

DHANRAJ BAID JAIN COLLEGE

(Autonomous)

Co-Educational Minority Institution

Owned & Managed by Tamil Nadu Educational and Medical Trust

Approved by Government of Tamil Nadu & Affiliated to University of Madras

Re-Accredited by NAAC with "A" Grade

Thoraipakkam, Chennai – 600 097.

DEPARTMENT OF COMPUTER SCIENCE

M.Sc. (Computer Science)



SYLLABUS

(Choice Based Credit System)

(Effective from the academic year 2026-2027)

No. of Semesters-04

Total No. of Credits: 91

CONTENTS

- i. Regulations
- ii. PO and PSO Description
- iii. PG – Template
- iv. Methods of Evaluation & Methods of Assessment
- v. Semester Index.
- vi. Subjects – Core, Elective, Non major, Skill Enhanced, Ability Enhanced, Extension Activity, Environment, Professional Competency
 - 1) Course Lesson Box
 - 2) Course Objectives
 - 3) Units
 - 4) Learning Outcome
 - 5) Reference and Text Books
 - 6) Web Sources
 - 7) PO & PSO Mapping tables

**MASTER DEGREE COURSE
M.Sc. - COMPUTER SCIENCE
SEMESTER SYSTEM WITH CREDITS
(Effective from the academic year 2026-2027)**

REGULATIONS

1. ELIGIBILITY FOR ADMISSION

A Candidates who have passed B.Sc. Computer Science, B.Sc., Computer Technology ,B.Sc., Information Technology or B.C.A degree not less than three years duration of Madras University or any other degree accepted as equivalent thereto by the Syndicate of Madras University.

2. ELIGIBILITY FOR THE AWARD OF DEGREE

A Candidate shall be eligible for the award of the degree only if he/she has undergone the prescribed course of study in our College which is affiliated to the University of Madras for a period of not less than two academic years, passed the examinations of all the Four Semesters prescribed earning 90 credits and fulfilled such conditions as have been prescribed therefor.

3. DURATION

- a) Each academic year shall be divided into two semesters. The first academic year shall comprise the first and second semesters, the second academic year comprise of the third and fourth semesters .The odd semesters shall consist of the period from June to November of each year and the even semesters from December to April of each year. There shall be not less than 90 working days for each semester.

4. COURSE OF STUDY

The Main Subject of study for Masters Degree shall consist of the following

PART-I: CORE COURSES Consisting of (a) Core Course

(b) Elective Course (c)Extra –Disciplinary Course (d) Application Lab and (e) Team Project.

PART-II: Soft skill courses and Internship program.

5. SCHEME OF EXAMINATION

Scheme of Examination shall be enclosed in APPENDIX - I

6. REQUIREMENTS FOR PROCEEDING TO SUBSEQUENT SEMESTER

- I. Candidates shall register their names for the First Semester Examination after the admission in the PG Courses.
- II. Candidates shall be permitted to proceed from the First Semester up to final semester irrespective of their failure in any of the semester examinations subject to the condition that the candidates should register for all the arrear subjects of earlier semesters along with current (subsequent) Semester Subjects.
- III. Candidates shall be eligible to go to subsequent semester, only if they earn, sufficient attendance as prescribed therefore by the college from time to time, provided in case of a candidate earning less than 50% of attendance in any one of the Semesters due to any extraordinary circumstance such as medical grounds, such candidates who shall produce Medical Certificate issued by the Authorised Medical Attendant (AMA), duly certified by the Principal of the College, shall be permitted to proceed to the next semester and to complete the Course of Study. Such candidates shall have to repeat the missed semester by re-joining after completion of Final Semester of the course, after paying the fee for the break of study as prescribed by the college from time to time.

7. PASSING MINIMUM

A candidate shall be declared to have passed in each paper/practical of the Main Subject of study wherever prescribed, if he/she secures NOT LESS THAN 50% of the marks prescribed for the examination. He/she shall be declared to have passed the whole examination, if he/she passes in all the papers and practicals wherever prescribed as per the scheme of examinations earning 90 CREDITS.

8. CLASSIFICATION OF SUCCESSFUL CANDIDATES

i) PART-I: CORE COURSES Consisting of (a) Core Course (b) Elective Course (c) Extra –Disciplinary Course (d) Application Lab and (e) Team Project.

Successful candidates passing the examinations for Core Courses together and securing the marks 1) 60 percent and above (2) 50 percent and above but below 60 percent in the aggregate of the marks prescribed for the Core Courses together shall be declared to have passed the examination in the FIRST CLASS (EXEMPLARY/DISTINCTION/FIRST CLASS) and SECOND CLASS respectively.

9. RANKING

Candidates who pass all the examinations prescribed for the course in the FIRST APPEARANCE ITSELF ALONE are eligible for Classification/Ranking/Distinction.

**MASTER DEGREE COURSE
M.Sc. - COMPUTER SCIENCE
SEMESTER SYSTEM WITH CREDITS
(Effective from the academic year 2026-2027)**

LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK FOR POSTGRADUATE EDUCATION	
Programme	M.Sc., Computer Science
Programme Code	MSCCS2078
Duration	PG - Two Years
Programme Outcomes (POS)	PO1- To possess advanced knowledge of Computing, Mathematical basics for contemporary Computing Specialization and Knowledge of defined problem domain PO2- To identify a prospective domain, review research literature and analyze the problems using mathematical methods and suggest PO3- To have the Ability to use design tools, design software as per needs and specifications. PO4- To apply acquired knowledge of the domain in investigating the software design, from design of experiments, analysis of data to provision of valid conclusions. PO5- To possess the skills to use modern software and hardware tools to analyse problems. PO6- To possess the knowledge of ethical and legal principles and cyber regulations PO7- To Possess ability for self-education and attitude for life-long learning in the broadest context of technological change PO8- To possess the ability to communicate scientific facts effectively in both verbal and written form to the society PO9- To possess the ability to understand the impact of IT solutions in a global and societal context. PO10- To possess the skill to find out the right opportunity for entrepreneurship for the betterment of an individual and society at large.
Programme Specific Outcomes (PSOs)	PSO1- Implement the concept of theory and technology

	with classical and modern techniques for solving the complex problems in Computer Science. PSO2-Be more curious towards learning new and emerging technologies that adapt quickly to changes. PSO3-Design, execute and evaluate computing projects in academia and industries using appropriate technologies. PSO4-Know the contextual knowledge in computing science research and communicate effectively with stakeholders with the society at large for enhancing the quality of life. PSO5-Be honest in upholding the ethical principles and social responsibilities along with socio-economic innovations.
--	--

METHODS OF EVALUATION		
Internal Evaluation	CAT-1 -5,	40 Marks
	CATII-5 MARKS	

(Effective from the academic year 2026-2027)

ALLOCATION OF CREDITS

Credit Distribution for PG Programme

Semester-I	Credit	Semester-II	Credit
Core-1	4	Core-6	4
Core-2	4	Core-7	4
Core - 3	4	Core - 8	2
Core - 4	2	Core-9	2
Core-5	2	Extra Disciplinary	3
Extra Disciplinary	3	Elective (Generic / Discipline Centric) -I	3
Soft Skill -1	2	Elective (Generic / Discipline Centric) -II	3
		Soft Skill -1	2
Total	21	Total	23

	ASSIGNMENT -5 MKS	
	ATTENDANCE -5 MKS	
	SEMINAR -5 MKS	
	MODEL EXAMINATIONS-10 MKS	
	ONLINE CERTIFICATE COURSE COMPLETION-5 MKS	
External Evaluation	End Semester Examination	100 Marks will be converted in to 60 Marks
Total		100 Marks

**EXTERNAL QUESTION PAPER PATTERN-
TOTAL MARKS -100 MARKS
CONDUCTED FOR 100 MARKS –CONVERSION TO 60 MARKS**

SECTION – A (30 words)

Answer Any Ten Questions out of Twelve Questions

10 X 2 = 20 marks

SECTION – B (200 words)

Answer Any FIVE Questions out of Eight Questions

5 x 7 = 35 marks

SECTION – C (500 words)

Answer any THREE Questions out of FIVE

3x 15 = 45 marks

TOTAL = 100 marks

**PROGRAMME OUTCOMES (PO) - PROGRAMME SPECIFIC OUTCOMES (PSO)
MAPPING**

PROGRAMME SPECIFIC OUTCOMES (PSO)					
	PO1	PO2	PO3	PO4	PO5
PSO1	3	3	3	3	3
PSO2	3	3	3	3	3
PSO3	3	3	3	3	3
PSO4	3	3	3	3	3
PSO5	3	3	3	3	3

Level of Correlation between PO's and PSO's

(Suggested by UGC as per Six Sigma Tool – Cause and Effect Matrix)

Assign the value

1 – Low

2 – Medium

3 – High

0 – No Correlation

S.N O	Course	SUBJECT NAME	Ins. Hours	Credits	MAX. MARKS		
SEMESTER I					Int.	Ext	Total
1	Core - 1	Advanced Data Structures and Algorithms	5	4	40	60	100
2	Core - 2	Advanced Python Programming	5	4	40	60	100
3	Core - 3	Artificial Intelligence and Machine Learning	5	4	40	60	100
4	Core - 4	Practical - 1: Advanced Data Structures and Algorithms Lab	4	2	40	60	100
5	Core - 5	Practical - 2 : Advanced Python Programming Lab	4	2	40	60	100
6	Extra Disciplinary	Information Security & Cryptography	5	3	40	60	100
7	Soft Skill-1	Communication skills for software engineers	2	2	40	60	100
		Total Credits		21			
SEMESTER II							
8	Core - 6	Soft Computing	5	4	40	60	100
9	Core - 7	Advanced Networks	5	4	40	60	100
10	Core - 8	Practical - 3: Soft Computing Lab	4	2	40	60	100
11	Core - 9	Practical - 4 : Elective II based Lab	4	2	40	60	100
	Core- 9A	Practical - 4: Full stack developmentLab (Elective II based Lab)			40	60	
	Core- 9B	Practical - 4: Advanced Java Programming Lab (Elective II based Lab)			40	60	
	Core- 9C	Practical - 4: Digital Image Processing Lab (Elective II based Lab)			40	60	
12	Extra Disciplinary	Principles of Compiler Design	4	3	40	60	100
13	Elective-I	(a) Cloud computing (or) (b) Internet of things (c) Data analytics	3	3	40	60	100
	Elective-II	(a) Full stack development (b) Advanced Java Programming (c) Digital Image processing	3	3	40	60	100
14	Soft Skill-2	General aptitude for competitive Examinations	2	2	40	60	100
	Internship	Internship during summer vacation of I year (4 to 5 weeks)					
		Total Credits		23			

QUESTION PAPER PATTERN

SECTION	UNIT-I	UNIT-II	UNIT-III	UNIT-IV	UNIT-V	TOTAL
	THEORY	THEORY	THEORY	THEORY	THEORY	
SECTION A (2 MARKS) ANSWER ANY TEN QUESTIONS 10 X 2 =20 MARKS	3	3	2	2	2	12
SECTION B (5 MARKS) ANSWER ANY FIVE QUESTIONS OUT OF EIGHT 5X 7 =35 MARKS	1	1	2	2	2	08
SECTION C (10 MARKS) ANSWER ANY 3 OUT OF 5 3 X 15=45 MARKS	1	1	1	1	1	05
TOTAL	5	5	5	5	5	25

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

Objectives:

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

Outcomes:

- Apply algorithm design principles, analyze algorithm performance using time and space complexity, and implement linear data structures.
- Design and implement tree-based data structures and analyze their operational efficiency.
- Apply graph algorithms and divide-and-conquer techniques to solve computational problems.
- Apply Greedy and Dynamic Programming techniques to solve optimization problems.
- Develop solutions using Backtracking and Branch and Bound techniques, and analyze NP-Hard and NP-Complete problems.

Syllabus**UNIT I: Introduction and Linear Data Structures**

Introduction to algorithms, characteristics of algorithms, pseudocode, performance analysis, time and space complexity, asymptotic notations (Big O, Omega, and Theta), amortized analysis, linear data structures, stacks, queues, linked lists, and their operations and applications.

UNIT II: Trees and Heaps

Introduction to trees, tree terminology, tree traversals, Binary Tree, Binary Search Tree (BST), AVL Tree, B-Tree, B+ Tree, comparison of tree structures, Fibonacci Heap, Fibonacci Heap operations (find minimum, insert, merge, extract minimum, decrease key, and delete), and complexity analysis of the above operations.

UNIT III: Graph Algorithms and Divide-and-Conquer

Graph representations, graph traversal techniques, minimum spanning trees (Kruskal's and Prim's algorithms), shortest path algorithms (Dijkstra's and Bellman-Ford algorithms), divide-and-conquer technique, Merge Sort, and Quick Sort.

UNIT IV: Greedy and Dynamic Programming

Greedy method and its applications, Job Sequencing with Deadlines, Huffman Coding, Dynamic Programming, Matrix Chain Multiplication, and 0/1 Knapsack Problem.

UNIT V: **Backtracking and Branch and Bound**

Backtracking and its applications including N-Queen and Graph Coloring problems, Branch and Bound method for the 0/1 Knapsack Problem, and an introduction to NP-Hard and NP-Complete problems.

Text Books

1. Introduction to Algorithms, MIT Press, 2022.
2. Data Structures and Algorithm Analysis in C++, Pearson, 2014.

Reference Books

1. The Algorithm Design Manual, Springer, 2020.
2. Algorithms, Addison-Wesley Professional, 2011.
3. Algorithm Design, Pearson, 2005.
4. Fundamentals of Data Structures in C++, Universities Press, 2008.

Web Resources

1. **NPTEL – Data Structures and Algorithms**
<https://nptel.ac.in>
2. **MIT OpenCourseWare – Introduction to Algorithms**
<https://ocw.mit.edu>

Mapping with Programme Outcomes:

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

S-Strong M-Medium L-Low

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

Objectives:

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

Outcomes:

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

UNIT I: Advanced Python Fundamentals

Introduction to Advanced Python – Functions and Recursion – Modules and Packages – File Handling – Exception Handling – User-Defined Exceptions – Regular Expressions – Lambda Functions – Iterators and Generators.

UNIT II: Object-Oriented Programming

Classes and Objects – Constructors and Destructors – Inheritance – Polymorphism – Operator Overloading – Decorators – Multithreading – Multiprocessing.

UNIT III: Data Processing with Python

Introduction to NumPy – Array Creation and Operations – Indexing and Slicing – Introduction to Pandas – Series and DataFrames – Data Cleaning and Filtering – Data Merging and Grouping.

UNIT IV: Database and Web Programming

Database Programming using SQLite – CRUD Operations – CSV and Excel File Handling – JSON Processing – Introduction to APIs – HTTP Requests – Web Scraping using BeautifulSoup.

UNIT V: Python Applications

GUI Programming using Tkinter – Logging and Debugging – Python Automation – Project Development – Applications of Python in Artificial Intelligence – Applications of Python in Data Science

Text Books

1. Core Python Programming – R. Nageswara Rao, Dreamtech Press, 2018.
2. Python for Data Analysis – Wes McKinney, O'Reilly Media, 2022.
3. Introduction to Machine Learning with Python – Andreas C. Müller and Sarah Guido, O'Reilly Media, 2016.

Reference Books

1. Learning Python – Mark Lutz, O'Reilly Media, 2013.
2. Python Machine Learning – Sebastian Raschka, Yuxi (Hayden) Liu and Vahid Mirjalili, Packt Publishing, 2022.
3. Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow – Aurélien Géron, O'Reilly Media, 2022.
4. Fluent Python – Luciano Ramalho, O'Reilly Media, 2022.

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

S-Strong M-Medium L-Low

Title of the Course/ Paper	Artificial Intelligence and Machine Learning		
Core – 3	I Year & I Semester	Credit: 4	

Objectives:

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

Outcomes:

- Explain the fundamental concepts of Artificial Intelligence, intelligent agents, problem-solving methods, and AI environments.
- Apply uninformed and informed search algorithms, including game-playing techniques such as Minimax and Alpha-Beta Pruning, to solve AI problems.
- Analyze and represent knowledge using predicate logic, semantic networks, frames, rule-based systems, and reasoning under uncertainty.
- Demonstrate an understanding of Machine Learning concepts, workflows, data preprocessing, feature engineering, and supervised, unsupervised, and reinforcement learning techniques.
- Develop and evaluate basic Machine Learning models using Python and Scikit-learn, and explain the architecture, components, and applications of expert systems in real-world scenarios.

Syllabus**UNIT I: Introduction to Artificial Intelligence**

Introduction to Artificial Intelligence (AI) – History and Foundations of AI – AI Problem Solving – Intelligent Agents – Agents and Environments – Rational Agents – Types of Environments – Structure of Intelligent Agents – Problem Formulation and Problem-Solving Agents.

UNIT II: Search Techniques and Game Playing

Introduction to Search Strategies – Uninformed Search (Breadth First Search, Depth First Search) – Informed Search (Hill Climbing, Best First Search, A* Algorithm) – Problem Reduction – Adversarial Search – Minimax Algorithm – Alpha-Beta Pruning – Evaluation Functions.

UNIT III: Knowledge Representation and Reasoning

Knowledge Representation – Predicate Logic – Semantic Networks – Frames – Rule-Based Knowledge Representation – Reasoning with Rules – Introduction to Uncertainty – Bayesian Reasoning – Applications of Knowledge-Based Systems.

UNIT IV: Introduction to Machine Learning

Introduction to Machine Learning (ML) – AI vs Machine Learning – Applications of Machine Learning – Types of Machine Learning (Supervised, Unsupervised and Reinforcement Learning) –

Machine Learning Workflow – Data Collection and Preprocessing – Feature Engineering – Training, Validation and Testing Datasets – Introduction to Classification and Regression.

UNIT V: Machine Learning with Python and Expert Systems

Introduction to Scikit-learn – Building Simple Machine Learning Models – Model Training and Prediction – Model Evaluation (Accuracy, Precision, Recall, F1-Score) – Introduction to Expert Systems – Components and Architecture of Expert Systems – Knowledge Acquisition – Intelligent Agents – Case-Based Reasoning – Applications of AI and Machine Learning.

Text Books

1. Artificial Intelligence: A Modern Approach – Stuart Russell and Peter Norvig, 4th Edition, Pearson Education, 2021.
2. Python Machine Learning – Sebastian Raschka, Yuxi (Hayden) Liu and Vahid Mirjalili, Packt Publishing, 2022.
3. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, O'Reilly Media

Reference Book

1. Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow – Aurélien Géron, O'Reilly Media, 2022.
2. Artificial Intelligence – Elaine Rich, Kevin Knight and Shivashankar B. Nair, McGraw-Hill
3. Machine Learning with Python Cookbook – Chris Albon, O'Reilly Media, 2023

Mapping with Programme Outcomes:

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	–	–	L	–	M	L	L	–
CO2	S	S	M	M	M	–	L	–	–	–
CO3	S	S	M	M	M	L	L	–	L	–
CO4	S	S	M	S	S	–	M	L	L	–
CO5	S	S	S	S	S	L	M	M	M	L

S-Strong M-Medium L-Low

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

Objectives:

- To provide the foundations of the practical implementation and usage of Algorithms and Data

Structures.

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

Outcomes:

- Implement linear data structures such as linked lists, stacks, and queues using C++.
- Develop tree-based data structures and perform their basic operations and traversals.
- Apply sorting and graph algorithms to solve computational problems.
- Implement algorithmic techniques such as Dynamic Programming and Backtracking to solve optimization problems.
- Design and develop applications using appropriate data structures and algorithms for real-world problem solving.

Implement the following using C++

1. Write a program to implement **Singly Linked List** operations: Creation, Insertion, Deletion, and Traversal.
2. Write a program to implement **Doubly Linked List** operations: Creation, Insertion, Deletion, and Traversal.
3. Write a program to implement **Stack** using Java Generics.
4. Write a program to implement **Queue** using Java Generics.
5. Write a program to implement **Quick Sort**.
6. Write a program to implement **Merge Sort**.
7. Write a program to create and traverse a **Binary Tree** (Preorder, Inorder, and Postorder).
8. Write a program to implement **AVL Tree** operations: Insertion, Deletion, and Traversal.
9. Write a program to implement **B-Tree** operations: Insertion and Traversal.
10. Write a program to implement **Graph Traversal** using **Breadth-First Search (BFS)** and **Depth-First Search (DFS)**.
11. Write a program to find the **Minimum Spanning Tree** using **Prim's** or **Kruskal's Algorithm**.
12. Write a program to find the **Shortest Path** using **Dijkstra's** or **Bellman-Ford Algorithm**.
13. Write a program to implement the **0/1 Knapsack Problem** using Dynamic Programming.

Recommended Texts:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++:Pearson

Education; 4th Edition,

Reference Books:

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

Web References:

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

Mapping with Programme Outcomes:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	S	M	S	L	L	L	L	–
CO2	S	M	S	M	S	L	M	–	–	–
CO3	S	S	M	S	S	–	M	L	L	–
CO4	S	S	S	S	S	L	M	L	L	–
CO5	S	M	S	S	S	M	M	M	M	L

S – Strong M – Medium L – Low – – No Correlation

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

Objectives:

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

Outcomes:

- Apply Python programming concepts, exception handling, object-oriented programming, and multithreading techniques to develop robust applications.
- Utilize the NumPy library to perform array manipulation, matrix operations, linear algebra computations, and image processing tasks.
- Analyze and manipulate datasets using the Pandas library by performing data preprocessing, transformation, merging, and statistical analysis.
- Create effective data visualizations using Matplotlib and Pandas, and demonstrate data analysis techniques for real-world datasets.
- Develop and evaluate basic machine learning models using Python libraries for data preprocessing, linear regression, and prediction of outcomes.

Advanced PYTHON Programming Lab Program List**1. Exception Handling**

a) Write a Python program to handle the following exceptions:

- i) ValueError
- ii) IndexError
- iii) NameError
- iv) TypeError
- v) ZeroDivisionError

b) Write a Python program to create and handle a user-defined exception.

c) Write a Python program to demonstrate the use of try, except, else, and finally.

2. NumPy Library

- a) Create NumPy arrays from a list and tuple with float data type.
- b) Demonstrate slicing, integer indexing, and boolean indexing.
- c) Find minimum, maximum, sum, and cumulative sum of an array.
- d) Demonstrate the use of ndim, shape, size, and dtype.

3. NumPy: Linear Algebra

- a) Find the rank, determinant, and trace of a matrix.
- b) Find the eigenvalues of a matrix.
- c) Perform matrix and vector operations (dot, inner, outer product).
- d) Solve a system of linear equations.
- e) Create a white image using NumPy.
- f) Convert a NumPy array to an image and an image to a NumPy array.
- g) Perform sorting, searching, and counting using NumPy methods.
- h) Demonstrate the use of the reshape() method.

4. Pandas Library

- a) Create a Pandas Series with labels.
- b) Create a Pandas Series from a dictionary.
- c) Create a Pandas DataFrame.
- d) Demonstrate describe(), head(), and tail() methods.
- e) Convert a Pandas DataFrame and Series into a NumPy array.
- f) Perform column selection, addition, and deletion.
- g) Perform row selection, addition, and deletion.
- h) Find the n-largest and n-smallest values in a DataFrame column.

5. Object-Oriented Programming

- a) Create a Person class with constructor and a method to calculate BMI.
- b) Demonstrate different types of inheritance.

c) Demonstrate operator overloading.

6. Multithreading

- Write a Python program to create two threads to count even numbers entered by the user.
- Write a Python program to create three threads that display "**Good Morning**", "**Hello**", and "**Welcome**" at intervals of 1, 2, and 3 seconds respectively.

Machine Learning Programs

7. Data Loading and Preprocessing

- Load a dataset using Pandas.
- Display dataset information and handle missing values.

8. Linear Regression

- Build a simple Linear Regression model.
- Predict values for new data.

Text Books

- Python Crash Course – Eric Matthes, No Starch Press, 2023.
- Python Machine Learning – Sebastian Raschka, Yuxi (Hayden) Liu and Vahid Mirjalili, Packt Publishing, 2022.

Reference Books

- Learning Python – Mark Lutz, O'Reilly Media, 2013.
- Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow – Aurélien Géron, O'Reilly Media, 2022.
- Python for Data Analysis – Wes McKinney, O'Reilly Media, 2022.
- Introduction to Machine Learning with Python – Andreas C. Müller and Sarah Guido, O'Reilly Media, 2016.

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	S	M	S	M	L	M	L	L
CO2	S	M	S	M	S	M	M	L	M	L
CO3	M	S	M	M	S	S	L	L	L	M
CO4	S	L	L	M	M	L	M	S	M	S
CO5	S	S	M	S	S	M	M	L	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Information Security and Cryptography		
Extra-Disciplinary	I Year & I Semester	Credit: 3	

Objectives:

- To acquire knowledge of Information security and attacks.
- To acquire knowledge of Cryptography and its techniques
- Describe Symmetric key ciphers.
- To acquire knowledge of Public key cryptosystems.
- To provide the ability to learn Hash functions for cryptography.

Outcomes:

- Test and evaluate security in systems and attacks
- Able to understand the various cryptography algorithms
- Compare various hash algorithms and their security features
- Analyze cryptographic applications in real-world case studies.
- Demonstrate how to secure a network. Evaluate a company's security policies and procedures.

Unit 1: Introduction: Security- Attacks- Computer criminals- Method of defense Program Security: Secure programs- Non-malicious program errors- Viruses and other malicious code- Targeted malicious code- Controls against program threats

Unit 2- Cryptography Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

Unit-3- Symmetric key Ciphers Data Encryption Standard-Block cipher principles-block cipher modes of operation-Advanced Encryption Standard (AES)-Triple DES-Blowfish-RC5 algorithm

Unit 4 Asymmetric key Ciphers Public Key Cryptography and Hash Algorithms -Principles of public key cryptosystems-The RSA algorithm- Diffie Hellman Key exchange.

Unit 5 Cryptographic Hash Functions -- Hash functions-Hash Algorithms (MD5, Secure Hash Algorithm
Case Studies on Cryptography and security: Secure Multiparty Calculation, Virtual Elections, Single sign On, Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability.

Recommended Text

M.sc. Computer Science w.e.f 2026-2027 onwards

- 1.C. P. Pfleeger, and S. L. Pfleeger, Security in Computing, Pearson Education, 4th Edition, 2003
- 2.Cryptography and Network Security”, William Stallings, Pearson Education, 8th Edition, 2023.
3. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 4th Edition.

Reference Books

- 1) Kaufman, Perlman, Speciner, Network Security, Prentice Hall, 2nd Edition, 2003
- 2) Eric Maiwald, Network Security : A Beginner’s Guide, TMH, 1999
- 3) Macro Pistoia, Java Network Security, Pearson Education, 2nd Edition, 1999
- 4) Whitman, Mattord, Principles of Information Security, Thomson, 2nd Edition, 2005

Website and e-Learning Source

- 1) <http://www.cs.gsu.edu/~cscyzq/courses/ai/aiLectures.html>
 - 2) <http://www.eecs.qmul.ac.uk/~mmh/AINotes/>
 - 3) <https://nptel.ac.in/courses/106106129>
- <https://nptel.ac.in/courses/>

Mapping with Programmers outcomes*

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

S-Strong M-Medium L-Low

Title of the Course/ Paper	Communication Skills for Software Engineers		
Soft Skill-1	I Year & I Semester	Credit: 2	

Objectives:

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

Outcomes:

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

1 Basics of Communication

1.1 Definition and process of communication

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

1.3 Communications barriers and how to overcome them

1.4 Barriers to Communication, Tools of Communication

2 Application of Grammar

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

2.2 Sentences and its types

2.3 Tenses

2.4 Active and Passive Voice

2.5 Punctuation

2.6 Direct and Indirect Speech

3 Reading Skill

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

4 Writing Skill

4.1 Picture composition

4.2 Writing paragraph 4.3

Notice writing

5 Listening and Speaking Exercises

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

Mapping with Programmers outcomes*

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	M	L	L	L
CO2	S	M	S	S	S	S	L	M	M	S
CO3	S	S	L	M	M	L	M	L	M	L
CO4	M	L	M	M	M	L	M	L	L	M
CO5	S	S	M	S	S	M	L	M	M	L

S-Strong M-Medium L-Low

Title of the Course/ Paper	Soft Computing	
Core-6	I Year & II Semester	Credit:4

Course Objective:

- Understand Neural Networks
- Learn Fuzzy Logic
- Analyze and Apply to Adaptive Neuro Fuzzy Inference System, Classification and Regression Trees, Data Clustering Algorithms.
- Analyze and Apply to Algorithms.
- Learn Artificial Intelligence

Course Outcomes:

- Understand the fundamental concepts and components of soft computing techniques such as **fuzzy logic, neural networks, and genetic algorithms.**
- Apply **fuzzy logic** and **fuzzy set theory** to solve real-world problems involving uncertainty and imprecision
- Design and implement **artificial neural networks** to learn from data and solve classification or regression problems.

Unit-1: Neural Networks : Introduction to ANS- adaline- BPN- Hopfield network- Boltzman machine- Self Organizing maps.

Unit-2: Fuzzy Logic : Fuzzy sets – Fuzzy rules and fuzzy reasoning – Fuzzy inference system – Mamdani fuzzy model – Sugeno fuzzy model – Tsukamoto fuzzy model.

Unit-3: Adaptive Neuro Fuzzy Inference System- Co active Neuro Fuzzy modelling- Classification and Regression Trees- Data Clustering Algorithms- Rule based structure- Neuro Fuzzy control 1-Neuro Fuzzy control 2- Fuzzy Decision Making.

Unit-4: Genetic Algorithms : Introduction- Implementation of GA- Reproduction- Cross over- Mutation- Coding- Fitness Scaling- Applications of GA.

Unit-5: ARTIFICIAL INTELLIGENCE: Introduction – Searching techniques – First order Logic – Forward reasoning – Backward reasoning –Semantic – Frames.

Text books:

1. James A. Freeman, David M. Skapura, "neural networks, algorithms, applications and programming techniques, Pearson Education, 1991
2. S.N. Sivanandam & S.N. Deepa, Principles of Soft Computing, Wiley Publications, 2nd Edition, 2011.
3. S. Rajasekaran & G.A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic & Genetic Algorithms, Synthesis & applications, PHI Publication, 1st Edition, 2009.
4. Jang, sun, Mitzutani, "Neuro Fuzzy and Soft computing", Prentice Hall India, 2006.
5. Stuart J. Russel, Peter Norvig, "Artificial intelligence- a novel approach". 2nd Edition, Pearson Education, 2003.

Reference Books

1. John A. Bullinaria Introduction to Neural Networks : Lecture 16, SOM fundamentals, 2004.
2. Jang, sun, Mitzutani, "Neuro Fuzzy and Soft computing", Prentice Hall India, 2006.
3. David E. Goldberg, "In Search optimization and machine learning", Pearson Education, 2005.
4. Eiben. A.E and Smith J.E, "Introduction to Evolutionary Computing, Genetic Algorithms".

Web Resources:

- <http://lcn.epfl.ch/tutorial/english/perceptron/html/learning.html>
- <https://en.wikipedia.org>
- <http://www.cs.bham.ac.uk/~jxb/NN/116.pdf>
- <https://www.calvin.edu/~pribeiro/othrlnks/Fuzzy/fuzzydecisions.htm>
- http://cse.unl.edu/~sscott/research/html/hga_conf/node3.shtml
- <http://geneticprogramming.com/ga/GAsoftware.html>
- www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt
- <https://www.calvin.edu/~pribeiro/othrlnks/Fuzzy/fuzzydecisions.htm>

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	S	M	S	M	L	M	S	L	L
CO 2	S	M	S	L	S	L	M	L	M	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	M	M	S	M	L	M	S	L	L
S-Strong	M-Medium		L-Low							

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

Objectives:

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

Outcomes:

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

Design and implement networking protocols.

- Design and implement networking applications.

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

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

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

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

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

Recommended Texts:

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

Reference Books:

1. C. Kaufman, R. Perlman, M. Speciner , Network Security. Private Communication in a Public World, Prentice Hall PTR; Second Edition;2022
2. J. F. Kurose, K. W. Ross , Computer Networking. A Top-Down Approach; Addison Wesley Longman; Fourth edition, 2008

Web References:

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

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10
CO 1	S	S	M	S	M	L	M	S	L	L
CO 2	S	M	S	L	S	L	M	L	M	S
CO 3	M	S	L	M	M	S	L	S	L	S
CO 4	L	S	S	L	S	M	S	L	S	M
CO 5	S	M	M	S	M	L	M	S	L	L
S-Strong	M-Medium		L-Low							

Title of the Course/ Paper	Practical - 3: SOFT COMPUTING LAB USING (PYTHON/MATLAB)		
Core – 8	I Year & II Semester	Credit: 2	

SOFT COMPUTING LAB USING (PYTHON/MATLAB)

Course Objectives

Master Fundamental Environments:

Familiarize students with MATLAB and Python programming platforms.

Explore Toolboxes: Introduce specialized toolboxes for Fuzzy Logic and Neural Networks

.Build Neural Networks: Teach the design and implementation of artificial neural networks.

Analyze Architectures: Compare different neural network structures and optimization algorithms.

Apply Genetic Algorithms: Demonstrate how to code and minimize complex fitness functions.

Course Outcomes

Upon successful completion of this lab course, students will be able to:

Navigate Environments: Configure and utilize MATLAB and Python programming environments smoothly.

Design Fuzzy Systems: Build and simulate fuzzy logic systems using interactive toolboxes.

Implement Neural Networks: Create functional artificial neural networks using both code and GUI tools.

Evaluate Network Structures: Choose appropriate neural network architectures for specific data patterns.

Train Algorithms: Apply and compare different training rules to optimize network weights.

Optimize Functions: Solve complex optimization problems by coding custom Genetic Algorithms.

List of Experiments:

- 1. Introduction to MATLAB/PYTHON & its environment.**
- 2. Introduction to MATLAB/PYTHON: Fuzzy Logic Toolbox, Fuzzy Logic Simulink Demos**
- 3. Introduction to MATLAB/PYTHON: Neural Network (NN) Toolbox, NN Simulink Demos**
- 4. MATLAB/PYTHON simulation: Artificial Neural Network (ANN) implementation**
- 5. MATLAB/PYTHON simulation: NN Tool Artificial Neural Network (ANN) implementation**
- 6. MATLAB/PYTHON simulation: Various structure of NN algorithms implementation**
- 7. MATLAB/PYTHON simulation: Training Algorithms of ANN.**
- 8. MATLAB/PYTHON simulation: Coding and minimizing a fitness function using GA.**

V. References:

- 1. J.S.R.Jang, C.T. Sun and E.Mizutani, Neuro-Fuzzy and Soft Computing, PHI / Pearson Education 2004.**

2. S.N.Sivanandam and S.N.Deepa, “Principles of Soft Computing”, Wiley India Pvt Ltd, 2011.

3. S.Rajasekaran and G.A.Vijayalakshmi Pai, “Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis and Applications”, Prentice-Hall of India Pvt. Ltd., 2006.

4. George J. Klir, Ute St. Clair, Bo Yuan, Fuzzy Set Theory: Foundations and Applications Prentice Hall, 1997.

5. David E. Goldberg, Genetic Algorithm in Search Optimization and Machine Learning Pearson Education India, 2013.Programs

Mapping with Programme Outcomes:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	M	S	M	M	S	S	L	L	L	M
CO4	S	L	M	M	M	L	M	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

S-Strong M-Medium L-Low

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

Objectives:

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

Outcomes:

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

Programs:

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

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

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

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

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

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

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

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

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

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

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

Recommended Texts:

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

Reference Books:

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

Web References:

1. <https://www.youtube.com/watch?v=9b9pLgaSQul>

Mapping with Programme Outcomes:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	M	M	M	L	M	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

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

Objectives:

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

Outcomes:

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

Programs

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

Recommended Texts:

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

Reference Books:

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

Web References:

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

Mapping with Programme Outcomes:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	M	M	M	L	M	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

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

Objectives:

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

Outcomes:

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

UNIT I: INTRODUCTION TO COMPILERS: Structure of a compiler – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA. UNIT II: SYNTAX ANALYSIS: Role of Parser – Grammars – Error Handling – Context-free grammars – Writing a grammar – Top Down Parsing - General Strategies Recursive Descent Parser Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser- LR (0)Item Construction of SLR Parsing Table - Introduction to LALR Parser - Error Handling and Recovery in Syntax Analyzer-YACC.

UNIT III: INTERMEDIATE CODE GENERATION: Syntax Directed Definitions, Evaluation Orders for Syntax Directed Definitions, Intermediate Languages: Syntax Tree, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking. UNIT IV: RUN-TIME ENVIRONMENT AND CODE GENERATION: Storage Organization, Stack Allocation Space, Access to Non-local Data on the Stack, Heap Management - Issues in Code Generation - Design of a simple Code Generator.

UNIT V: CODE OPTIMIZATION: Principal Sources of Optimization – Peep-hole optimization - DAG- Optimization of Basic Blocks- Global Data Flow Analysis - Efficient Data Flow Algorithm.

Recommended Texts:

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

Reference Books:

1. Randy Allen, Ken Kennedy, Optimizing Compilers for Modern Architectures: A Dependence based Approach, Morgan Kaufmann Publishers, 2002.
2. Steven S. Muchnick, Advanced Compiler Design and Implementation, Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint 2003.
3. Keith D Cooper and Linda Torczon, Engineering a Compiler, Morgan Kaufmann Publishers Elsevier Science, 2004.
4. V. Raghavan, Principles of Compiler Design, Tata McGraw Hill Education Publishers, 2010.
5. Allen I. Holub, Compiler Design in C, Prentice-Hall Software Series, 1993.

Web References:

1. https://www.youtube.com/watch?v=k4QXWFZZq1E&list=PLENQMW_c1dimxH Uu6KjuBC2rOlAaoLozF

Mapping with Programme Outcomes:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	S	M	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	M	L	M	L

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

Objectives:

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

Outcomes:

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

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

Unit II: Service Delivery and Deployment Models: Service Models (XaaS): Infrastructure as a Service (IaaS) - Platform as a Service (PaaS) - Software as a Service(SaaS) -Deployment Models: Types of cloud - Public cloud - Private cloud - Hybrid cloud – Service level agreements - Types of SLA – Lifecycle of SLA- SLA Management

Unit III: Cloud Resource Virtualization: Virtualization as Foundation of Cloud – Understanding Hypervisors – Understanding Machine Image and Instances - Managing Instances – Virtual Machine Provisioning and Service Migrations Cloud Computing Applications and Paradigms: Existing Cloud Applications and Opportunities for New Applications - Architectural Styles for Cloud Applications - Workflows: Coordination of Multiple Activities - Coordination Based on a State Machine Model: The ZooKeeper - The Map Reduce Programming Model - A Case Study: The Grep The Web Application.

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

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

Recommended Texts:

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

. Reference Books:

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

Web References:

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

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	M	S	L	M	L	M	L
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

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

Objectives:

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

Outcomes:

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

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

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

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

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

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

Billing Solutions in the Internet of Things Business Models for the Internet of Things - Network Dynamics: Population Models – Information Cascades - Network Effects - Network Dynamics:

Structural Models - Cascading Behavior in Networks - The Small World Phenomenon.

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

Recommended Texts:

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

Reference Books:

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

Web References:

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

Mapping with Programme Outcomes:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

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

Objectives:

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

Outcomes:

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

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

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

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

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

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

Recommended Texts:

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

Reference Books:

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

Web References:

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

Mapping with Programme Outcomes:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	S	S	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

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

Objectives:

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

Outcomes:

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

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

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

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

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

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

Recommended Texts:

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

Reference Books:

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

Web References:

1. https://www.alvinisd.net/cms/lib03/TX01001897/Centricity/Domain/1077/beginning_html5_and_css3.pdf
2. <https://www.tutorialspoint.com/angularjs/index.htm>
3. <https://www.youtube.com/watch?v=9b9pLgaSQul>

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

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

Objectives:

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

Outcomes:

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

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

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

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

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

algorithm.

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

Recommended Texts:

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

Reference Books:

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

Web References:

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

Mapping with Programme Outcomes:

Mapping with Programmers outcomes*

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	M	S	S	L	S	M	L	M	L	L
CO2	S	M	M	L	M	L	M	L	M	S
CO3	S	S	L	M	S	S	L	M	L	M
CO4	M	L	S	M	M	L	S	L	M	S
CO5	S	S	M	S	L	M	L	M	M	L

General aptitude for competitive Examinations	Year & Semester: I Year II Semester	
Pre-requisite	Basic knowledge in numerical ability	
Learning Objectives: (for teachers: what they have to do in the class/lab/field) • To improve the quantitative skills of the students • To prepare the students for various competitive exams		
Course Outcomes: (for students: To know what they are going to learn) CO1: To gain knowledge on LCM and HCF and its related problems CO2: To get an idea of age, profit and loss related problem solving. CO3: Able to understand time series simple and compound interests CO4: Understanding the problem related to probability, and series CO5: Able to understand graphs, charts		
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)		
Units	Contents	Required Hours
I	Numbers - HCF and LCM of numbers - Decimal fractions - Simplification - Square roots and cube roots - Average - problems on Numbers	06
II	Problems on Ages - Surds and Indices - percentage - profits and loss - ratio and proportion - partnership - Chain rule.	06
III	Time and work - pipes and cisterns - Time and Distance - problems on trains - Boats and streams - simple interest - compound interest - Logarithms - Area - Volume and surface area - races and Games of skill.	06
IV	Permutation and combination - probability - True Discount	06

	- Bankers Discount - Height and Distances - Odd man out & Series.	
V	Calendar - Clocks - stocks and shares - Data representation - Tabulation - Bar Graphs - Pie charts - Line graphs	06
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	
Learning Resources: ● Recommended Texts 1. . “Quantitative Aptitude”, R.S. AGGARWAL., S. Chand & Company Ltd., ● Web resources: Authentic Web resources related to Competitive examinations		