University Of Kerala

Learning Outcomes-based Curriculum Framework
(LOCF) **Post-Graduate Programme**

M.Sc. Computer Science

Regulations, Curriculum Framework and Syllabus

(Affiliated Colleges Syllabus effective from 2024 Admission Onwards)

UNIVERSITY OF KERALA

2024

SCHEME AND SYLLABUS OF M.Sc. COMPUTER SCIENCE

(2024 ADMISSION ONWARDS)

A. Objectives

1. To develop an interest in the candidates towards a career in academics and research, and to enable them with sufficient knowledge to become a competent academician.
2. To equip the students with adequate exposure and skills to empower them to catch a deserving position in the software industry.
3. To develop an interest in promoting the use of Computer Science for the positive development of our society and the environment.
4. To enable the students to contest for regional/national/international level competitive examinations.

B. Duration:

The duration of the course is four semesters in 2 years.

C. Eligibility

Candidates for admission to M. Sc. Programme in Computer Science should have passed:

(i) A Degree course with minimum three years duration after 10+2 with not less than 50% marks or 2 CGPA[S] out of 4 in Computer Science/Computer Application/Electronics /Mathematics / Statistics as main or an equivalent Degree recognized by the University of Kerala for the purpose.

(ii) Any Science degree with minimum three years duration after 10+2 with not less than 50% marks or 2 CGPA[S] out of 4 with Computer Science/Computer Application as one of the main/subsidiary/core subject or an equivalent Degree recognized by the University of Kerala for the purpose.

The candidate shall meet all other requirements in the prospectus published by the University from time to time.

D. Assessment

Assessment will be done in two components: Continuous Assessment (CA) and End-Semester Assessment (ESA).

(i) Continuous Assessment (CA)

Theory Courses: In addition to classroom lectures, students shall be assigned to application problems, classroom presentations, group activities, etc. Case studies/industry visits may also be organized. At least two tests shall be conducted for each course. Short viva may be carried out to assess assignments.

CA Mark for Theory Courses (Total: 25)

Attendance	5
Test	10
Assignment & Activities	10

Lab Courses: Each Lab course shall be completed under the supervision of a faculty member. **The students shall undertake a case study for each practical course.** The case study can be done as a team of 2 members if necessary**.** The practical record includes both lab exercises and a case study report.

CA Mark for Lab Courses (Total: 25)

Attendance	5	Lab performance	5
Internal Lab Test	5	Case Study	5
Record	5		

Seminar:

Each student shall present a seminar on any topic of interest related to the M. Sc Computer Science Programme. They should get the paper approved by the Faculty member in charge of the seminar and shall present it in the class.

CA Mark for Seminar (Total: 100)

Topic & Content of Seminar	20
Report	30
Presentation & Defense	50

Case Study:

Each student is required to present a detailed case study, which involves investigating a specific topic of interest related to the core/elective courses offered in the M.Sc. Computer Science Programme. Prior to presentation, students must obtain approval for their paper from the faculty member overseeing the case study and deliver it during class.

CA Mark for Case Study (Total: 100)

Continuous interaction and Discussion with guide:	20
Report	30
Presentation and Viva	50

Project work:

The project work shall be completed individually by each student under the guidance of a faculty member from the department. **The project work will consist of two phases: the minor project in the third semester and the major project in the fourth semester. The major project in the fourth semester will build upon the work initiated in the minor project during the third semester.**

An internal assessment team consisting of at least three members, chaired by the Head of Department or a senior faculty member shall be constituted at the college every year by the end of the second semester. The project guide of the candidate can be one of the members of the team. The Project proposals and synopsis submission shall be done at the beginning of the third semester itself. It is advisable to select the project topic and area keeping the following objectives in mind:

(i) The project work shall give enough opportunity for the students to apply some of the skills and knowledge gained through the theory courses.

(ii) The student shall get an exposure in developing industry type applications/utility software for computer systems or mobile devices/in studying and analyzing theoretical concepts and presenting a comparative analysis of state-of-the-art techniques/in developing new or improved algorithms/in the use of soft computing techniques in selected areas/discipline

If the student chooses to do the project in an organization other than the college, the department shall ensure the following:

1. A qualified person supervises the project. The External Supervisor shall be a post graduate in either Science/Applied Science/Engineering branches. He/She shall have at least three years of experience in running/managing/implementing/supervising such projects. A declaration shall be obtained in this regard from that person, and shall be kept by the Department.
2. An attendance statement and performance feedback shall be obtained from the External supervisor. The student has to present periodic reports and attend for evaluation process before the internal assessment team at the College as per the schedule.

Minor Project (Total: 100)

For Minor Project, internal evaluation will be conducted by College , utilizing an assessment team comprising a minimum of three members, led by either the Head of Department or a senior faculty member, with the project guide of the candidate potentially serving as one of the team members. Students are required to submit their reports by the specified deadline.

CA Mark for Minor Project (Total: 100)

Literature Survey /System study	25 marks
Interim Report	20 marks
Methodology / Design	20 marks

Presentation	20 marks
Viva - Voce	15 marks

CA Mark for Major Project (Total: 50)

Study Phase activities & Report:	10 marks
Methodology/Design	10 Marks
Implementation	10 marks
Results & Findings	10 marks
Presentation	10 marks

ii. End Semester Assessment (ESA)

The University will conduct the end-semester Assessment for all courses. A student with 75% attendance in a course is eligible to attend the University examination.

Theory Courses: (Total: 75)

The question paper consists of two parts:

Part A (27 marks). Nine compulsory questions, of 3 marks each.
Part B (48 marks). Students must answer one out of two questions from each module. Each question carries eight marks.

Lab Courses: (Total: 75)

Lab examinations shall be conducted in each college by two examiners appointed by the University, of which one shall be from other colleges. The External Examiner will finalize the marks in consultation with the internal examiner. The questions for the examination shall be prepared before each examination and approved by the board of examiners. A candidate shall be asked to answer one out of two questions given to him.

The marks will be distributed as follows**. (Total: 75)**

Description of procedure: 10 marks [The procedure/algorithm/flow chart/pseudo code for solving the problem(s) shall be explained in the answer sheet.]
Preparation of program: 15 marks
Logic &Output : 20 marks. *20 marks shall be distributed as follows:*
(i) 15 marks for the correct output of the given problem.
(ii) 5 marks for completing the modifications suggested by the examiner(s) in the given questions during the examination hours.
[The program/code shall work for all cases of the given problem. Different test cases and answers shall be written in the answer sheet.
Viva: 15 marks. Viva shall be on the problem domain, based on the programming tool used, from the

area of study for solving the problem/from the theory concepts related to the area.

Case study Report and Viva : 15 marks

Major Project (Total: 50)

Major Projects shall be evaluated at the examination centers by a panel of two examiners appointed by the university, one of which shall be from other colleges. The project report shall be finalized after the internal Assessment. The candidates shall present the findings/output of their work before the examiners during the examination hours. The examiners will conduct a viva voce also. At the time of evaluation, students are required to present the contents of the Minor project along with the major project, ensuring a comprehensive overview of their project's progression and outcomes.

ESA for Major Project			
Report of the work	10 marks	Literature Survey/System study	10 marks
Methodology /Design	5 marks	Presentation	10 marks
Findings/Implementation	10 marks	Viva Voce	5 marks

Comprehensive Viva (Total: 100)

It is mandatory that the Comprehensive Viva shall be conducted by separate examiners other than Project Evaluation. The viva will be carried out by a panel of two examiners appointed by the University, of which one shall be from outside the college. Though the viva shall be based on the entire syllabus contents, the candidates may be given an opportunity to opt a set of subjects, not less than 40% of the programme. However, the candidate, in any case, shall not be asked to write answers to the questions given by the examiners.

E. Question Paper Pattern:

The maximum mark for the theory examinations will be 75 and the time duration will be 3 hours. The question paper shall contain two parts; Part-A and Part-B.

Part-A shall be for 27 marks and shall contain 9 compulsory short answer questions. Each question carries 3 marks. At Least one question from each module.

Part B shall be for 48 marks and shall contain 12 questions, 2 questions from each module out of which the student has to answer 1 question from each module. Each question carries 8 marks.

F. Pass Requirements:

For each subject(including practical), a student should get a minimum of 40% marks in the university examinations and 50% aggregate for the CA and ESA together for all theory and practical courses except

Major Project. For Major Project and Comprehensive viva-voce in the 4th semester each student should get a minimum of 50% for the university examination and 50% aggregate for the CA and ESA together. Classification of passed candidates will be as per the University norms.

PROGRAMME OUTCOMES (PO) for M. Sc Computer Science Programme

	PROGRAMME OUTCOMES (PO)
PO1	To develop an interest in the candidates towards a career in academics and research, and to enable them with sufficient knowledge to become a competent academician
PO2	To apply knowledge of mathematical, scientific, and computer science to evaluate, analyze, synthesize, model and integrate technologies to develop new computer systems for applied engineering systems.
PO3	To equip the students with adequate exposure and skills to empower them to catch a deserving position in the software industry.
PO4	Ability to identify, critically analyze and formulate complex computing problems using fundamentals of computer science and application domains.
PO5	To develop an interest in promoting the use of Computer Science for the positive development of our society and the environment
PO6	Recognize the need for and develop the ability to engage in continuous learning as a Computing professional.
PO7	Create, identify and apply appropriate techniques, resources, and modern computing tools to complex computing activities.
PO8	To enable the students to contest for regional/national/international level competitive examinations.

Programme Specific Outcomes

The students on completion of M.Sc (Computer Science) Programme will be able to:

PSO1	**Communicate computer science concepts, designs, and solutions effectively and professionally**
PSO2	Apply knowledge of computing to produce effective designs and solutions for specific problems
PSO3	Use of software development tools, software systems, and modern computing platforms to solve real life problems
PSO4	Investigate research gaps, analyze and carry out research in the specialized/emerging areas
PSO5	Apply knowledge of recent computing technologies, skills and current tools of computer science
PSO 6	Utilize skills and knowledge for computing practice with commitment on social, ethical, cyber and legal values

Mapping of PSO to PO

PO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
PO1				✔		
PO2	✔	✔				
PO3			✔			
PO4		✔		✔		
PO5					✔	
PO6						✔
PO7			✔	✔		✔
PO8						✔

CURRICULUM FRAMEWORK

Semester I		Credits	L	T	P	CA	ESA	Total
CS 511	Mathematical Foundations of Computer Science	4	3	1		25	75	100
CS 512	Operating Systems and Virtualization Technologies	4	3	1		25	75	100
CS 513	Data Structures & Algorithms	4	3	1		25	75	100
CS 514	Full Stack Web Development	4	3	1		25	75	100
CS 515	Full Stack Web Development Lab	2			4	25	75	100
CS 516	Data Structures & Algorithms Lab	2	-		4	25	75	100
CS 517	Seminar	1	-	1	-	100		100
	Total	**21**	**12**	**5**	**8**			**700**

Semester II		Credits	L	T	P	CA	ESA	Total
CS 521	Computer Networks	4	3	1	-	25	75	100
CS 522	Software Engineering	4	3	1	-	25	75	100
CS 523	Introduction to Machine Learning	4	3	1	-	25	75	100
CS 524	Elective I	4	3	1	-	25	75	100
CS 525	Computer Networks Lab	2	-		4	25	75	100
CS 526	Machine Learning Lab	2	-		4	25	75	100
CS 527	Case Study	1	-	1	-	100		100
	Total	**21**	**12**	**5**	**8**			**700**

Semester III		Credits	L	T	P	CA	ESA	Total
CS 531	Automata Theory & Compiler Design	4	3	1	-	25	75	100
CS 532	Digital Image Processing	4	3	1	-	25	75	100
CS 533	Information retrieval	4	3	1	-	25	75	100
CS 534	Elective II	4	3	1	-	25	75	100
CS 535	Minor Project	3	-	-	5	100		100
CS 536	Digital Image Processing Lab	2		-	4	25	75	100
	Total	**21**	**12**	**4**	**9**			**600**

Semester IV		Credits	L	T	P	CA	ESA	Total
CS 541	Research Methodology & Report Writing	4	3	1		25	75	100
CS 542	Elective III	4	3	1		25	75	100
CS 543	Major Project	10			17	50	50	100
CS 544	Comprehensive Viva	3					100	100
	Total	**21**	**6**	**2**	**17**			**400**

Elective Courses		Credits
Elective I		
CS 524 A	Cyber Forensics & Cyber Laws	4
CS 524 B	Human Computer Interaction	4
CS 524 C	Natural Language Processing	4
Elective II		
CS 534 A	Big Data Analytics	4
CS 534 B	Cloud Computing Technologies	4
CS 534 C	Mining massive datasets	4
Elective III		
CS 542 A	Deep learning	4
CS 542 B	Optimization Techniques	4
CS 542 C	Cryptography And Network Security	4

SEMESTER I	COURSE CODE : CS 511	CREDIT :4

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Course Outcomes		CL	KC
CO1	Apply sequence, product sets, and partitions in solving problems	Ap	P
CO2	Analyze linear mappings and their properties	An	C
CO3	Apply matrix decomposition techniques in solving problems	Ap	P
CO4	Analyze properties and theorems related to graphs and trees	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I Fundamentals: Sets and subsets, operation on sets, Inclusion Exclusion principle, pigeonhole principle, sequence, product sets and partitions Relations, Matrix representation of relations, classification of relations ,N-ary relations, equivalence relation, Functions , Permutation Functions ,Growth of functions, Partially ordered sets, Lattices.

Module II Linear Algebra: Systems of Linear Equations, Matrices, Solving Systems of Linear Equations, Vector Spaces, Linear Independence, Basis and Rank, Linear Mappings

Module III Matrix Decompositions: Determinant and Trace, Eigenvalues and Eigenvectors, Cholesky Decomposition, Eigen decomposition and Diagonalization, Singular Value Decomposition, Matrix Approximation

Module IV Graph theory, Basic concepts of graph theory, Graph terminology and Special types of graph, representation of graph, graph isomorphism, planar and non-planar graphs, Euler paths and circuits, Hamiltonian paths and circuits, Trees spanning tree, theorems on trees.

Module V Vector Calculus: Differentiation of Univariate Functions, Partial Differentiation and Gradients, Gradients of Vector-Valued functions, Gradients of Matrices, Useful identities, Backpropagation and Automatic Differentiation

Module VI Probability, Axioms of probability, conditional probability, General aspects, Random variables, Scatter diagram, Sample, uniform distribution, Basic rules of probability

References:

1. Marc Peter Deisenroth, A Aldo Faisal and Cheng Soon Ong (2021). Mathematics for Machine Learning, Cambridge University Press. https://mml-book.github.io/
2. Lieven Vandenberghe, Stephen P. Boyd (2018). Introduction to Applied Linear Algebra Vectors, Matrices, and Least Squares, Cambridge University Press.https://web.stanford.edu/~boyd/vmls/
3. Kuldeep Singh (2020). Linear Algebra: Step by Step. Oxford University Press.
4. Gilbert Strang (2020). Linear Algebra and Learning from Data, Wellesley Publishers.
5. Bernard Kolan c, Busby & Sharon Ross, Discrete Mathematical Structures[PHI]
6. J.P.Tremblay&R.Manohar, Discrete Mathematical Structures with Application to computer science[TataMcGraw-Hill]
7. C.J.Liu, Elements of Discrete Mathematics , MGH
8. Johnsonbaugh, Discrete Mathematics, Pearson Ecducation,2007
9. Grassmann,Logic and DisctereMathermatics: A Computer Science Perspective,Person Education,2007
10. Higher Engineering Mathematics, Dr.B.S.Grewal
11. Advanced Engineering Mathematics 9th Edition , Erwin Kreyszig

SEMESTER I	COURSE CODE : CS 512	CREDIT :4

OPERATING SYSTEMS AND VIRTUALIZATION TECHNOLOGIES

Course Outcomes		CL	KC
CO1	Recall the various types of operating systems, including mainframe, server, personal computer, and embedded systems.	R	F
CO2	Understand the concepts of process management, file management, and directory management in operating systems.	U	C
CO3	Apply knowledge of CPU scheduling algorithms, such as First Come First Served and Round Robin, to solve scheduling problems.	Ap	P
CO4	Analyze deadlock scenarios and assess the effectiveness of deadlock prevention and detection algorithms in operating systems.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I Introduction: Types of OS - Mainframe, server, multiprocessor, Personal computer, handheld, embedded, sensor-node, real-time, smart card. Operating System Concepts, System Calls - process management, file management, directory management, Miscellaneous System Calls. Operating System Structure. System boot process. Open-Source Operating Systems.

Module II Processes: Process concept, Process scheduling, Operations on processes, Inter-process communication. Overview of threads. Process Synchronization: Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic problems of Synchronization. Simple programs using fork(), semaphores and other IPC mechanisms should be discussed in class

Module III CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms - First come First Served, Shortest Job First, Priority scheduling, Round robin scheduling, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling. Overview of Linux scheduling. Deadlocks: System Model, Necessary conditions, Resource allocation graphs, Deadlock prevention, Deadlock avoidance - Banker's algorithms, Deadlock detection, Recovery from deadlock.

Module IV Memory Management: Concept of address spaces, Swapping, Contiguous memory allocation, Segmentation, Paging. Virtual memory, Demand paging, Page replacement algorithms. File System: File concept, Access methods, Tree-structured directories, File system mounting, Protection. File System Implementation: File System structure, implementation.

Module V: Physical and virtual machines, Traditional and virtual computing, Understanding

irtualization, Applications and limitations of virtualization, Simulations and Emulations, Challenges in Virtualized environment, tools and technologies in virtualized environments. Hypervisors: types, architecture

Module VI Container: fundamentals, Comparison with virtual machines, Container technologies, Configuring a container engine, Container virtual networking, Container orchestration and clustering, Images and containers.

References:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne (2013). Operating System Concepts, 9th edition, John Wiley & Sons.
2. Andrew S. Tanenbaum, Herbert Bos (2016). Modern Operating Systems, 4th edition, Pearson Education India
3. William Stallings (2018), Operating systems - Internals and Design Principles, 9th Edition, Pearson Education, PHI.
4. Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau (2018), Operating Systems: Three Easy Pieces, Arpaci-Dusseau Books. Available Online: https://pages.cs.wisc.edu/~remzi/OSTEP/
5. Garry Nutt, NabenduChaki, SarmisthaNeogy, Operating Systems, 3rd Edition, Pearson Education.
6. D. M. Dhamdhere (2011), Operating Systems, 2nd Edition, Tata McGraw Hill.
7. Chris Wolf , Erick M. Halter, Virtualization: From the Desktop to the Enterprise, APress 2005.
8. Matthew Portnoy, Virtualization Essentials, Wiley; Second edition (2016)
9. Sean P. Kane, Karl Matthias, Docker: Up & Running - Shipping Reliable Containers in Production, Second Edition, O'Reilly

SEMESTER I	COURSE CODE :CS 513	CREDIT :4

Full Stack Web Development

Course Outcomes		CL	KC
CO1	Recall the structure of HTML elements and attributes.	R	F
CO2	Understand the purpose and significance of HTML5 semantic elements in webpage organization.	U	C
CO3	Apply CSS to style HTML elements using different methods such as inline, internal, and external stylesheets.	Ap	P
CO4	Design and develop HTML forms with appropriate input types and validation using HTML5 attributes.	Cr	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I : Introduction to WWW and HTML, Structure of HTML, HTML elements and attributes, Headings,Paragraphs, Formatting tags, line breaks, Comments, Links, Images, Lists, HTML5 Semantic Elements (header, footer, nav, section, article, nav, aside), HTML Tables.

Module II: HTML Forms (input, select, textarea, button, datalist), Input types (text, password, submit, radio, checkbox, date, email), Input attributes (value, readonly, disabled, maxlength, autocomplete, list, min, max, placeholder), HTML5 form validation (required and pattern attribute of input type), Applying style to html using CSS (Inline, Internal and External CSS, Colors, Fonts, Borders, Padding, Applying style using class and id attribute), Positioning Elements: Absolute Positioning, Relative Positioning, Basics of Responsive CSS, Media port & Media Queries

Module III : Introduction, JavaScript Fundamentals - variables, operators, data types, strings, arrays, functions, objects, control statements, events, querySelector, DOM Manipulation - JavaScript Console, Arrow Functions, Intervals, Local Storage.

Module IV : overview of JSON - JSON Data Interchange Format: Syntax,Data Types, Object, JSON Schema, using APIs in Javascript.

Module V : Python basics - variables, control statements, lists, tuples, dictionaries, sets, functions, modules, basics of object oriented programming in python, decorators, lambda functions, exceptions.

Module IV: Introduction to web applications, django framework, routes, templates, conditionals, django forms, sessions, Basics of Django Models, Migrations, Django Admin, User Management.

References:

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, Internet & World Wide Web How to Program 5/e, Pearson, 2018.
2. Julie C. Meloni, PHP, MySQL & JavaScript All in One, Sams Teach Yourself, 6/e, Sams,2017.
3. https://docs.djangoproject.com/en/4.2/
4. https://cs50.harvard.edu/web/2020/
5. David Flanagan, Javascript, The Definitive Guide, 7/e, O'Reilly, 2020. [6] Lindsay Bassett, Introduction to JavaScript Object Notation: A To-the-Point Guide to JSON.

SEMESTER I	COURSE CODE :CS 514	CREDIT :4

DATA STRUCTURES & ALGORITHMS

Course Outcomes		CL	KC
CO1	Recall the fundamental data structures.	R	F
CO2	Understand the concept of algorithmic problem solving.	U	C
CO3	Apply algorithms.	Ap	P
CO4	Analyze the efficiency classes of algorithms.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I Introduction to Algorithms, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental Data Structures, Fundamentals of the Analysis of Algorithm Efficiency - The Analysis Framework, Asymptotic Notations and Basic Efficiency Classes.

Module II: Brute Force and Exhaustive Search - Selection Sort and Bubble Sort, Sequential Search and Brute-Force String Matching, Exhaustive Search.

Module III: Trees- Binary Trees – level and height of the tree, complete-binary tree, Graphs – representation of graphs, BFS and DFS, Divide-and-Conquer - Mergesort, Quicksort, Binary search, Binary Tree Traversals.Greedy Technique - Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm.

Module IV: Dynamic Programming - The Knapsack Problem and Memory Functions, Warshall's and Floyd's Algorithms.

Module V: File Structures: Concepts and organization, Sequential file, Indexed sequential files, Direct files.

Module VI: P, NP, and NP-Complete Problems. Backtracking - n-Queens Problem, subset-sum problem.

Text books

1. Anany Levitin (2012). Introduction to the Design and Analysis of Algorithms, 3rd edition,Pearson.

References:

1. Thomas H. Cormen. Charles E. Leiserson. Ronald L. Rivest. Clifford Stein. Introduction to Algorithms, Third Edition. The MIT Press.
2. Jon Kleinberg, Eva Tardos (2013), Algorithm Design, First Edition, Pearson Education India
3. Jeff Erickson (2019), Algorithms, First Edition, (Online) https://jeffe.cs.illinois.edu/teaching/algorithms/book/Algorithms-JeffE.pdf
4. Steven S. Skiena (2020), The Algorithm Design Manual, 3/E, Springer
5. Alfred V. Aho, John E. Hopecroft, Jeffrey D. Ullman (2002), Design and Analysis of Computer Algorithms, Pearson
6. Michael T. Goodrich, Roberto Tamassia (2006), Algorithm Design: Foundations, Analysis and Internet Examples, Wiley
7. Robert Sedgewick, Philippe Flajolet, An Introduction to Analysis of Algorithms, Addison-Wesley Professional
8. Tim Roughgarden (2017), Algorithms Illuminated: Part 1: The Basics

SEMESTER I	COURSE CODE : CS 515	CREDIT :2

FULL STACK WEB DEVELOPMENT LAB

Course Outcomes		CL	KC
CO1	Recall HTML elements, CSS properties, JavaScript data types, and Django framework components	R	F
CO2	Explain HTML5 semantic elements, CSS-HTML relationships, JavaScript concepts, Python OOP, and Django MVC architecture.	U	C
CO3	Create HTML forms, apply CSS styles, manipulate the DOM with JavaScript, handle form submissions in Python, and implement Django views/templates.	Ap	P
CO4	Evaluate CSS layout techniques, debug JavaScript code, compare form validation approaches, assess Python function performance, and reflect on Django design choices.	An	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create, KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

1. Create a web pages using basic html formatting tags
2. Create a web page containing images and hyperlinks
3. Create a web page containing table
4. Create a web page containing all types of lists
5. Create a simple webpage introducing yourself. Include sections such as a brief biography, hobbies, and contact information. Utilize HTML elements like headings, paragraphs, links, and images to structure and format your web page effectively.
6. Create a web page using various CSS selectors
7. Design a registration form for a website. Include fields for username, email, password, and date of birth. Apply CSS styling to enhance the visual appearance of the form elements. Implement form validation using HTML5 attributes and JavaScript to ensure that all required fields are filled out correctly before submission.
8. Develop a simple to-do list application using HTML, CSS, and JavaScript. Allow users to add new tasks, mark tasks as completed, and delete tasks. Implement event listeners and DOM manipulation techniques to dynamically update the task list based on user actions
9. Create a weather forecast application using an external weather API. Allow users to input their location and retrieve current weather information such as temperature, humidity, and conditions. Utilize JSON data format to handle API responses and display the weather data dynamically on the webpage.
10. Implement a simple calculator application using Python. Define classes for basic mathematical

operations such as addition, subtraction, multiplication, and division. Allow users to input two numbers and choose an operation to perform. Display the result of the calculation to the user.

11. Build a basic blog application using Django. Create models for blog posts with attributes like title, content, and publication date. Implement views and templates to display a list of blog posts, view individual posts, and add new posts. Include functionalities for user authentication and authorization to allow registered users to create, edit, and delete their own posts.

SEMESTER I	COURSE CODE :CS 516	CREDIT :2

DATA STRUCTURES & ALGORITHMS LAB

Course Outcomes		CL	KC
CO1	understand and apply various sorting algorithms in Python.	R	C
CO2	Students will analyze and create Python programs for searching algorithms and string matching techniques.	An, Cr	P
CO3	Students will demonstrate conceptual understanding and practical application of graph traversal algorithms using Python.	Ap	C
CO4	Design and implement Python programs for file handling operations, integrating conceptual understanding with procedural skills.	Cr	C, P

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create, KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

1. Write Python functions to implement a selection sort algorithm for sorting a list of integers.
2. Write Python functions to implement a bubble sort sort algorithm for sorting a list of integers.
3. Write Python functions to implement a merge sort sort algorithm for sorting a list of integers.
4. Write Python functions to implement a quick sort sort algorithm for sorting a list of integers.
5. Write Python functions to implement a binary search algorithm.
6. Implement the sequential search algorithm for finding a target value in a list of integers.Extend the implementation to perform brute-force string matching for finding a pattern within a text.
7. Design a Python program to represent a graph using an adjacency list or matrix. Implement depth-first search (DFS) and breadth-first search (BFS) algorithms to traverse a graph
8. Implement a Python class to represent a binary tree. Include methods for inserting nodes, traversing the tree in preorder, inorder, and postorder
9. Implement a Python program to solve the subset-sum problem using exhaustive search.
10. Design a Python program to demonstrate sequential file handling. Implement functions to read data from a sequential file, add new records, and search for specific records based on user input.

SEMESTER I	COURSE CODE :CS 517	CREDIT :1

SEMINAR

COURSE OBJECTIVES:

Each student shall present a seminar on any topic of interest related to the core / elective courses offered in the M. Sc. Computer Science programme / recent trends in the field of Computer Science. He / she shall select the topic based on the references from National or International journals of repute. They should get the paper approved by the Programme Co-Ordinator / Faculty member in charge of the seminar and shall present it in the class. Every student shall participate in the seminar.

COURSE OUTCOMES:

The students should undertake a detailed study on the topic and submit a report at the end of the semester. Marks will be awarded based on the topic, presentation, participation in the seminar and the report submitted.

SEMESTER II	COURSE CODE : CS 521	CREDIT :4

COMPUTER NETWORKS

Course Outcomes		CL	KC
CO1	Recall network application architectures, transport services, application-layer protocols, and DNS services.	R	F
CO2	Understand the concepts of HTTP connections, message format, DNS working, and HTTP/2 impact.	U	C
CO3	Apply knowledge to differentiate HTTP connections, interpret HTTP messages, troubleshoot DNS, and optimize HTTP performance.	Ap	P
CO4	Analyze the relationship between application-layer protocols and network services, and assess the impact of HTTP/2 on web performance.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I Application Layer: Network Application Architectures, Process, Transport Services, Application-Layer Protocols, Web and HTTP - Non-Persistent and Persistent Connections, HTTP Message Format, cookies, HTTP/2. DNS - Services, working, DNS records and messages..

Module II Video Streaming and Content Distribution Networks - Internet Video, HTTP Streaming, Content Distribution Networks, Case Studies: Netflix and YouTube. **Transport Layer -** Transport-Layer Services, Relationship Between Transport and Network Layers, Multiplexing and Demultiplexing, Connectionless Transport: UDP - Segment structure.

Module III Connection-Oriented Transport - TCP, connection, segment structure, reliable data transfer, flow control and connection management. Principles of Congestion Control - Approaches. **Network Layer -** Forwarding and Routing: The Data and Control Planes, SDN Approach. Router - Input Port Processing and Destination-Based Forwarding, switching, Packet Scheduling.

Module IV The Internet Protocol (IP) - IPv4 Addressing, Obtaining a Block of Addresses, Network Address Translation (NAT), IPv6, IPv4 to IPv6 transition. Generalized Forwarding and SDN Overview.

Module V The Network Layer: Control Plane, Routing Algorithms, Link-State (LS) Routing Algorithm, Distance-Vector (DV) Routing Algorithm. Intra-AS Routing on the Internet: OSPF, Routing Among the ISPs: BGP. The SDN Control Plane Overview. ICMP.

Module VI Link Layer - Services, Error-Detection and Correction techniques - Parity Checks, CRC, Switched Local Area Networks - Link-Layer Addressing and ARP, Ethernet, Virtual Local Area Networks

(VLANs), Trends in Data Center Networking.

Text Books:

1. James F Kurose and Keith W Ross, "Computer Networking: A Top-Down Approach", Eighth Edition, Pearson Education.
2. Behrouz A. Forouzan , Firouz Mosharraf, "Computer Networks: A Top-Down Approach", Standard Edition, McGraw Hill.

References:

1. Tanenbaum, 'Computer-networks' 4th-edition"
2. Behrouz A Forouzan,' Data Communication and Computer networks', 4th ed, McGraw Hill
3. Achyut S Godbole, 'Data communications and network's, McGraw-Hill, Second ed

SEMESTER II	COURSE CODE :CS 522	CREDIT :4

SOFTWARE ENGINEERING

Course Outcomes		CL	KC
CO1	Recall the key concepts of Professional Software Development, Software Engineering Diversity, and Software Engineering Ethics	R	F
CO2	Understand the principles of Agile development techniques and their relevance in modern software development	U	C
CO3	Apply Requirements Modeling techniques to analyze and document software requirements for a given case study	Ap	P
CO4	Create an Agile development plan for a hypothetical project, considering architectural design decisions and requirements engineering processes	Cr	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create, KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module 1: Introduction to Software Engineering - Professional Software Development, Software Engineering, Software Engineering Diversity, Internet Software Engineering, Software Engineering Ethics. Case Study - Weather Station, Digital Learning Environment**.**

Module II: Software Processes - Software Process Models, Activities- Software Specification, Software Design and Implementation, Software Validation, Software Evolution, Coping with Change, Process Improvement.

Module III: Introduction to Agile methods - Agile development techniques. Software Requirement specification-Functional and non-functional requirements, Requirements engineering processes, Requirements elicitation and specification. Architectural design decisions. Architectural views and patterns.

Module IV: Requirements Modeling - Scenario Based Modeling, Class based modeling, Functional Modeling, Behavioral modeling.

Module V: Implementation issues - reuse, configuration management, Host-target development, Open-source development. Configuration Management - Version Management, Change Management, Release Management.

Module VI: Software Testing - Development testing - Unit testing, Component testing, System testing, User testing. Service Oriented Software Engineering - Architecture, RESTful services, Service interface design and implementation.

Text Books

1. Sommerville, I. (2017). Software Engineering, 10th Edition. Pearson Education.

References

1. Maxim, B. R., Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach, 8th Edition. McGraw-Hill Education.
2. Sommerville, I. (2019). Engineering Software Products: An Introduction to Modern Software Engineering. Pearson Education.
3. Jalote, P. (2013). An Integrated Approach to Software Engineering. Springer.

SEMESTER II	COURSE CODE :CS 523	CREDIT :4

INTRODUCTION TO MACHINE LEARNING

Course Outcomes		CL	KC
CO1	Recall machine learning basics and types.	R	F
CO2	Grasp ML concepts and statistical measures.	U	C
CO3	Implement regression, classification, and clustering.	Ap	P
CO4	Examine model metrics and data suitability.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module 1: Introduction to Machine Learning Understanding Machine Learning: Definition and Scope, Comparison: Machine Learning Vs. Traditional Programming, Basics of Machine Learning: How it Works, Applications of Machine Learning, Feature Selection Techniques, Understanding Data: Descriptive Statistics (Mean, Median, Mode), Measures of Spread, Types of Learning: Supervised, Unsupervised, Semi-supervised, Challenges in Machine Learning

Module II: Regression:-Introduction to Regression, Types of Regression: Linear, Multiple Linear, Non-Linear (Polynomial), Model Evaluation Metrics for Regression.

Module III: Classification:- Introduction to Classification, Logistic Regression, Decision Trees, Naïve Bayes Classification, Support Vector Machines (SVM), K-Nearest Neighbours (KNN), Model Evaluation Metrics for Classification.

Module IV: Clustering:- Introduction to Clustering, Requirements and Types of Data in Cluster Analysis, Major Clustering Methods: Partitioning (K-means, K-medoids, CLARANS), Hierarchical (Agglomerative, BIRCH), Density-based (DBSCAN)

Module V: Advanced Multivariate Analysis:- Dimensionality Reduction Techniques: Principal Component Analysis (PCA), Multidimensional Scaling, Evaluating Model Performance: Precision, Recall, Confusion Matrix, Cross Validation, Bootstrap Sampling, Improving Model Performance with Ensemble Learning: Bagging, Boosting

Module VI: Deep Learning:-Introduction to Deep Learning, Convolutional Neural Networks (CNN), Building Blocks of CNN, Transfer Learning and Pretrained Models, Real-world Applications of Deep Learning.

References

1. Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani

2. Python Machine Learning" by Sebastian Raschka and Vahid Mirjalili
3. C. Bishop (2010), *Pattern Recognition and Machine Learning*, Springer.
4. Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville
5. K. Murphy (2012), *Machine Learning: A Probabilistic Perspective*, MIT Press.
6. Brett Lantz, *Machine Learning with R*, Packt Publishing, 2nd Edition.
7. Tom Micheal (1997), *Machine Learning*, Mcgraw Hill
8. Simon Rogers, Mark Girolami, *A First course in Machine Learning*, CRC Press, First Indian reprint, 2015.
9. N P Padhy, *Artificial Intelligence and Intelligent Systems*, Oxford University Press, 1st Edition.
10. E. Alpayidin, *Introduction to Machine Learning*, Prentice Hall of India (2005)
11. T. Hastie, RT Ibrashiran and J. Friedman, *The Elements of Statistical Learning*,Springer 2001

SEMESTER II	COURSE CODE : CS 524 A	CREDIT :4

COMPUTER FORENSICS AND CYBER LAWS

Course Outcomes		CL	KC
CO1	Recall factual information such as types of computer forensics systems, categories of cyber crime, and salient features of the IT Act, 2000.	R	F
CO2	Understand conceptual ideas such as the nature and scope of cyber crime, jurisdiction in cyberspace, and provisions of E-commerce in Indian law.	U	C
CO3	Apply procedural knowledge in utilizing digital evidence collection techniques, forensic software and hardware, and applying cyber laws to real-world cases.	Ap	P
CO4	Analyze metacognitive aspects including the effectiveness of forensic technology, implications of cyber crime on society and economy, and the impact of case laws on cyber crime prosecution	An	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create, KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I. Computer Forensics Fundamentals -Introduction to Computer Forensics-Types of Computer Forensics Systems- Preparing for computer investigations, Vendor and Computer Forensics Services-Computer Forensics Evidence and Capture- Data Recovery- Duplication and Preservation of Digital Evidence-Digital Investigation. Understanding data recovery workstations and software, conducting an investigation, completing the case, requirements for forensic lab certification, determining the physical requirements for a computer forensics lab.

Module II .Introduction to Digital Forensics - Forensic Software and Hardware - Analysis and Advanced Tools - Forensic Technology and Practices - Forensic Ballistics and Photography - Face, Iris and Fingerprint Recognition - Audio Video Analysis.

Module III: Introduction and Overview of Cyber Crime - Nature and Scope of Cyber Crime - Categories of Cyber Crime - Property Cyber Crime. **Cyber crime issues**- Unauthorized Access to Computers - Computer Intrusions - White collar Crimes - Viruses and Malicious Code - Internet Hacking and Cracking - Virus Attacks – Software Piracy.

Module IV :Introduction to Cyber Crime Investigation - Investigation Tools – Discovery - Digital Evidence Collection - Evidence Preservation - E-Mail Investigation – Tracking - IP Tracking - E-Mail Recovery - Search and Seizure of Computers - Recovering Deleted Evidences - Password Cracking.

Module V. Introduction to cyber Law: Evolution of the IT Act, Genesis and Necessity, Salient features of the IT Act, 2000, various authorities under IT Act and their powers. Penalties & Offences, Cyber Space Jurisdiction (a) Jurisdiction issues under IT Act, 2000. (b) Traditional principals of Jurisdiction (c) Extra terrestrial Jurisdiction (d) Case Laws on Cyber Space Jurisdiction.

Module V1 E-Commerce and Laws in India (a)E – Commerce; Issues and provisions in Indian Law (b) E – Governance; concept and practicality in India (c) E – Contracts and its validity in India. **Intellectual Property Rights**, Domain Names and Trademark Disputes (a) Concept of Trademarks / in Internet Era (b) Cyber Squatting(c) Copyright in Computer Programmes (d) Concept of Patent Right (i) Relevant Provisions of Patent Act 1970

References:

1. Nelson Phillips and Enfinger Steuart, ―Computer Forensics and Investigations‖, Cengage Learning, New Delhi, 2009.
2. Kevin Mandia, Chris Prosise, Matt Pepe, ―Incident Response and Computer Forensics ―Tata McGraw -Hill, New Delhi, 2006.
3. John R. Vacca , 'Computer Forensics-Computer Crime Scene Investigation' , CHARLES RIVER MEDIA, INC.Boston, Massachusetts
4. F.Enfinger and C.Steuart B.Nelson,A.Phillips,' Guide to Computer Forensics And Investigations '
5. Raghu Santanam, M. Sethumadhavan, Mohit Virendra ,'Cyber Security, Cyber Crime and Cyber Forensics: Applications and Perspectives' , Information Science Reference.
6. Pfleeger , Charles P. and Shari L, 'Security in Computing' Upper Saddle River, NJ Prentice Hall.
7. PavanDuggal , ' Cyberlaw – The Indian Perspective ' Saakshar Law Publications. 8. Murugan S Cyber Forensics
8. Darren R. HayesA Practical Guide to Digital Forensics Investigations.

Web References:

https://onlinecourses.swayam2.ac.in/cec21_ge10/preview

https://onlinecourses.swayam2.ac.in/cec20_lb06/preview

https://www.cs.nmt.edu/~df/lectures.html

https://www.classcentral.com/course/swayam-information-security-and-cyber-forensics 23006

SEMESTER II	COURSE CODE : CS 524 B	CREDIT :4

HUMAN COMPUTER INTERACTION

Course Outcomes		CL	KC
CO1	Recall characteristics of text entry devices, positioning methods, display types, and computer memory basics.	R	F
CO2	Comprehend user-centric interface designs, usability evaluation plans, ethical Human-AI interaction strategies, and inclusive design solutions.	U	C
CO3	Implement interaction styles, ergonomic principles, usability testing methods, and universal design concepts in interface creation.	Ap	P
CO4	Examine interaction models, design processes, HCI patterns, and Human-AI interaction principles for underlying concepts and implications.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module 1:The Human: Human Memory, Thinking. Computer - Text entry devices, positioning, pointing and drawing, display devices, VR and 3D interaction, understanding computer memory speed and capacity, storage format and standards

Module II: **Interaction:** Models of interaction, Frameworks and HCI, Ergonomics, Interaction Styles. **Interaction Design:** What is design?, Process of design, user focus, scenarios, navigation design.

Module III: Principles to support usability, HCI Patterns, Evaluating a software through user participation, Web usability, usability testing.

Module IV: Universal Design: Multimodal interaction, Designing for diversity.

Module V: Human-AI interaction: Interactive machine learning, IML applications, Human-Ai interaction guidelines. Explainable AI, AI Ethics and Fairness.

Module VI: Cognitive models Goal and task hierarchies Design Focus: GOMS saves money Linguistic models The challenge of display-based systems Physical and device models Cognitive architectures.

Text Books

1. Alan Dix (2004). *Human-Computer Interaction.* 3rd edition.
2. Steve Krug, *Don't Make Me Think*, *Revisited*: A *Common Sense Approach* to *Web Usability* (Voices That Matter). 3rd Edition.

References

1. A Review of User Interface Design for Interactive Machine Learning (TIIS 2018) by Dudley and Kristensson
2. Guidelines for Human-AI Interaction (CHI 2019) by Amershi et al
3. Biases in AI Systems (Communications of ACM, 2021), Srinivasan and Chander.
4. Effects of Explanations in AI-Assisted Decision Making: Principles and Comparisons (ACM Transactions on Interactive Intelligent Systems, 2022), Wang and Yin,

SEMESTER II	COURSE CODE :CS 524 C	CREDIT :4

NATURAL LANGUAGE PROCESSING

Course Outcomes		CL	KC
CO1	Recall significant events in NLP history and identify levels of analysis in NLP, including phonetics, phonology, morphology, syntax, semantics, and pragmatics.	R	F
CO2	Comprehend foundational linguistic concepts pertinent to NLP and articulate the practical challenges and uses of NLP in various contexts	U	C
CO3	Apply text preprocessing methods and statistical techniques like N-gram models and machine learning algorithms to process and analyze textual data for NLP tasks	Ap	P
CO4	Analyze the performance of NLP models using evaluation metrics and critically assess the societal and ethical ramifications of advanced NLP technologies	An	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module 1: Introduction to Natural Language Processing (NLP)- Definition and Scope of NLP, History and Evolution of NLP, Levels of Analysis in NLP: Phonetics, Phonology, Morphology, Syntax, Semantics, Pragmatics, Challenges and Applications of NLP, Overview of Machine Learning in NLP

Module II: Fundamentals of Linguistics for NLP:- Phonetics and Phonology: Speech Sounds and Sound Systems, Morphology: Word Formation and Structure, Syntax: Sentence Structure and Grammar, Semantics: Meaning Representation and Interpretation, Pragmatics: Context and Discourse Analysis

Module III: Text Processing and Preprocessing:-Text Representation: Tokens, Types, and Corpus Analysis, Text Normalization: Tokenization, Stopword Removal, Stemming, Lemmatization, Part-of-Speech Tagging and Named Entity Recognition, Sentiment Analysis and Opinion Mining

Module IV: Statistical Methods in NLP- Probability and Information Theory Basics, Language Models: N-gram Models, Smoothing Techniques, Machine Learning Models for NLP: Naïve Bayes, Maximum Entropy Models, Evaluation Metrics for NLP Tasks: Precision, Recall, F1-score

Module V: Advanced NLP Techniques;-Sequence Labeling: Hidden Markov Models (HMMs), Conditional Random Fields (CRFs), Syntax Parsing: Dependency Parsing, Constituency Parsing, Semantic Analysis: Word Sense Disambiguation, Semantic Role Labeling, Discourse Analysis: Coreference Resolution, Coherence Modeling

Module VI: Applications and Future Directions in NLP- Practical Applications of NLP: Information Retrieval, Machine Translation, Question Answering, Emerging Trends in NLP: Deep Learning, Neural Language Models, Ethical and Societal Implications of NLP Advancements

References

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition" by Daniel Jurafsky and James H. Martin
2. Foundations of Statistical Natural Language Processing" by Christopher D. Manning and Hinrich Schütze
3. Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit" by Steven Bird, Ewan Klein, and Edward Loper
4. Dash, Niladri Sekhar Corpus Linguistics and Language Technology, New Delhi:Mittal Publications 2005.
5. James Allen, Natural Language Understanding (2nd Edition) 2nd Edition- 2017.
6. Ruslan Mitkov, The Oxford Handbook of Computational Linguistics, OxfordUniversity Press 2003.
7. Philipp Koehn , Statistical Machine Translation, 2016.
8. Geoffrey Sampson and Diana McCarthy: 2004. Corpus Linguistics: Readings in a Widening Discipline. Continuum Press.

SEMESTER II	COURSE CODE : CS 525	CREDIT :2

COMPUTER NETWORKS LAB

Course Outcomes		CL	KC
CO1	Recall Wireshark functionalities, identify HTTP GET message, analyze UDP packet fields, and count ICMP packets sent and received.	R	F
CO2	Comprehend network packet structures, explain TCP header variability, and grasp application layer protocols and DNS workings	U	C
CO3	Execute packet saving and filtering, perform ping packet capture, and implement HTTP server and client in Python.	Ap	P
CO4	Analyze TCP header length variability, compare ping response times, and evaluate congestion control principles.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

LIST OF EXPERIMENTS

1. Familiarize Wireshark - Installation, understand interfaces, start/stop capture, understand packet listing and details windows.
2. Load a webpage and inspect the packets using wireshark.
    a. Find the HTTP GET message sent.
    b. Inspect Ethernet frame, IP datagram, TCP segment, and HTTP message header information.
    c. Save the relevant packets.

    Use any http web page.

    Example: http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html
3. Do a packet trace and answer the following questions
    a. Select a UDP packet and determine the fields in UDP header
    b. Determine the length of UDP header field
    c. Select a TCP packet and determine the fields in TCP header
    d. Determine the length of TCP header field. Why are field lengths variable?
4. Send 10 ping messages to a remote server (*google.com*) and capture the packets using wireshark.
    a. Filter the ICMP packets
    b. Determine the number of ICMP packets send and received
    c. Determine the protocol number from IP datagram field

    d. Write down and compare the response times from ping command output and wireshark packets
5. Implement a simple HTTP server in Python.
6. Implement a simple HTTP client in Python for the server created above.
7. Implement a multicast server and client.
8. Build a basic chat application using TCP sockets in Python.

SEMESTER II	COURSE CODE : CS 526	CREDIT :2

MACHINE LEARNING LAB

Course Outcomes		CL	KC
CO1	Develop Python scripts to implement classification and clustering algorithms, demonstrating practical application of machine learning techniques.	Ap	P
CO2	Compare and contrast the performance of various classification algorithms and clustering methods, facilitating critical examination of their effectiveness.	An	C
CO3	Assess the effectiveness of classification models through the examination of precision, recall, and confusion matrix metrics, enabling critical evaluation of model performance.	E	M
CO4	Design Python programs for dimensionality reduction using PCA, fostering the creation of solutions for high-dimensional data analysis.	E	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create, KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

1. Implement a Python program to perform feature selection using techniques like correlation analysis and recursive feature elimination using Breast Cancer Wisconsin (Diagnostic) Data Set" available on the UCI Machine Learning Repository: Dataset
2. Write Python code to calculate descriptive statistics (mean, median, mode) and measures of spread for a given dataset. Utilize the "Iris" dataset, which is a classic dataset for learning statistics and machine learning: Dataset
3. Design a Python program to implement linear regression and multiple linear regression models using libraries like scikit-learn. Evaluate the models using appropriate evaluation metrics such as Mean Squared Error (MSE) and R-squared.Use the "Boston Housing" dataset available in scikit-learn's datasets module: Dataset in scikit-learn
4. Develop a Python script to build and evaluate logistic regression, decision tree, and Naïve Bayes classification models on a given dataset. Use evaluation metrics like accuracy, precision, recall, and F1-score. Logistic Regression, Decision Trees, Naïve Bayes: Use the "Titanic" dataset available on Kaggle: Titanic Dataset
5. Write Python code to implement a Support Vector Machine (SVM) classifier and a K-Nearest Neighbors (KNN) classifier. Compare their performance using classification metrics.Use the "Iris" dataset or the "Wine Quality" dataset.
6. Implement K-means clustering algorithm using Python to cluster a given dataset into k clusters. Visualize the clustering results using Matplotlib or Seaborn. : Use the "Iris" dataset or generate synthetic data with distinct clusters.
7. Write Python code to perform hierarchical clustering (agglomerative) on a dataset and visualize the dendrogram.Use the "Iris" dataset or generate synthetic data with hierarchical structure.
8. Develop a Python program to apply Principal Component Analysis (PCA) for dimensionality

reduction on a high-dimensional dataset. Visualize the reduced dataset using scatter plots. Use the "Wine Quality" dataset or the "Digits" dataset available in scikit-learn: Digits Dataset

9. Write Python code to evaluate model performance using precision, recall, and confusion matrix. Implement cross-validation and bootstrap sampling techniques for model validation. Use a classification dataset such as "Titanic" or "Iris."
10. Design a Python script to build a Convolutional Neural Network (CNN) model using TensorFlow or Keras for image classification. Train the model on a dataset such as CIFAR-10 or MNIST.

SEMESTER II	COURSE CODE :CS 527	CREDIT :1

CASE STUDY

COURSE OBJECTIVES:

A case study is a detailed investigation done by a student on a specific topic. It is a milestone and precursor to the final presentation of the Project. The students must implement a published article from the Research and Development area. The presentation will be oral. A faculty member is assigned by the Department council for each student to select the case. The case study report should contain the case's background, analysis, alternatives, recommendations, and implementation plan. Students can use the presentation aids to deliver the theoretical aspects of the work. The interaction with the audience, students, and faculty is beneficial for the student to strengthen the different aspects of the presentation, such as presentation skill, depth of knowledge, language and rendering, and defending the questions.

ASSESSMENT:

Continuous interaction and Discussion with guide: 20 marks

Continuous interaction and work progress will lead to a valuable contribution to the final project work.

Case study Report: 30 marks

A technical report on studies and experiments will improve your technical writing skill.

Presentation and Viva: 50 marks

The presentation skills of the students are evaluated systematically.

.

SEMESTER III	COURSE CODE : CS 531	CREDIT :4

AUTOMATA THEORY & COMPILER DESIGN

Course Outcomes		CL	KC
CO1	Recall the concepts of Automata Theory, including Formal Language, Regular Expressions, and Chomsky Hierarchy of Grammar.	R	F
CO2	Understand various parsing techniques such as Top-Down (Recursive Descent, Predictive) and Bottom-Up (Shift-Reduce, LR, SLR, CLR, LALR) parsing.	U	**C**
CO3	Analyze Context-Free Grammars (CFG), Parse Trees, and the concepts of Derivation, Ambiguity, Left Recursion, and Left Factoring in grammars.	An	C
CO4	Evaluate different representations of Intermediate Code such as DAGs, Three-Address Code, Quadruples, Triples, and Single Static Assignment (SSA) form.	E	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I :Automata Theory: Concepts of Automata Theory-Formal Language and Regular Expressions**,** Chomsky Hierarchy of Grammar , Regular Grammar, Finite Automata – DFA, NFA, Conversion of regular expression to NFA, NFA to DFA. Finite Automata with Epsilon Transitions, Eliminating Epsilon Transition, FAs & Regular Expressions, Minimization of DFA, FA with outputs

Module II :Context Free grammars : CFG , Parse Trees ,Derivation, Ambiguity in Grammar , Removal of Left Recursion , Left Factoring , Push Down Automata-Languages , Equivalence of PDA's and CFG's , Deterministic Pushdown Automata,

Module III: Turing Machines: Transition Diagrams for Turing Machines, Language of a Turing Machine Turing Machines and Halting, Multitape Turing Machines, Equivalence of OneTape and Multitape TM's, Undecidable Problems about Turing Machines

Module IV : Compiler : Phases of Compiler **,**role of Lexical Analyzer, specification & recognition of Tokens using Regular Expressions, Syntax Analysis: Parsing ,Top-Down Parsing: Recursive Descent parsing, Predictive parsing, LL(1) Grammars. Bottom-Up Parsing: Shift Reduce parsing LR , SLR , CLR & LALR parsers, Compiler Construction Tools

Module V: Syntax Directed Translation: Dependency Graphs, S-Attributed & L-Attributed Definition, Type Checking, Intermediate Code Generation: Intermediate code Representations, DAGs ,Three-Address code, Quadruples, Triples, SSA

Module VI: Code Generation: Issues in Design of Code generator, Static & Stack Allocation Basic Blocks & Flow Graphs, Optimization of Basic blocks, Simple Code Generator, Code Optimization: Principal sources of optimization, Peephole Optimization.

Text Books

1. Aho, Ullman, Ravi Sethi , 'Compilers Principles, Techniques and Tools' , Pearson Education.
2. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman ,' Introduction to automata theory, languages and computation'
3. Sipser ' Introduction to Theory of computation ' ,2nd Edition, Thomson.

Reference:

1. Andrew W.Appel, 'Modern Compiler Construction in C ', Cambridge University Press.
2. LOUDEN, ' Compiler Construction , Principles & Practice' , Thomson.

SEMESTER III	COURSE CODE :CS 532	CREDIT :4

DIGITAL IMAGE PROCESSING

Course Outcomes		CL	KC
CO1	Remember the steps involved in digital image processing to achieve procedural knowledge.	R	P
CO2	Understand the principles behind different image restoration filters to grasp conceptual knowledge.	U	**C**
CO3	Apply spatial filtering techniques for image enhancement, demonstrating procedural knowledge	Ap	P
CO4	Analyze the effects of different spatial filters on image quality, demonstrating conceptual knowledge.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module 1: Image Processing – Digital Image representation - Types of Images- Sampling and Quantization - Steps in Digital Image Processing - Applications of Image Processing, Color Models – RGB,HSI, YCbCr models

Module II: Spatial Domain: Gray level transformations – Histogram processing – Basics of Spatial Filtering–Smoothing and Sharpening Spatial Filtering, Convolution and Correlation

Module III: Frequency Domain: Introduction to Fourier Transform – Smoothing and Sharpening frequency domain filters – Ideal, Butterworth and Gaussian filters.

Module IV: Image Restoration: Noise models – Mean Filters – Order Statistics – Adaptive filters –Band reject Filters – Band pass Filters– Inverse Filtering – Wiener filtering.

Module V: Image Segmentation-Edge detection– Thresholding - Region based segmentation – Region growing – Region splitting and merging – Watershed segmentation algorithm.

Module VI: Compression: Fundamentals – Image Compression models – Error Free Compression – Lossy compression– Image Compression standards

References

1. A.K. Jain, “Fundamentals of Digital Image Processing Fundamentals of Digital Image Processing: United States Edition”, 2005.

2. Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", 3ed., PHI, 2007.
3. William K. Pratt, “Digital Image Processing: PIKS Scientific Inside”, Wiley Interscience, 4ed., 2007.
4. S.Jayaraman, S. Esaki Rajan, T.Veera Kumar, “Digital Image Processing”, Tata McGraw Hill Pvt. Ltd., 2010.
5. A. Rosenfeld and A.C. Kak, “Digital Picture Processing”.

SEMESTER III	COURSE CODE : CS 533	CREDIT :4

INFORMATION RETRIEVAL

Course Outcomes		CL	KC
CO1	Recall the Boolean retrieval model components and steps for determining term vocabulary.	R	F
CO2	Explain the impact of the web on IR and the differences between parametric and zone indexes.	U	**C**
CO3	Apply wildcard queries and blocked sort-based indexing for efficient retrieval.	Ap	P
CO4	Analyze the differences between the vector space model and other retrieval models, and analyze ethical issues in dialogue system design.	An	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module 1: Introduction, The impact of the web on IR, Boolean retrieval model, Document delineation, determining vocabulary of terms

Module II: Search structures for dictionaries, wildcard queries,spelling correction,Index construction - Blocked sort-based indexing, Single-pass in-memory indexing,Distributed indexing, Dynamic indexing

Module III: Parametric and zone indexes - weighted zone scoring, learning weights. Term frequency and weighting - Inverse document frequency, Tf-idf. Vector space model - queries as vectors, computing vector scores.

Module IV: Top K document retrieval, champion lists. Evaluation - System evaluation, standard test collections, Evaluation of ranked and unranked retrieval sets. Result snippets.

Module V: Language Models - types. Query likelihood model introduction. LM versus other models. Natural Language Processing: Pretraining, Word embedding, Word Embedding with Global Vectors (GloVe) (*Overview Only*).

Module VI: Chatbots & Dialogue Systems - Chatbots - training, Fine Tuning for Quality and Safety, Learning to perform retrieval as part of responding, Evaluation. Dialogue Systems - design, Ethical issues

Text Books

1. Christopher D. Manning. Prabhakar Raghavan. Hinrich Schütze. Introduction to Information Retrieval Cambridge University Press.

References

1. D. Jurafsky, J.H. Martin, Speech and Language Processing, 3rd ed. Jan 2022 (Online available at https://web.stanford.edu/~jurafsky/slp3/)..
2. A. Zhang, Z. C. Lipton, M. Li, A. J. Smola, Dive into Deep Learning, 2021 (Available online at https://d2l.ai/index.html).

SEMESTER III	COURSE CODE :CS 534 A	CREDIT :4

BIG DATA ANALYTICS

Course Outcomes		CL	KC
CO1	Recall the key features of Big Data analytics tools	R	F
CO2	Understand the architecture of Hadoop ecosystems and its components.	U	**C**
CO3	Apply MapReduce algorithms to process large-scale data sets efficiently.	Ap	P
CO4	Analyze the challenges faced in implementing Big Data analytics solutions.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I Big Data – Definition, Characteristics, Features – Big Data Applications –Structure of Big Data-Evolution of analytic scalability - Analytic innovation. Modern Data Analytic Tools. Characteristics of big data analytics, Need of big data analytics, Classification of analytics, Challenges to big data analytics, Importance of big data analytics.

Module II Mining data streams: Introduction To Streams Concepts – Stream Data Model and Architecture - Stream Computing – Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream –HDFS concepts– MapReduce Execution

Module III Hadoop foundation for analytics:Features, Key advantage and Versions of Hadoop, Essential of Hadoop ecosystems, RDBMS versus Hadoop, Key aspects and Components of Hadoop, Hadoop architectures.Hadoop MapReduce Introduction to MapReduce, MapReduce Architecture.

Module IV: Processing data with Hadoop using MapReduce, Developing a Map Reduce Application-How MapReduce Works, Input Phase, Mapper , Combiner, Shuffle and Sort, Reducer, Output Phase, Advantage of MapReduce, Limitations Of MapReduce

Module V NoSQL Big Data Management, MongoDB :Introduction,Types of NoSQL databases, why NoSQL , Advantages of NoSQL, Use of NoSql in industry, SQL Vs NoSQL, Introduction to MongoDB,

Module VI Hive : Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Function Hbase : HBasics, Concepts, Clients, Example, Hbase Versus RDBMS

Text Books

1. Seema Acharya and SubhashiniChellappan, “Big Data and Analytics”, Wiley IndiaPvt. Ltd, 2016.
2. Big data now –O”Reilly
3. Colleen Mccue, “Data Mining and Predictive Analysis: Intelligence Gathering and Crime

Analysis", Elsevier, 2007
4. Computing in the Apache Hadoop 2 Ecosystem", 1stEdition, Pearson Education, 2016. ISBN-13: 978-9332570351
5. Anil Maheshwari, "Data Analytics", 1st Edition, McGraw Hill Education, 2017. ISBN 13: 978-9352604180

Reference:

1. Judith Hurwitz, Alan Nugent, Dr. Fern Halper and Marcia Kaufman "Big Data" by Wiley Publications, 2014.
2. AnandRajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2012.
3. Paul Zikopoulos, Chris Eaton, Paul Zikopoulos, "Understanding Big Data: Analytics forEnterprise Class Hadoop and Streaming Data", McGraw Hill, 2011.
4. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, 2007
5. Pete Warden, Big Data Glossary, O'Reilly, 2011

SEMESTER III	COURSE CODE : CS 534 B	CREDIT :4

CLOUD COMPUTING TECHNOLOGIES

Course Outcomes		CL	KC
CO1	Recall the characteristics of different cloud deployment models (public, private, hybrid).	R	F
CO2	Explain the relationship between virtualization and cloud computing architectures.	U	**C**
CO3	Apply risk management principles to mitigate security challenges in cloud computing.	Ap	P
CO4	Analyze the impact of cloud computing on enterprise-wide risk management strategies.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I: Foundation: Cloud Computing Basics, Its Characteristics, Pros & Cons, Technologies, Seven-Step Model, Public and Private Cloud, Cloud Infrastructure, Cloud Service and deployment Models.

Module II: Architectures: Cloud Life Cycle Model, Role of Cloud Modelling and Architectures, Reference Model, Cloud Industry Standard, Logical Architectures, Developing Holistic Cloud Computing Reference Model, Basic Principles, Model for federated Cloud Computing, Cloud Eco system model, Cloud Governance, **Virtualization:** Types, Architectures and softwares, Virtual Clustering and its pitfalls.

Module III: Virtualization in Grid and Cloud, Anatomy of Cloud Infrastructure, CPU virtualization, Network and Storage Virtualization, **Data Storage:** Enterprise data storage, Data storage Management, File Systems, Data Stores, Grid Oriented Storage (GOS), CDMI, Data Intensive Technologies for Cloud Computing, Distributed Data Storage, Applications Utilising Cloud Storage. Value of Cloud Computing, Legal Implications

Module IV: Web Services, Infrastructure Services, On-Demand Computing, Web Application Framework, Cloud Type and Services, SaaS, PaaS, IaaS, STaaS, DaaS, INaaS, Cloud Service development tool, Risks in cloud computing, Risk Management, Cloud Impact, Enterprises wide risk management, Types of Risks, Current State, Content level Security, Confidentiality, Integrity and Availability, Security Authorization and challenges, Software requirements and testing.

Module V: SOA Introduction, SOA Communication, Operation, Components of Cloud and SOA, SOA and Cloud, Tools for building Cloud, Programming in Cloud, Cloud Mashups, Cloud Tools: VMWARE, EUCALYPTUS, CLOUDSIM, OPENNEBULA, NIMBUS, Microsoft Cloud Services, Google Cloud Applications.

Module VI: Amazon web components and services, Elastic Compute Cloud (EC2), Amazon Storage System and database services, Cloud Based Solutions, Cloud Computing Services, Future trends of cloud computing,

Mobile Cloud computing, Automatic Cloud Engine, Multimedia Cloud, Energy awareness, Jungle Computing

Text books

1. Cloud Computing, A practical approach for learning and implementation, A.Srinivasan&J.Suresh, Pearson, 2017
2. Cloud computing a practical approach - Anthony T.Velte Toby J. Velte Robert Elsenpeter, TATA McGraw-Hill , New Delhi – 2010
3. Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online - Michael Miller - Que 2008

References

1. Cloud Computing (Principles and Paradigms), Edited by RajkumarBuyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, Inc. 2011

SEMESTER III	COURSE CODE : CS 534 C	CREDIT :4

MINING MASSIVE DATASETS

Course Outcomes		CL	KC
CO1	Remember the definition of Data Mining and its statistical limits, demonstrating factual knowledge	R	F
CO2	Understand the concept of Similarity Search, including techniques like shingling of documents and distance measure	U	**C**
CO3	Apply advanced algorithms such as PageRank and CURE for Link Analysis and clustering in non-Euclidean spaces, demonstrating procedural knowledge	Ap	P
CO4	Analyze the effectiveness of online advertising strategies, recommendation systems, and social-network graph mining algorithms	An	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I: Data Mining-Introduction-Definition of Data Mining-Statistical Limits on Data Mining, MapReduce and the New Software Stack-Distributed File Systems, MapReduce, Algorithms Using MapReduce.

Module II:: Similarity Search: Finding Similar Items-Applications of Near-Neighbor Search, Shingling of Documents, Similarity-Preserving Summaries of Sets, Distance Measures. Streaming Data: Mining Data Streams-The Stream Data Model , Sampling Data in a Stream, Filtering Streams.

Module III: Link Analysis-PageRank, Efficient Computation of PageRank, Link Spam Frequent Itemsets-Handling Larger Datasets in Main Memory, Limited-Pass Algorithms, Counting Frequent Items in a Stream. Clustering-The CURE Algorithm, Clustering in Non-Euclidean Spaces, Clustering for Streams and Parallelism.

Module IV: Advertising on the Web-Issues in On-Line Advertising, On-Line Algorithms, The Matching Problem, The Adwords Problem, Adwords Implementation.

Module V: Recommendation Systems-A Model for Recommendation Systems, Content-Based Recommendations, Collaborative Filtering, Dimensionality Reduction, The NetFlix Challenge.

Module VI: Mining Social-Network Graphs-Social Networks as Graphs, Clustering of Social-Network Graphs, Partitioning of Graphs, Simrank, Counting Triangles.

Text books

1. Jure Leskovec, Anand Rajaraman, Jeff Ullman, Mining of Massive Datasets, 3 rd Edition.

References

1. Jiawei Han & Micheline Kamber, Data Mining–ConceptsandTechniques3rdEditionElsevier.
2. Margaret H Dunham, Data Mining Introductory and Advanced topics, PEA.
3. Ian H.Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann.

SEMESTER III	COURSE CODE : CS 535	CREDIT :3

MINOR PROJECT

COURSE OBJECTIVES:

The project work aims to develop the work practice in students to apply theoretical and practical tools/techniques to solve real-life problems related to industry or current research. The project work can be a design project/experimental project and/or computer simulation project on any of the topics related to the field of Computer Science.

The project work is chosen / allotted individually on different topics. Work of each student shall be supervised by one or more faculty members of the department. The students shall be encouraged to do their project work in the College itself. If found essential, they may be permitted to carry out their major project outside the College.

The student is required to undertake the minor project during the third semester and the same is continued in the 4th semester (Major project).

In the Minor Project, the students are expected to select an emerging research area in the field of Computer Science. After conducting a detailed literature survey, they should compare, analyze research works done and review recent developments in the area and prepare an initial design of the work to be carried out as a Major Project. It is mandatory that the students should refer to National and International Journals and conference proceedings while selecting a topic for their Project. Emphasis should be given to introduction to the topic, literature survey, and scope of the proposed work along with some preliminary work carried out on the Project topic in Minor Project.

Minor Projects consist of preliminary work, two reviews of the work and the submission of a preliminary report. The initial review will spotlight the project's focal point, objectives, preliminary report, and its defined scope. Subsequently, the second review will assess the project's advancement, methodology/design, and lay out the forthcoming plans for completion by the end of the 4th semester.

Project Submission and Review Process

1. The student shall submit a proposal for different projects before the concerned project guide. The guide shall select and finalize one of the proposals. However if all proposals are not acceptable, he may be asked to submit new/modified proposals. The candidate shall prepare and submit a synopsis of the accepted proposal. A record of the accepted synopsis of each candidate shall be maintained in the department.
2. A detailed study of the requirements and feasibility of the proposed work shall be conducted by the candidate with the help of the project guide. A study phase report shall be presented before the assessment team within one month from the beginning of project work during the first review.

The design of proposed work shall be completed and presented before the assessment team for the second review. The design shall be finalized with suggested corrections/updates.

3. A short presentation explaining the proposed work and the candidate should present the current status of the Project work and the assessment will be made on the basis of the work and the presentation, by a panel of internal examiners in which one will be the internal guide. The examiners should give their suggestions to the students so that it should be incorporated in the major Project.

SEMESTER III	COURSE CODE : CS 536	CREDIT :2

DIGITAL IMAGE PROCESSING LAB

Course Outcomes		CL	KC
CO1	Perform Histogram Equalization in Python:	R	P
CO2	Implement Spatial Filtering (Smoothing and Sharpening):	Ap	P
CO3	Utilize Fourier Transform and Frequency Domain Filters:	Ap	P
CO4	Generate Noisy Images and Compare Noise Reduction Filters:	Cr	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create, KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

1. Write Python code to perform histogram equalization on a grayscale image.Display the original and equalized images along with their histograms.
2. Implement a smoothing spatial filter (e.g., averaging filter) and a sharpening spatial filter (e.g., Laplacian filter) using convolution
3. Apply the filters to a given grayscale image and display the results.
4. Utilize Python libraries (e.g., NumPy, OpenCV) to perform Fourier Transform on an input image.
5. Implement ideal, Butterworth, and Gaussian frequency domain filters for smoothing and sharpening. Apply these filters to a given image and visualize the effects.
6. Generate synthetic noisy images using different noise models (e.g., Gaussian, salt-and-pepper). Implement mean filters, median filters, and adaptive filters for noise reduction.
7. Compare the performance of these filters on noisy images in terms of restoration quality.
8. Implement edge detection algorithms (e.g., Sobel, Canny) using Python.
9. Implement lossless compression techniques (e.g., Huffman coding) and lossy compression techniques (e.g., JPEG compression) using Python.
10. Evaluate the trade-offs between compression ratio and image quality for different compression standards.

SEMESTER IV	COURSE CODE : CS 541	CREDIT :4

RESEARCH METHODOLOGY & REPORT WRITING

Course Outcomes		CL	KC
CO1	Recall and list the different types of research methods and statistical measures discussed in the syllabus.	R	F
CO2	Explain the significance of research in various fields and understand the principles behind statistical analysis.	U	C
CO3	Apply appropriate research methodologies and statistical techniques to analyze data and formulate research hypotheses.	Ap	P
CO4	Create comprehensive scientific reports using LaTeX, incorporating data analysis, literature review, and proper referencing.	Cr	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create, KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I : Research: Definitions, Objectives and types of research: Motivation; Research Methods, Significance of Research, overall processes involved in research, Defining and formulating the research problem, Literature review - need, importance and various sources for literature searching and information gathering. Identifying gap areas from literature review, formation of research hypothesis, Research Design, The Scientific Method- Observation Questions- Hypothesis- Experimentation, Criteria of good research.

Module II : Statistics for research: Data, information and system model, Missing frequencies, Frequency Distribution, Cumulative Frequency Distribution, Graphical Representation of data, Types of data analysis-(Descriptive and inferential concepts only), Measures of Central Tendency, dispersion, measures of symmetry, kurtosis, Linear Correlation and Linear Regression Analysis, Types of correlation, Karl Pearson's coefficient of correlation, Spearman's Rank correlation coefficient.

Module III: Introduction to R, data types and control structures in R, Reading data files (with different formats) in R. Overview of statistical functions in R Language –rnorm, mean, avg, median, var, sd, scale, sort, rank, quantile, aov, cor, lm, coefficients, confint, summary, plotting functions - hist, plot, curve and boxplot.

Module IV: Critical Communication-Presenting and publishing research work- seminar, workshop, symposium, conference. Types, need and significance of Technical writing, in computer science research. Reporting and thesis writing – writing a proposal, Structure and components of scientific reports - Types of report – Technical reports and thesis– Different steps in preparation of report – Layout – Illustrations and tables - Bibliography, Mechanics of writing research report, referencing and footnotes, Referencing styles.

Module V: Technical writing in Latex: LaTeX compilation, formatting, writing books as chapters, designing header and footer, designing chapters and sections, creating lists, tables, inserting images, setting labels and reference, index, list of figures and tables, math formulae, hyperlinks, bookmarks, bibliography

Module VI : . LaTeX Classes: article, book, report, slides, IEE Tran; Publishing research papers:- Structure of a research paper, awareness on paper publication formats-IEEE Tran, Impact factor, h, hb, g indices, research repositories- WoS & Scopus; DOI, Ethics in research, Intellectual property rights, Patents, Plagiarism and Plagiarism checking tools.

Reference:

1. Kothari, C.R., Research Methodology: Methods and Techniques. New Age International. Publishers
2. Mr. Ranjit Chitale., Statistical and Quantative Methods
3. S.P.Gupta, Statistical Methods., Sultan Chand, NewDelhi
4. Kottwiz, LaTEX for Beginners.
5. Kopka., A guide to Latex.

SEMESTER IV	COURSE CODE : CS 542 A	CREDIT :4

DEEP LEARNING

Course Outcomes		CL	KC
CO1	Recall key components of neural networks and semantic models.	R	F
CO2	Grasp the principles of gradient descent, backpropagation, and word vector representations.	U	C
CO3	Implement techniques like dropout and Named Entity Recognition in practical scenarios.	Ap	P
CO4	Construct architectures for tasks such as image captioning and encoder-decoder systems.	An	P

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I: Introduction: Feed forward Neural networks, Gradient descent and the back propagation algorithm, Unit saturation, the vanishing gradient problem, and ways to mitigate it. RelU Heuristics for avoiding bad local minima, Heuristics for faster training, Nestors accelerated gradient descent, Regularization, Dropout

Module II : Convolutional Neural Networks: Architectures, convolution/pooling layers, Recurrent Neural Networks: LSTM, GRU, Encoder Decoder architectures.

Module III: Deep Unsupervised Learning: Autoencoders, Variational Auto-encoders, Adversarial Generative Networks, Auto-encoder and DBM Attention and memory models, Dynamic Memory Models

Module IV : Applications of Deep Learning to Computer Vision: Image segmentation, object detection, automatic image captioning, Image generation with Generative adversarial networks, video to text with LSTM models, Attention Models for computer vision tasks

Module V: Applications of Deep Learning to NLP: Introduction to NLP and Vector Space Model of Semantics, Word Vector Representations: Continuous Skip-Gram Model, Continuous Bag-of-Words model (CBOW), Glove, Evaluations and Applications in word similarity

Module VI : Analogy reasoning: Named Entity Recognition, Opinion Mining using Recurrent Neural Networks: Parsing and Sentiment Analysis using Recursive Neural Networks: Sentence Classification using Convolutional Neural Networks, Dialogue Generation with LSTMs.

Text Books:

1. Deep Learning by Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT Press.
2. The Elements of Statistical Learning by T. Hastie, R. Tibshirani, and J. Friedman, Springer.
3. Probabilistic Graphical Models. Koller, and N. Friedman, MIT Press.

References

1. Bishop, C, M., Pattern Recognition and Machine Learning, Springer,2006.
2. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd,2009.
3. Golub, G.,H., and Van Loan,C., F., Matrix Computations, JHU Press,2013.
4. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education,2004.
5. Vinod Chandra S S, Anand H S - "Machine Learning: A Practitioners Approach", Prentice Hall of India, New Delhi, 2020.

SEMESTER IV	COURSE CODE : CS 542 B	CREDIT :4

OPTIMIZATION TECHNIQUES

Course Outcomes		CL	KC
CO1	Recall the basic definition, scope, objectives, phases, models, and limitations of Operations Research.	R	F
CO2	Explain the methodology of Operations Research, including linear programming problem formulation, graphical method, simplex method, artificial variables, big-M method, two-phase method, degeneracy, and unbounded solutions.	U	C
CO3	Apply solution methods for Operations Research, including transportation problem formulation, solution techniques like the Northwest corner rule, Matrix minima method, least cost method, Vogel's Approximation method, MODI method, and handling degeneracy in transportation problems.	Ap	P
CO4	Analyze game theory concepts such as competitive games, rectangular games, saddle points, minimax method of optimal strategies, value of the game, dominance principle, and mixed strategy for 2 x 2 games.	An	M

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I: Introduction to Operations Research: Basics definition, scope, objectives, phases, models and limitations of Operations Research**.**Modeling in Operations Research -Solution methods for O.R-Methodology of O.R Linear Programming Problem-Formulation-Graphical method-Simplex method Artificial variables, big-M method, two-phase method, degeneracy and unbounded solutions.Concept of Duality: Formulation of dual LPP, Duality theorem, advantages of duality, Dual simplex algorithm,-Big M method-Two phase method.

Module II: Transportation Problem-Formulation, Solution, Unbalanced Transportation Problem, Methods to find initial basic feasible solution-Northwest corner rule-Matrix minima method, least cost method, Vogel's Approximation method. Solving TP -MODI method – Degeneracy in TP-Maximization in TP.

Module III: Assignment Problem- Hungarian method of assignment-Maximization in assignment problem. Formulation. Hungarian method for optimal solution. Solving unbalanced problem. Travelling salesman problem and crew assignment problem.

Module IV: Games Theory- Competitive games, rectangular game, saddle point, minimax (maximin) method of optimal strategies, value of the game. Solution of games with saddle points, dominance principle. Rectangular games without saddle point – mixed strategy for 2 x 2 games

Module V: Network Models- Definition, Shortest Route problem, Maximum flow problem. CPM & PERT:

Network representation, Critical Path Computations

Module VI: Queuing Theory-Elements of a queuing system-Kendall's notation-Operating characteristics-Poisson process. Exponential distribution - mean and variance - Birth and death process. Queuing models based on Poisson process-Single server models with finite and infinite capacity-Multi server models with finite and infinite capacity.

References:

1. P. K. Gupta and D. S. Hira, "Operations Research", S. Chand & Co.
2. Sharma J.K., Mathematical models in operations research, TMH.
3. Thaha H.A, Operations Research, Pearson.
4. Winston, Operations Research Applications and Algorithms, Cengage.
5. L. R Potti, Operations Research.
6. Kanthi Swaroop, P.K Guptha, ManMohan, Operations Research
7. Operations Research: Principles And Practice, A. Ravindran, Don T. Phillips, James J. Solberg

SEMESTER IV	COURSE CODE : CS 542 C	CREDIT :4

CRYPTOGRAPHY AND NETWORK SECURITY

Course Outcomes		CL	KC
CO1	Recall key security attack types, encryption principles, public-key cryptography principles, and authentication algorithms	R	F
CO2	Grasp the concepts of security services, cipher block modes, RSA algorithm, and digital signatures	U	**C**
CO3	Utilize knowledge of internet standards, message authentication, key management, and primality testing in practical contexts within the domain of security	Ap	P
CO4	Examine and break down vulnerabilities like buffer overflows, analyze hash functions and authentication requirements, and scrutinize integrity checks in wireless network security for deeper understanding.	An	C

(CL- Cognitive Level: R-remember, U-understand, Ap- Apply, An-Analyse, E- Evaluate, Cr- Create,
KC - Knowledge Category: F-Factual, C- Conceptual, P-Procedural, M- Metacognitive)

COURSE CONTENTS

Module I : Introduction to Security:- Security Attacks -Interruption, Interception, Modification and Fabrication, Security Services and Mechanisms A model for Internetwork security, Internet Standards and RFCs, Buffer overflow & format string vulnerabilities, TCP session hijacking, ARP attacks, route table modification, UDP hijacking, and man-in-the-middle attacks.

Module II : conventional Encryption Principles: Conventional encryption algorithms, cipher block modes of operation, location of encryption devices, key distribution Approaches of Message Authentication, Secure Hash Functions and HMAC.

Module III: Public-Key Cryptography: Principles of Public-Key Cryptography, RSA Algorithm, Key Management, Diffie-Hellman Key Exchange Cryptographic Hash Functions- Message Authentication, Secure Hash Algorithm (SHA-512), Authentication requirements, HMAC.

Module IV: Digital Signatures: Elgamal Digital Signature Scheme. Key Management and Distribution, Network security - Transport-level Security- Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS, Secure Shell (SSH).Wireless Network Security, Integrity checks.

Module V: Authentication Algorithms: MD5 - Secure hash algorithm (SHA), Digital Signatures - authentication protocols - Digital signature standards (DSS), Introduction to Number Theory, Fermat's and Euler's Theorems, Testing for Primality Miller-Rabin Algorithm A Deterministic Primality Algorithm Distribution of Primes.

Module VI:E-Mail Security: Pretty Good Privacy, Security and cryptography in everyday life, Virus and antivirus software, Honeypots, Traffic flow security, Firewalls, Electronic Money-Encryption/Decryption in e-mail, OpenPGP, SIM card Authentication.

References:

1. William Stallings, "Cryptography and Network security Principles and Practices",Pearson/PHI.
2. Wade Trappe, Lawrence C Washington, "Introduction to Cryptography with coding theory", Pearson.
3. W. Mao, "Modern Cryptography – Theory and Practice", Pearson Education.
4. Charles P. Pfleeger, Shari Lawrence Pfleeger – Security in computing – Prentice Hall of India.
5. AtulKahate, "Cryptography and Network Security", 2ed. Tata McGraw Hill, 2003.
6. Bernard Menezes, Network Security and Cryptography-Cengage Learning India,2011.
7. Bruce Schneier, "Applied Cryptography: Protocols, Algorthms, and Source Code in C", Second Edition, John Wiley and Sons Inc, 2001.

SEMESTER IV	COURSE CODE : CS 543	CREDIT :10

MAJOR PROJECT

Main project phase-II is a continuation of project phase-I started in the third semester. There would be two reviews in the fourth semester, first in the middle of the semester and the second at the end of the semester. First review is to evaluate the progress of the work. Second review would be a pre -submission presentation before the internal assessment committee to assess the quality and quantum of the work done.

It is encouraged to prepare at least one technical paper for possible publication in journals or conferences. The project report (and the technical paper(s)) shall be prepared without any plagiarized content and with adequate citations, in the standard format specified by the University.

Guidelines for doing M. Sc. Project work (No. Ac.AII/1/48/2019)

1. Each student should do an independent original research work as a part of their project work. The nature of the work can be theoretical, experimental, simulation etc. This should not be mere duplication of previous reports. Submission of same work by more than one student is not allowed. It will be the responsibility of respective supervisors and Head of Department to ensure non duplication and plagiarism of any sort. Further the head of the department and respective guides should inform the students about the implications of plagiarism as per the UGC guideline.
2. The text of the report should be written as Normal Body Text Font size: 12, Times New Roman, Double Spacing, and justified. Paragraph Heading Font Size: 14, Times New Roman, Underlined, Left Aligned. Chapter Heading Font Size: 20, Times New Roman, Centre Aligned. The project report should not be more than 60 pages.
3. The thesis should contain an **Introduction Chapter**: which should give an introduction to the scientific problem and should provide clear motivation required to carry out the project work.
4. **Materials / Methods**: In this chapter the student should clearly mention the Materials / Methods / Characterization technique / Theoretical Background / Mathematical tools / Software etc. which are used to carry out the project work .They may also mention the institution where the work / experiment has been carried out.
5. **Results and Discussion:** In this chapter students should describe the results obtained. Also a detailed discussion of the results should be included followed by Conclusion and Future scope.
6. Brief abstract < 500 words should be included
7. Proper scales and units should be given for all graphs
8. **References**: References should be cited in the text and a uniform format should be used for all references.
9. Figures and tables should have proper caption and continuous numbering in each chapter. Figures taken from the internet /books etc. should be properly acknowledged

SEMESTER IV	COURSE CODE : CS 544	CREDIT :3

COMPREHENSIVE VIVA

It is mandatory that the Comprehensive Viva shall be conducted by separate examiners than that for Project Evaluation. The viva will be carried out by a panel of two examiners appointed by the University, of which one shall be from outside the college. Though the viva shall be based on the entire syllabus contents, the candidates may be given an opportunity to opt a set of subjects, not less than 40% of the programme. However, the candidate, in any case, shall not be asked to write answers to the questions given by the examiners.