

Objectives:

- The course intends to develop talent, facilitate employability enabling the incumbent to excel and sustain in a highly competitive world of business.
- The programme aims to bring about personality development with regard to the different behavioural dimensions that have far reaching significance in the direction of organisational effectiveness.
- To make students know about self-awareness, life skills, soft skills, need for personal development etc**.**

Outcomes:

- The student will be able to understand, analyse develop and exhibit accurate sense of self.
- Think critically.
- Demonstrate knowledge of personal beliefs and values and a commitment to continuing personal reflection and reassessment.
- Learn to balance confidence with humility and overcome problems associated with personality

Unit 1: Personality Development : A Must for Leadership and Career Growth

Case 1: One's Personality Sends Out a Signal that Others Read
Case 2: Same Person: Consciously Different Personalities can be Powerful
Case 3: There isn't One Right Personality
Learnings About Personality Development from the Three Cases
Personality Analysis - Freudian Analysis of Personality Development - Swami Vivekananda's Concept of Personality Development - Physical Self- Energy Self - Intellectual Self - Mental Self - Blissful Self - Personality Begets Leadership Qualities - Interpersonal Skills - Resolving Conflict - A Smiling Face - Appreciative Attitude - Assertive Nature - Communication - Skills-Listening Skills -Developing Empathy - The Personality Attribute of Taking Bold Decisions - Personality Types and Leadership Qualities - Mapping the Different Personality Types - Perfectionists-Helpers-Achievers-Romantics-

Observers -Questioners - Enthusiasts or Adventurers-Bosses or-Asserters-Mediators or Peacemakers - Personality Tests - Example of a Personality Test: Jung Typology Test - Personality Assessment

Unit 2: Soft Skills: Demanded by Every Employer

Case I: Dr Devi Shetty

Case II: Abraham Lincoln

Case III: Jeff Immelt

Lessons from the Three Case Studies - Change in Today's Workplace: Soft Skills as a Competitive Weapon - Antiquity of Soft Skills - Classification of Soft Skills - Time Management -Attitude -Responsibility - Ethics, Integrity, Values, and Trust -Self-confidence and Courage - Consistency and Predictability - Teamwork and Interpersonal Skills - Communication and Networking - Empathy and Listening Skills - Problem Solving, Troubleshooting and Speed-reading - Leadership

Unit 3: Your Resume or Curriculum Vitae: The First Step Forward

The Strategy of Resume Writing—From an Employer's Perspective

Strategy I: The Resume Should Reveal ose Personality Traits that Align with the Organization's Values

Strategy II: The Resume Should Convince the Potential Employer of Right Fitment to the Opening

Strategy III: The Resume Should Show to the Employer the Bene ts that the Candidate Will Bring in

A Favourable First Impression—The 'Career Objective' in the Resume - The Main Body of the Resume - Clarity and Crispness of the Resume - Format and Content of the Resume - A Fresher's Resume - Examples - Example of a Well-written Resumeby an Experienced Professional -Example of a Well-written Resumeof a Fresh Graduate - Example of a Poorly Written Resume - Writing a Modern Resume - How is the Modern CV Different from the Traditional One? - Various Modern ResumeFormats -

Unit 4: Group Discussion: A Test of Your Soft Skills

Case Studies - Learnings from the Three Case Studies - Ability to Work as a Team - Communication Skills, Including Active Listening - Non-verbal Communicatio - Leadership and Assertiveness - Reasoning - Ability to Influence - Innovation, Creativity and Lateral Thinking - Flexibility - Group Discussion Types - The Responsibility of the First Speaker - Concluding the Discussion — The Technique of Summing Up

Recommended Texts:

1. Personality Development and SOFT SKILLS, BARUN K. MITRA, Oxford University Press

Reference Books:

1. Communicating Effectively in English, Book-I by Revathi Srinivas; Abhishek Publications, Chandigarh.

2. Communication Techniques and Skills by R. K. Chadha; Dhanpat Rai Publications, New Delhi.

Web References:

1. http://www.mindtools.com
2. http://www.letstalk.com.in
3. http://www.englishlearning.com
4. http://learnenglish.britishcouncil.org/en/
5. http://swayam.gov.in

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	L	L	M	S	M	S	L	S	S	M
CO2	S	M	L	M	L	S	M	L	M	S
CO3	M	S	S	L	M	S	L	M	S	M
CO4	S	L	S	M	S	L	L	M	M	S
CO5	S	M	L	S	L	M	S	L	M	S

S-Strong M-Medium L-Low

Title of the Course/Paper	Documentation and Interview skills for Software Engineers		
Soft Skill -	Year - Semester		Credit: 2

Objectives:

- Ensure that you understand what the job involves, and that you have the necessary skills
- Make sure you do want to work for the company
- Check that the philosophy/values of the company match your personal requirements
- Find out more about the job, training, career structure etc.

Outcomes:

- Understand the purpose of interviews
- Be aware of the processes involved in different types of interviews
- Know how to prepare for interview
- Be clear about the importance of self presentation

Unit 1: Job Interviews: The Gateway to the Job Market

Types of Interviews - Groundwork Before the Interview - Abide by the Dress Code - Importance of Body Language in Interviews - Need for Proper Articulation - **Probable Interview Questions**: Tell Us about Yourself - Would You Call Yourself a Team Player? - **Few Tricky Questions and Possible Answers:**Why Should We Employ You? - Do You Have Off ers from Other Companies? - What Salary are You Expecting? - How Much do You ink You are Worth? - What Kind of a Culture are You Comfortable with? - What is More Important to You—Salary or Growth Opportunities? - What do You Know about Our Company? - Tell Us about Your Strengths and Weaknesses - Where do You See Yourself in 5 or 10 Years? - What are Your Plans for Higher Studies? - When Leading a Team, How

Will You Motivate Your Team Members and Resolve Any Differences between them? - What Has Been the Biggest Challenge You Have Faced, and How Did You Handle It? - What Do You ink are the Essential Qualities of a Good Employee? - You Claim to be Computer-savvy. Can You Mention Any Innovative Way to Enhance the Sales of the Company Using Your Computer Knowledge and Skills? — Concluding an Interview - Telephonic or Video Interview—A Growing Trend - Disadvantages of Telephonic or Video Interview - **A Mock Interview:**Why did the Interview Team Select Vikram? - Why did the Interview Team not Select Chandra and Amit?

Unit 2:Body Language: Reveals Your Inner Self and Personality

Emotions Displayed by Body Language:Aggressive - Submissive - Attentive - Nervous - Upset - Bored - Relaxed - Power - Defensive—Handshake—The Most Common Body Language— Eyes— A Powerful Reflection of One's Inner Self —Entry to My Space— Personal Zones May Vary: Intimate Zone - Personal Zone - Social Zone - Public Zone - Typical Body Language when Zones are Intruded — Body Language Exhibited During Different Professional Interactions -Interview - Manager's Discussions with a Subordinate Employee - Discussions with Supervisor - Presentation to a Large Audience - Group Discussions - Video-conference

Unit 3: Enhance Your Writing Skill to Create an Impression

Fifteen Principles to Increase Clarity of Communication - Use Short, Simple and Clear Words - Use Short Sentences - Do not Cram Di erent Points into One Sentence - Using Compact Substitutes for Wordy Phrases - Remove Redundant Words and Expressions - Avoid Use of Mixed Metaphors - Avoid Hackneyed and Stilted Phrases - Avoid Verbosity in the Use of Common Prepositions - Do not Twist the Word Order - Present Similar Ideas in a Sentence with Same Structural and Grammatical Form - Make Positive Statements Without Being Hesitant or Non-committal - e Statements Without Being Hesitant or Non-committal - Avoid Pompous Words and Phrases - Use Active Instead of Passive Voice - Ensure Correct Spelling and Grammar in the Text - Substitute Easily-understood Words for Words Imported from Other Fields - Edit-Edit-Edit - The Reader's Perspective - Clarity of Thought - Clarity of Text - Example of Poorly and Well-written Texts

Unit 4: Fog Index: Provides Guidance for Proper Writing

Fog Index or Clarity Index -Examples of Passages with High and LowFog Index - Infogineering Clarity Rating - Flesch Kincaid Reading Ease Index - Other Readability Indices - Checking Grammar, Spelling and Voice - Clarity of Verbal Communication - Case 1 - Case 2

Recommended Texts:

1. Personality Development and SOFT SKILLS, BARUN K. MITRA, Oxford University Press

Reference Books:

1. **C**ommunicating Effectively in English, Book-I by Revathi Srinivas; Abhishek Publications, Chandigarh.
2. Communication Techniques and Skills by R. K. Chadha; Dhanpat Rai Publications, New Delhi.

Web References:

1. http://www.mindtools.com
2. http://www.letstalk.com.in
3. http://www.englishlearning.com
4. http://learnenglish.britishcouncil.org/en/
5. http://swayam.gov.in

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	L	L	M	S	M	S	L	S	S	M
CO2	S	M	L	M	L	L	M	S	M	S
CO3	M	S	S	L	M	S	L	M	S	M
CO4	S	L	S	M	S	L	L	M	M	S
CO5	S	M	L	S	L	M	S	L	M	S

S-Strong M-Medium L-Low

Title of the Course/Paper	**Team Project**		
Soft Skill -	Year - Semester		Credit: 2

Objectives:

- Understand programming language concepts, particularly object-oriented concepts or go through research activities.
- Plan, analyze, design and implement a software project or gather knowledge over the field of research and design or plan about the proposed work.
- Learn to work as a team and to focus on getting a working project done on time with each student being held accountable for their part of the project.
- Learn about and go through the software development cycle with emphasis on different processes - requirements, design, and implementation phases.

Outcomes:

- Demonstrate the ability to locate and use technical information from multiple sources.
- Demonstrate the ability to communicate effectively in speech and writing.
- To demonstrate a depth of knowledge of modern technology.
- To do the Project Scheduling, tracking, Risk analysis, Quality management and Project Cost estimation using different techniques.
- To complete an independent research project, resulting in at least a thesis publication, and research outputs in terms of publications in high impact factor journals, conference proceedings.

Project:

- Any Computer related project has to be developed using latest software as a team.
- The project must be presented for viva-voce at the end of the semester.
- Students will write up a project report, which is an essay to provide a complete record of all the work carried out in their projects.
- The student project reports will be assessed solely according to academic marking guidelines by the supervisor(s) of the student project.
- If the work of the candidate is found to be insufficient and plagiarism, the supervisor(s) will decide the further process.

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	L	L	M	S	M	S	L	S	S	M
CO2	S	M	L	M	L	S	M	L	M	S
CO3	L	S	S	L	M	L	M	S	S	M
CO4	S	L	S	M	S	L	L	M	M	S
CO5	S	M	L	S	L	M	S	L	M	S

S-Strong M-Medium L-Low

S.A.C. SEPT'2022