ST JOSEPH'S UNIVERSITY

BENGALURU-27

ST. JOSEPH'S INSTITUTE OF INFORMATION TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS

SYLLABUS FOR POSTGRADUATE PROGRAMME

M.Sc. (CS&AI)

2026-27

Objectives of the M.Sc CS & AI Programme are:

1. Master advanced machine learning algorithms, including supervised, unsupervised, and reinforcement learning.
2. Build data engineering skills for designing scalable data pipelines, databases, and distributed computing systems.
3. Develop expertise in deep learning and neural networks for applications such as image recognition and natural language processing.
4. Implement advanced natural language processing (NLP) techniques for text analysis and understanding.
5. Promote ethical AI and data science practices, focusing on fairness, transparency, and privacy.
6. Develop skills in data visualization to effectively communicate complex insights to both technical and non-technical stakeholders.
7. Gain proficiency in cloud computing platforms and tools for deploying and scaling AI/ML applications.
8. Cultivate knowledge in big data analytics and distributed computing technologies like Hadoop and Spark.
9. Develop the ability to deploy and integrate AI models into production environments with an emphasis on scalability and performance.
10. Encourage interdisciplinary collaboration and problem-solving by working on real-world AI/data science projects with domain experts.

Program Outcomes of the M.Sc CS & AI Programme are:

1. **PO1: Comprehensive Technical Expertise:** Gain in-depth understanding and sophisticated technical abilities in cyber security and artificial intelligence to empower graduates to successfully plan, execute, and oversee complex systems and technologies.

2. **PO2: Innovative Problem-Solving Abilities:** Develop student capacity to evaluate intricate issues and implement creative, data-driven, and AI-powered solutions to tackle cyber security problems and new technology developments.

3. **PO3: Research and Development Proficiency:** Equip graduates with the abilities to perform autonomous and collaborative research, contributing to developments in AI and cyber security through scientific inquiry, experimentation, and creativity.
4. **PO4: Leadership and Professional Competence**: Develop your professional ethics, project management abilities, and leadership traits to guide teams and advance technological initiatives in fast-paced, interdisciplinary settings.
5. **PO5: Effective Communication**: Develop your capacity to successfully convey intricate technical ideas, research results, and creative solutions to a variety of audiences using written, spoken, and digital media.
6. **PO6: Ethical and Legal Compliance**: Instill a solid grasp of cyber security regulations, legal frameworks, and ethical norms to guarantee that technology is used responsibly and securely while adhering to international standards.
7. **PO7: Self-Directed and Continuous Learning**: To keep up with changing technology, market trends, and new issues in cyber security and artificial intelligence, promote lifelong learning and flexibility.

SEMESTER 1

Code Number	Title	No of Hours of teaching per week	No of credits	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total marks
THEORY						
CSAI7126	Applied Mathematics for Cyber Security and Artificial Intelligence	04	04	50	50	100
CSAI7226	Design and Analysis of Algorithms	04	04	50	50	100
CSAI7326	Computer Networks and Information Security	04	04	50	50	100
CSAI7426	Cyber Law , Ethics and Criminology	04	04	50	50	100
CSAI7526	Advanced Python Programming for AI	04	04	50	50	100
PRACTICALS						
CSAI7P126	Design and Analysis of algorithm and Information Security Lab	04	02	25	25	50
CSAI7P226	Python programming Lab	04	02	25	25	50
Total Number of credits: 24						

SEMESTER 2

Code Number	Title	No of Hours of teaching per week	No of credits	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total marks
THEORY						
CSAI8126	Advanced Operating System concepts with Linux	04	04	50	50	100
CSAI8226	Cryptography Techniques	04	04	50	50	100
CSAI8326	Cloud Security	04	04	50	50	100
CSAI8426	Machine Learning Essentials	04	04	50	50	100
CSAI8526	Big Data Analytics	04	04	50	50	100
PRACTICALS						
CSAI8P126	Cryptography Techniques Lab	04	02	25	25	50
CSAI8P226	Machine Learning Techniques Lab	04	02	25	25	50
Total Number of credits: 24						

SEMESTER 3

Code Number	Title	No of Hours of teaching per week	No of credits	Continuous Internal Assessment (CIA)Marks	End Semester Marks	Total marks

THEORY						
CSAI9126	**Ethical Hacking and Penetration Testing**	**04**	**04**	**50**	**50**	**100**
CSAI9226	**Digital Forensics**	**04**	**04**	**50**	**50**	**100**
CSAI9326	**Natural Language Processing**	**04**	**04**	**50**	**50**	**100**
CSAI9426	**Deep Learning and Neural Networks with Quantum Computing**	**04**	**04**	**50**	**50**	**100**
DEPARTMENT ELECTIVE						
CSAIDE9526	**Artificial Intelligence for Cybersecurity**	**04**	**04**	**50**	**50**	**100**
CSAIDE9626	**DevOps in AI**	**04**	**04**	**50**	**50**	**100**
PRACTICALS						
CSAI9P126	**Natural Language Processing Lab**	**04**	**02**	**25**	**25**	**50**
CSAI9P226	**Digital Forensics Lab**	**04**	**02**	**25**	**25**	**50**
CSAIRM9726	**Research Paper Presentation**	**02**	**02**	**25**	**25**	**50**
Total Number of credits: 26						

SEMESTER 4

DEPARTMENT OF COMPUTER SCIENCE AND COMPUTER APPLICATIONS(PG) (2025-2026)							
Semester 4	**Code Number**	**Title**	**No of Hours of teaching per week**	**No of credits**	**Continuous Internal Assessment (CIA)Marks**	**End Semester Marks**	**Total marks**
Project	**CSAIINT 0P126**	**Industry Internship / Project Work**	32	16	200	200	400
Certification Course	**CSAI0P 226**	**Online/ Certification Course**	3	3	50	50	100
Total Number of credits: 19							

Course Code: CSAI7126	**Course Title: Applied Mathematics for Cyber Security and Artificial Intelligence**
Course Credits: 04	**Hours/Week: 04**
Total Contact Hours: 60	**Formative Assessment Marks: 50**
Exam Marks: 50	**Exam Duration: 2 Hrs**

Course Outcomes (COs):

After completing the course, students will be able to:

1. Apply advanced mathematical logic to model and analyze secure systems
2. Use linear algebra techniques in AI and cryptographic applications
3. Apply probability and statistical methods to cyber threat analysis
4. Perform statistical inference and data analytics for intelligent systems
5. Formulate and solve optimization problems in AI-driven cyber security

UNIT I: Advanced Mathematical Logic and Discrete Structures (12 Hours)

Propositional and Predicate Logic: Syntax and semantics, Quantifiers and inference rules, Proof techniques: Direct, indirect, contradiction, and induction, Boolean Algebra

:Boolean functions, Karnaugh maps, Boolean minimisation techniques and Lattices and Boolean lattices, Discrete structures: Relations and closures and Equivalence relations and partial orders, Recurrence relations and solving techniques

UNIT II: Linear Algebra and Matrix Computations (12 Hours)

Vector spaces and subspaces,Linear transformations and matrix representation, Rank, nullity, and dimension, Eigenvalues, eigenvectors, diagonalization,Quadratic forms and matrix factorization : LU, QR, and Singular Value Decomposition (SVD),

UNIT III: Advanced Probability Theory and Stochastic Models (12 Hours)

Axiomatic probability theory, Random variables (discrete and continuous), Joint distributions and marginal distributions, Conditional expectation and variance, Bayes' theorem and Bayesian inference, Stochastic processes: Markov chains and Poisson processes, Entropy and information measures.

UNIT IV: Statistical Methods and Data Analytics (12 Hours)

Sampling techniques and experimental design, Estimation theory: Point and interval estimation and Maximum likelihood estimation, Hypothesis testing: Parametric and non-parametric tests, Multivariate statistical analysis: Principal Component Analysis (PCA) and Cluster analysis, Regression analysis: Linear and logistic regression

UNIT V: Optimization, Graph Theory and Mathematical Foundations of Machine Learning (12 Hours)

Graph theory: Directed and undirected graphs and Connectivity, trees, shortest path algorithms, Combinatorial optimization, Linear and non-linear programming, Convex optimization techniques, Mathematical foundations of machine learning: Loss functions and Gradient descent and optimization methods, Game theory fundamentals

Text Books

1. **Rosen, Kenneth H.** *Discrete Mathematics and Its Applications*,8th Edition, McGraw-Hill Education, New York, 2019.
2. **Strang, Gilber,** *Linear Algebra and Its Applications*,5th Edition, Cengage Learning, Boston, 2016.
3. **Ross, Sheldon M,** *A First Course in Probability*,10th Edition, Pearson Education, New Delhi, 2019.

Reference Books

1. **Devore, Jay L,** *Probability and Statistics for Engineering and the Sciences*, 9th Edition, Cengage Learning, Boston, 2016.
2. **Bishop, Christopher M,** *Pattern Recognition and Machine Learning*, 1st Edition, Springer-Verlag, New York, 2006.
3. **Cover, Thomas M. and Thomas, Joy A,** *Elements of Information Theory*, 2nd Edition, Wiley-Interscience, Hoboken, New Jersey, 2006.
4. **Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron**, *Deep Learning* 1st Edition, MIT Press, Cambridge, Massachusetts, 2016.
5. **Stallings, William**
   Cryptography and Network Security: Principles and Practice, 8th Edition, Pearson Education, New Delhi, 2020.
6. **Boyd, Stephen and Vandenberghe, Lieven**, *Convex Optimization*, 1st Edition, Cambridge University Press, Cambridge, 2004.
7. **West, Douglas B,** *Introduction to Graph Theory*,2nd Edition, Prentice Hall, Upper Saddle River, New Jersey, 2001.

BLUE PRINT

Unit Nos.	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit 1	12	13
Unit 2	12	13
Unit 3	12	18
Unit 4	12	18
Unit 5	12	18
Total	**60**	**80**
Maximum marks for the paper = 50		

Course Code: CSAI7226	Course Title: Design and Analysis of Algorithms
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 50
Exam Marks: 50	Exam Duration: 2 Hrs

Course Objectives:

1. To develop the ability to design efficient algorithms using fundamental paradigms and apply them to cybersecurity and AI problems.
2. To analyze time and space complexity and evaluate algorithmic efficiency for large-scale secure and intelligent systems.
3. To apply graph, heuristic, and optimization algorithms in areas such as secure networking, intrusion detection, and intelligent decision-making.
4. To integrate algorithmic techniques with security and AI requirements, balancing performance, scalability, and robustness.

Course Outcomes:

1. Design and implement efficient algorithms using appropriate paradigms to solve cybersecurity and AI-related problems.
2. Analyze and compare algorithms based on time and space complexity for secure and large-scale systems.
3. Apply graph, greedy, dynamic programming, and heuristic algorithms to security applications such as secure routing, attack graphs, and intelligent threat analysis.
4. Select and justify suitable algorithms by considering performance, scalability, and security requirements in AI-driven cybersecurity systems.
5. To enable students to critically evaluate and optimize algorithms for secure, scalable, and intelligent systems in cybersecurity and AI.

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	*	*	*				*
CO2	*	*	*				*
CO3	*	*	*	*		*	*
CO4	*	*	*	*	*	*	*
CO5	*	*	*	*	*	*	*

Unit 1	INTRODUCTION	12 hours

Introduction: Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Fundamentals of the Analysis of Algorithmic Efficiency: Analysis Framework, Time and space complexity, Asymptotic Notations and Basic Efficiency Classes, Algorithm design paradigms.

Brute Force: Selection Sort and Bubble Sort.

Unit II	**DIVIDE AND CONQUER AND GREEDY METHOD**	**12 hours**

Divide and Conquer: Merge sort, Quicksort, Multiplication of Long Integers, Strassen's Matrix Multiplication.

Greedy Method: Minimal Spanning Tree (Prims and Kruskal's Algorithm), Single source Shortest Paths Problem: Dijkstra's algorithm and Bellman-Ford , Fractional Knapsack Problem.

Unit III	**SEARCH, HEURISTICS, AND ADVERSARIAL ALGORITHMS**	**10 hours**

Uninformed search: BFS, DFS, Iterative Deepening,

Informed search: Greedy Best-First Search, A*, Heuristic design: admissibility and consistency, Game-tree search: Minimax, Alpha–Beta pruning.

Unit IV	**DYNAMIC PROGRAMMING AND GRAPH THEORY**	**14 hours**

Dynamic programming: Computing a Binomial Coefficient, Warshall's and Floyd's Algorithms, 0/1 Knapsack Problem and Memory Functions. All pair Shortest path algorithms,

Graph theory: Graph representations in AI and cybersecurity, Connectivity and ordering: Connected and strongly connected components, Topological sorting (DAGs), Flow networks: Maximum flow problem, Ford–Fulkerson method (overview), Maximum flow
and minimum cut. Attack graphs and trust graphs

Unit V	**ADVANCED ALGORITHM DESIGN AND TECHN IQUES**	**12hours**

Backtracking: N-Queen's Problem, Sum of Subset Problem.

Branch-and-Bound: Travelling Salesperson Problem, Assignment Problem

Decision Trees: Decision Trees for Sorting

NP and NP-Complete Problems: Basic Concepts, Non- Deterministic Algorithms, P, NP, NP Complete, and NP-Hard classes.

TEXT BOOKS

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to algorithms (4th ed.). MIT Press.
- E. Horowitz and S. Sahani, Fundamentals of Computer Algorithms, Galgotia, New Delhi.

REFERENCES

- Aho, J. Hopcroft and J.Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley.

- S.E.Goodman and S.T.Hedetniemi, Introduction to the Design and Analysis of Algorithms, McGraw Hill.
- G.Brassard, and P.Bratley, Algorithmics, PHI.
- S.K.Basu, Design Methods and Analysis of Algorithms, PHI.

Course Code: CSAI7326	Course Title: COMPUTER NETWORKS AND SECURITY
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 50
Exam Marks: 50	Exam Duration: 2 Hrs

Course Objectives

The course aims to enable students to:

1. Understand the structure and functioning of computer networks.
2. Analyze common network threats and security vulnerabilities.
3. Learn cryptographic methods used for securing data communication.
4. Study security mechanisms such as firewalls, IDS, and IPS.
5. Apply secure practices for protecting wired and wireless networks.

Course Outcomes

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

- CO1: Explain the fundamental concepts, components, and topologies of computer networks and wireless networking.
- CO2: Analyze and contrast network models (OSI/TCP-IP) and various network protocols, including routing and application layer protocols.
- CO3: Apply cryptographic techniques (symmetric/asymmetric) and understand the architecture of firewalls and intrusion detection/prevention systems for data security.
- CO4: Identify wireless network threats and evaluate different Wi-Fi security standards (WEP, WPA, WPA2, WPA3).
- CO5: Implement strategies for secure network infrastructure management, including device hardening, access control, and patch management.

CO PO MAPPING MATRIX

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	*	*			*			
CO2	*	*	*	*	*			
CO3	*	*	*	*	*			
CO4	*	*	*	*				
CO5	*	*	*	*				

Unit I – Introduction to Computer Networks **12Hrs**

Introduction to computer networks, types of networks, network components, topologies, network models, data transmission techniques, wireless networking, basics of network security, subnetting, and emerging networking trends. Routers, switches, hubs, firewalls, load balancers, proxy servers, network interface cards (NICs), wireless access points, modems, network attached storage (NAS), VPN concentrators, and content delivery networks (CDNs).

Unit II – Network Protocols **12Hrs**

OSI Vs TCP/IP protocol suite, Ethernet,IEEE standards, Routing protocols:OSPF,Distance vector routing,ARP, DNS, DHCP, HTTP, FTP, SMTP, SNMP, ICMP, IPv4 and IPv6, BGP, TLS/SSL, and NTP.

Unit III – Data Security **12Hrs**

Introduction to Cryptography:

Security Architecture,Three aspects of security,CIA triad,security attacks, Model for security,classical encryption methods,stream ciphers :CeaserAffine polyalphabetic ciphers: Vignere , Hill ciphers.Block ciphers: Fiestel, DES,AES, block cipher modes, key distribution techniques, random number generation, hash functions, and asymmetric encryption.

Firewalls and Intrusion Detection/Prevention: Introduction to firewalls, packet filtering firewalls, stateful inspection firewalls, intrusion detection systems (IDS), intrusion prevention systems (IPS), security policies, and rule configuration.

Unit IV – Wireless Network Security **12Hrs**

Introduction to wireless security, wireless threats and vulnerabilities, Wi-Fi security standards, WEP, WPA, WPA2, WPA3, Wi-Fi Protected Setup (WPS), enterprise wireless security, and wireless security best practices.

Unit V – Secure Network Infra structure management **12Hrs**

Secure configuration management, access control and authentication, network segmentation and zoning, patch management, network device hardening, encryption and data protection, endpoint security, and security policies and procedures.

Text Books

1. William Stallings, *Cryptography and Network Security: Principles and Practice*, Pearson Education, 2017.
2. Andrew S. Tanenbaum and David J. Wetherall, *Computer Networks*, Pearson Education, 2022.

Reference Books

1. Charlie Kaufman, Radia Perlman, Mike Speciner, *Network Security: Private Communication in a Public World*, O'Reilly Media, 2011.
2. James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach*, 7th Edition, Pearson, 2017.
3. Christof Paar and Jan Pelzl, *Understanding Cryptography*, Springer, 2010.

BLUE PRINT

Unit Nos.	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit 1	12	13
Unit 2	12	13
Unit 3	12	18
Unit 4	12	18
Unit 5	12	18
Total	**60**	**80**
Maximum marks for the paper = 50		

Course Code	**CSAI7426**
Course Title	**Cyber Law, Ethics and Criminology**
Course Credit	**4**
Total Contact Hours	**60 Hrs.**

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

- **CO1:** Explain the legal framework governing cyberspace, including the IT Act, digital contracts, and jurisdictional issues.
- **CO2:** Identify, classify, and analyze various types of cyber crimes using criminological theories.
- **CO3:** Examine ethical issues related to cyber space, data privacy, intellectual property, and Artificial Intelligence.
- **CO4:** Apply legal procedures and forensic principles for handling digital evidence and cyber crime investigations.
- **CO5:** Evaluate global cyber laws, data protection regulations, and emerging legal challenges in AI and cloud computing.

Programme Outcomes (POs)

- **PO1:** Apply mathematical, scientific, and computing fundamentals to solve cyber security problems.
- **PO2:** Analyze complex cyber threats and legal challenges using critical thinking.
- **PO3:** Design secure, ethical, and legally compliant cyber systems.
- **PO4:** Apply modern tools and technologies, including AI, in cyber security and forensics.
- **PO5:** Understand professional, ethical, and legal responsibilities in cyber security practice.
- **PO6:** Communicate effectively with stakeholders and society regarding cyber risks and legal compliance.
- **PO7:** Engage in independent learning and adapt to emerging cyber laws and technologies

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	*	**	*	–	**	*	*
CO2	–	***	*	**	**	*	*
CO3	–	**	**	–	***	**	*
CO4	*	**	**	***	**	*	*
CO5	–	**	***	**	***	**	**

UNIT I: Foundations of Cyber Law (12 Hours)

Introduction to Cyber Law and Legal Framework, Evolution of Cyber Space and Cyber Jurisprudence, Information Technology Act, 2000 – Objectives and Scope, Amendments to IT Act (2008), Digital Signatures, Electronic Governance, Certifying Authorities, Cyber Contracts and E-Commerce Legal Issues, Jurisdiction issues in Cyberspace, Role of Cyber Law in AI-driven Systems.

UNIT II: Cyber Crimes and Digital Criminology (12 Hours)

Introduction to Cyber Criminology, Classification of Cyber Crimes: Crimes against Individuals, Crimes against Property, Crimes against Organizations and Crimes against Government, Cyber Stalking, Cyber Defamation, Identity Theft, Cyber Obscenity and Pornography, Financial Frauds and Cryptocurrency Crimes, Cyber Crime and Cyber Terrorism, Criminological Theories applied to Cyber Crime, Role of AI in Predicting and Preventing Cyber Crimes.

UNIT III: Cyber Ethics and Professional Responsibility (12 Hours)

Ethics in Cyberspace, Ethical Theories and Models, Privacy, Surveillance, and Data Protection Ethics, Ethical Issues in Artificial Intelligence: Bias in AI, Explainable AI and Responsible AI, Intellectual Property Rights (IPR), Software Piracy and Plagiarism, Codes of Ethics (ACM, IEEE, ISC²).

UNIT IV: Digital Evidence, Forensics and Legal Procedures (12 Hours)

Digital Evidence: Types and Characteristics, Legal Admissibility of Digital Evidence, Chain of Custody, Cyber Forensics Process Model, Role of Expert Witness, Investigation Procedures under IT Act, AI Tools in Digital Forensics, Case Studies on Cyber Crime Investigations. Phishing Email Scam Investigation, Online Banking Fraud Case and Social Media Account Hacking Case.

UNIT V: Global Cyber Laws, Data Protection and Emerging Issues (12 Hours)

International Cyber Laws and Treaties, GDPR Regulations, Indian Data Protection Laws (DPDP Act, 2023), Cyber Warfare and National Security, Legal Challenges in Cloud Computing and AI, Ethical and Legal Issues in Autonomous Systems, Future Trends in Cyber Law and AI Governance.

Text Books

1. Pavan Duggal, *Cyber Law in India*, 2nd Edition, Saakshar Law Publications, New Delhi, 2014.
2. Talwant Singh, *Cyber Law & Information Technology*, Revised Edition, Tata McGraw-Hill Education, New Delhi, 2016.
3. David S. Wall, *Cybercrime: The Transformation of Crime in the Information Age*, 2nd Edition, Polity Press, Cambridge, 2018.

Reference Books

1. Eoghan Casey, *Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet*, 3rd Edition, Academic Press (Elsevier), London, 2011.
2. Orin S. Kerr, *Computer Crime Law*, 4th Edition, West Academic Publishing, St. Paul, Minnesota, 2018.
3. Luciano Floridi, *The Ethics of Information*, 1st Edition, Oxford University Press, Oxford, 2013.
4. Luciano Floridi (Editor), *The Oxford Handbook of Information and Computer Ethics* 1st Edition, Oxford University Press, Oxford, 2010.
5. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*,1st Edition, MIT Press, Cambridge, Massachusetts, 2016.
6. William Stallings, *Cryptography and Network Security: Principles and Practice* 8th Edition, Pearson Education, New Delhi, 2020.
7. Chris Reed & John Angel (Editors), *Computer Law: The Law and Regulation of Information Technology*, 7th Edition, Oxford University Press, Oxford, 2018.
8. Paul Craig & Gráinne de Búrca, *EU Law: Text, Cases, and Materials*, 6th Edition, Oxford University Press, Oxford, 2015.

Course code :CSAI7526	Course Title: Advanced Python Programming for AI
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 50
Exam Marks: 50	Exam Duration: 02Hrs

Course Outcomes (COs):

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

CO1	Knowledge	Acquire a comprehensive understanding of fundamental Python concepts and syntax.
CO2	Understand	Develop an understanding of advanced Python concepts and techniques, including object-oriented programming, functional programming, and module/package management.
CO3	Application	Apply Python programming skills to develop practical solutions to real-world problems, including data manipulation, web scraping, GUI development, API integration, and database management.
CO4	Analysis	Analyse and evaluate different approaches and techniques in Python programming to select the most appropriate solution for a given problem.
CO5	Synthesis	Synthesize knowledge and skills acquired throughout the course to design and implement complex software projects, demonstrating creativity and innovation.
CO6	Evaluation	Create fully functional Python applications and projects that showcase proficiency in various Python libraries, frameworks, and tools, meeting the requirements of diverse stakeholders and industries.

Content	Hours
Unit 1	
Python Fundamentals **Introduction to Python**: Overview of Python programming language Installing Python and setting up development environment Basic syntax, variables, Strings and data types, Conditional statements, Loops, Loops manipulation **Data Structures and Functions:** List, Dictionary, Tuple, Set, Functions, and Parameter Passing Error Handling with try-except. **Object Oriented Programming** : Concept of class, object and instance, inheritance **Packages**: modules, importing own module as well as external modules, Understanding Packages	**12**
Unit 2	

Data Manipulation with NumPy , Pandas and Plotting with Matplotlib **Introduction to NumPy:**	**12**

Arrays, Basic array operations and slicing Introduction to Pandas data structures, Series and Data Frame, Data ingestion, manipulation, and cleaning with Pandas. **Introduction to Data Visualization:** Importance of data visualization, Introduction to Matplotlib for plotting Basic plotting: line plots, scatter plots, bar plots Customizing plots with labels, titles, and legends.	
Unit 3	
Graphical User Interface (GUI) Development Introduction to GUI Programming, GUI Development with Tkinter, Handling Events and User Input **Widget** : Advanced Widget Customization (Tkinter Wrappers: CustomTkinter, ttkbootstrap, Tkinter Canvas), Data Visualization in GUI Applications, Building Interactive Applications, GUI Integration with Data Manipulation, Deployment, and Distribution.	**12**
Unit 4	
Web Scraping and API **Web Scraping** Introduction to Web Scraping Overview of Web Scraping, HTML Fundamentals, Inspecting Web Pages, Web Scraping with Beautiful Soup, Scraping Multiple Pages and Pagination, Handling Forms and Logins, Handling Dynamic Content. **APIs** Introduction to APIs, Understanding APIs, API Documentation, Authentication and Authorization, Making API Requests and API Data Manipulation.	**12**
Unit 5	
Introduction to MongoDB and Project Work Overview of MongoDB, Introduction to PyMongo, Installation and Setup, Basic CRUD Operations, Data Modelling and Schema Design, Data Replication and Sharding, Transactions and Consistency. **Project Work and Applications**	**12**

REFERENCES:

1. “Fluent Python: Clear, Concise, and Effective Programming”, Second Edition (Grayscale Indian Edition) Paperback – 6 May 2022
2. “Python in a Nutshell Paperback” by Alex Martelli , Anna Ravenscroft, Steve Holden – 4 May 2017
3. "Python Crash Course" by Eric Matthes, 2nd Edition, published by No Starch Press in 2019.

4. "Automate the Boring Stuff with Python" by Al Sweigart, 2nd Edition, published by No Starch Press in 2019.
5. "Effective Python: 90 Specific Ways to Write Better Python" by Brett Slatkin, 2nd Edition, published by Addison-Wesley Professional in 2020.
6. Python Distilled (Developer's Library) 1st Edition-22 September 2021

BLUE PRINT

Unit Nos.	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit 1	12	13
Unit 2	12	13
Unit 3	12	18
Unit 4	12	18
Unit 5	12	18
Total	**60**	**80**
Maximum marks for the paper = 50		

Course Code : CSAI7P126	Course Title: Design and Analysis of Algorithms and Information Security Lab
Course Credits: 2	Hours/Week: 04
Total Contact Hours: 44	Formative Assessment Marks: 50
Exam Marks: 50	Exam Duration: 02Hrs

1. Program to implement Linear Search and Binary Search for detecting suspicious IP addresses in network security logs and analyze their execution time.
2. Program to implement Quick Sort for sorting large-scale network traffic or AI training data and compare their performance.
3. Program to implement Merge Sort for stable and efficient organization of encrypted data blocks in secure storage systems.
4. Program to implement Activity Selection using a Greedy approach for efficient scheduling of intrusion detection processes.
5. Program to implement Dijkstra's algorithm to determine the shortest and most secure routing path in a communication network.
6. Program to implement Prim's algorithm to design a minimum-cost secure network infrastructure.
7. Program to implement the 0/1 Knapsack algorithm using Dynamic Programming for optimal allocation of encrypted files in limited secure storage.
8. Program to find the Longest Common Subsequence (LCS) between malware signatures to measure similarity and detect variants.
9. Program to solve the N-Queens problem using Backtracking and relate it to access-control constraint satisfaction.
10. Program to experimentally analyze and compare the time complexity of two algorithms commonly used in AI data preprocessing.
11. Execution of basic network commands and network configuration
12. 2. Write a socket program for the implementation of echo.
13. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the
14. Write a program on a datagram socket for client/server to display the messages on client side, typed at the server side.
15. Write a program to implement the Token Passing algorithm.
16. To configure WPA2/WPA3 and analyze authentication using Wi-Fi Router, Wireshark
17. Write a program to encrypt and decrypt a Password.
18. Implement the substitution mono alphabetic technique by using the Caesar Cipher algorithm.
19. To configure a firewall with IDS/IPS rules to monitor, detect, and block unauthorized network access
20. To implement the Data Encryption Standard (DES) algorithm to encrypt and decrypt a given plaintext using a secret key.
21. To develop an application that secures digital information through AES-based encryption

Course Code : CSAI7P226	Course Title: Python Programming lab
Course Credits: 1.5	Hours/Week: 03
Total Contact Hours: 36	Formative Assessment Marks: 25
Exam Marks: 25	Exam Duration: 02Hrs

Program List

1. Python Basics:
    a) Write a program to calculate the factorial of a number using a recursive function.
    b) Implement a program to find the sum of all elements in a given list.
    c) Create a program to count the occurrences of each word in a text file.

2. Data Manipulation with Pandas and NumPy :
    a) Load a dataset using Pandas and display basic information such as the number of rows, columns, and data types.
    b) Perform data cleaning tasks like handling missing values and removing duplicates in a dataset.
    c) Use NumPy to create arrays and perform basic operations such as element-wise addition, subtraction, multiplication, and division.

3. Plotting with Matplotlib:
    a) Create a line plot to visualize the trend of a time-series dataset.
    b) Generate a scatter plot to analyze the relationship between two variables in a dataset.
    c) Plot multiple data sets on the same figure with customized colors, markers, and labels.

4. Object-Oriented Programming:
    a) Implement a class representing a bank account with methods to deposit, withdraw, and check balance.
    b) Create a class hierarchy representing different shapes (e.g., circle, rectangle, triangle) with methods to calculate area and perimeter.

5. Web Scraping:
    a) Write a Python program to scrape a website of your choice and extract specific information. The program should:
    - Use BeautifulSoup library for parsing HTML.
    - Extract data such as headlines, links, or any relevant information from the webpage.
    - Display the extracted information in a structured format.

6. GUI (Graphical User Interface):
    a) Design a Python GUI application using Tkinter that calculates the area of a circle. The program should accept user input for the radius, and when the user clicks a button, it should display the calculated area.
    b) Develop a GUI-based to-do list application using Tkinter. The application should allow users to add tasks, mark tasks as completed, and delete tasks from the list.
    c) Create a simple calculator using Tkinter that performs basic arithmetic operations (addition, subtraction, multiplication, division). The calculator

should have buttons for each operation and a display area to show the result of calculations.

7. Web Scraping:
    a) Write a Python script to scrape a news website and extract the headlines and summaries of the latest articles. Display this information in the console.
    b) Develop a web scraper to extract product information (such as name, price, and description) from an e-commerce website. Save this information to a CSV file.
    c) Create a Python program to scrape a weather website and extract the current temperature, humidity, and weather conditions for a specific location. Display this information in a user-friendly format.

8. API (Application Programming Interface):
    a) Develop a Python script that interacts with the OpenWeatherMap API to retrieve the current weather conditions for a given city. Display the temperature, weather description, and other relevant data.
    b) Write a program to access the Twitter API and fetch recent tweets based on a specific hashtag. Display the usernames and tweet content.

9. MongoDB:

    a) Write a Python script that connects to a MongoDB database and inserts a new document into a collection. Include fields such as name, age, and email address.

    b) Develop a program to query a MongoDB database for documents that match specific criteria (e.g., find all users aged between 20 and 30). Display the results.

    c) Create a Python script that updates a document in a MongoDB collection. For example, modify the email address of a user based on their name. Display the updated document after the modification.

10. MongoDB :

    a) Write a Python script to connect to a MongoDB database and insert documents into a collection.

    b) Implement CRUD operations (Create, Read, Update, Delete) using PyMongo library on a MongoDB collection.

    c) Develop a Flask application that interacts with MongoDB to store and retrieve user data (e.g., user profiles, blog posts).

Course Code: CSAI 8126	Course Title: ADVANCED OPERATING SYSTEM CONCEPTS WITH LINUX
Course Credits: 04	**Hours/Week: 04**
Total Hours: 60	**Formative Assessment Marks: 50**
Exam Marks: 50	**Exam Duration: 2 Hrs**

Course Objectives:

1. To understand the various functions of the Operating system.
2. To learn the concept of resources management of Operating systems.
3. To learn about various problems and solutions in distributed systems.
4. To Understand the concept of fault tolerance.

Course Outcomes:

1. CO1: Explain the fundamental structure and operations of an operating system, analyze the role of system calls, and illustrate the process model within the Linux architecture.
2. CO2: Develop and debug shell scripts using Bash to automate tasks, utilizing advanced Linux utilities, filters, and programming constructs like control structures, functions, and interrupts.
3. CO3: Analyze the key issues in distributed systems and apply theoretical foundations such as Lamport's Logical Clock and Vector Clock to solve problems related to mutual exclusion and deadlock detection.
4. CO4: Evaluate different algorithms and protocols for distributed resource management, including Distributed Shared Memory and distributed scheduling, and assess fault tolerance mechanisms like checkpointing and voting protocols.
5. CO5: Differentiate between the architectures of multiprocessor and database operating systems and critique the algorithms used for concurrency control and reliability in these specialized environments.

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	*	*	*	*	*			
CO2	*	*	*	*	*			
CO3	*	*	*	*	*			
CO4	*	*	*					
CO5	*	*	*	*		*		

Unit 1	Introduction to Operating System	12 hours
Introduction, Components of Operating System, Operating System Operations, Protection and Security. Computing Environment. Abstract View of OS: User view, System View, Operating System Services, System Calls: Concept, Types of System Calls Linux Architecture, Process model.		
Unit II	**SHELL PROGRAMMING IN LINUX**	**12 hours**
Linux Utilities - File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities. SedScripts, Operation, Addresses, Commands, Applications, awk- Execution, Fields and Records, Scripts, Operation, Patterns, Actions, Associative Arrays, String and Mathematical functions, System commands in awk, Applications. Shell programming with Bourne again shell(bash) - Introduction, shell responsibilities, pipes and Redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts. Introduction to VI Editor		
Unit III	**DISTRIBUTED OPERATING SYSTEMS**	**12 hours**
Introduction – Issues – Communication network and Primitives, – Theoretical Foundations: Inherent Limitations - Lamport's Logical Clock; Vector Clock; Causal Ordering; Global State; Cuts; Termination Detection. Distributed Mutual Exclusion Non-Token Based Algorithms – Lamport's Algorithm - Token-Based Algorithms Suzuki-Kasami's Broadcast Algorithm – Distributed Deadlock Detection – Issues Centralized Deadlock-Detection Algorithms - Distributed Deadlock-Detection Algorithms. Agreement Protocols – Classification - Solutions –Applications.		
Unit IV	**DISTRIBUTED RESOURCE MANAGEMENT AND FAULT TOLERANCE**	**12 hours**
Distributed File systems – Architecture – Mechanisms – Design Issues – Distributed Shared Memory – Architecture – Algorithm – Protocols - Design Issues. Distributed Scheduling – Issues – Components – Algorithms. Basic Concepts-Classification of Failures – Basic Approaches to Recovery; Recovery in Concurrent System; Synchronous and Asynchronous Check pointing and Recovery; Check pointing in Distributed Database Systems; Fault Tolerance; Issues - Two-phase and Non- blocking Commit Protocols; Voting Protocols; Dynamic Voting Protocols.		
Unit V	**MULTIPROCESSOR AND DATABASE OPERATING SYSTEMS**	**12 hours**

Structures – Design Issues – Threads – Process Synchronization – Processor Scheduling -Memory Management – Reliability / Fault Tolerance(two); Database Operating Systems – Introduction – Concurrency Control – Distributed Database Systems – Concurrency Control Algorithms.	
Text Book(s)	
1.	Advanced Concepts in operating systems by Mukesh Singhal and Niranjan Shivaratri, Mc. Graw Hill Publication, 2017
2.	Andrew S. Tanenbaum, Herbert Bos, "Modern Operating Systems, 5th Edition", Pearson, 2022.
3.	Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts, 10th Edition", John Wiley & Sons, Inc., 2021.
4.	Sumitabha Das, “Unix concept and Programming”, McGraw Hill education, 4th Edition, 2015.
Reference Books	
1.	William Stallings, "Operating Systems: Internals and Design Principles, 9th (or 10th) Edition", Pearson, 2021.
2.	Kenneth H. Rosen, Douglas Host, Rachel Klee, et al., "UNIX: The Complete Reference, 6th Edition", McGraw-Hill/Osborne, 2017.

BLUE PRINT

Unit Nos.	Number of Hours	Total marks for which the questions are to be asked (including bonus questions)
Unit 1	12	13
Unit 2	12	13
Unit 3	12	18
Unit 4	12	18
Unit 5	12	18
Total	**60**	**80**
Maximum marks for the paper = 50		

Course Code	**CSAI8226**
Course Title	**Cryptography Techniques**
Course Credit	**4**
Total Contact Hours	**60 Hrs.**

Course Outcomes (COs)

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

- **CO1:** Understand the fundamental concepts, goals, and mathematical foundations of cryptography.
- **CO2:** Analyze and apply classical and symmetric key cryptographic techniques.
- **CO3:** Understand and implement public key cryptographic algorithms and key exchange mechanisms.
- **CO4:** Explain cryptographic hash functions, message authentication codes, and digital signatures.
- **CO5:** Apply cryptographic techniques in secure communication protocols and emerging security applications.

Program Outcomes (POs)

- **PO1:** Apply knowledge of mathematics, science, and computing fundamentals.
- **PO2:** Analyze problems and identify appropriate computing solutions.
- **PO3:** Design, implement, and evaluate computing-based systems.
- **PO4:** Use modern tools, techniques, and technologies in computing.
- **PO5:** Understand professional, ethical, legal, and security responsibilities.
- **PO6:** Communicate effectively and work in teams.
- **PO7:** Engage in lifelong learning and adapt to emerging technologies.

Course Outcomes \ Program Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	*	*					

Course Outcomes \ Program Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO2	*	*	*				
CO3	*	*	*	*			
CO4	*	*	*	*	*		
CO5		*	*	*	*	*	*

Unit 1 — Introduction to Cryptography & Mathematical Foundations Hours: 12

History and evolution of cryptography, Information security goals: confidentiality, integrity, authentication, non-repudiation, Cryptographic terminology: plaintext, ciphertext, keys, cryptanalysis, Introduction to symmetric vs asymmetric cryptography, Mathematical tools for cryptography: Modular arithmetic, Prime numbers and factorization, Greatest common divisor, Euclid's algorithm and Finite fields

Unit 2 — Classical and Modern Symmetric Cryptosystems Hours : 12

Classical ciphers: Caesar, Shift, Substitution, Transposition, Classical Cipher Attacks, Limitations of Classical Cryptosystems. Feistel cipher structure, Triple Data Encryption Standard (DES), Blow fish and Modes of operation: ECB – Electronic Codebook,CBC – Cipher Block Chaining,CFB – Cipher Feedback, OFB – Output Feedback, CTR – Counter Mode, GCM – Galois/Counter Mode.

Unit 3 — Public-Key Cryptography (Asymmetric) Hours: 12

Principles of public-key cryptography, RSA algorithm: key generation, encryption/decryption, security assumptions, Diffie–Hellman Key Exchange, ElGamal Cryptosystem, Introduction to Elliptic Curve Cryptography (ECC).

Unit 4 — Cryptographic Hash Functions, MACs & Digital Signatures Hours: 12

Properties and applications of hash functions, MD5, SHA families (SHA-1, SHA-2, SHA-3), Message Authentication Codes (MAC), HMAC (Hash-based Message Authentication Code), Digital signatures: principles and verification and Comparison of signature schemes.

Unit 5 — Key Management Protocols

Hours:12

Key distribution and management techniques, Public Key Infrastructure (PKI) and certificates, Authentication protocols (Kerberos) Secure communication protocols, Basics of Quantum Cryptography, QKD Protocols, Quantum Key Distribution (QKD), Introduction to post-quantum cryptography, Algorithms and Real-world applications (VPNs, secure email)

Textbooks

1. William Stallings — *Cryptography and Network Security: Principles and Practice*
   • Comprehensive, widely adopted for academic courses. Slow introduction from basics to advanced protocols.
2. Jonathan Katz & Yehuda Lindell — *Introduction to Modern Cryptography*
   • Strong theoretical foundation with modern formal approaches and proofs.

Reference Books

1. Bruce Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd Edition (20th Anniversary Edition), 2015.
2. Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone, *Handbook of Applied Cryptography*, 1st Edition, 1996.
3. Christof Paar & Jan Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*, 1st Edition, 2010.
4. Behrouz A. Forouzan & Debdeep Mukhopadhyay, *Cryptography and Network Security*, 3rd Edition, 2015.

Course Code: CSAI8326	Course Title: Cloud Security
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 50
Exam Marks: 50	Exam Duration: 2 Hrs

Course Objectives

1. Introduce basic cloud computing concepts, models, architectures, and security standards.
2. Help students understand cloud security goals, privacy issues, and compliance requirements.
3. Familiarize students with cloud threats, attacks, and intrusion detection techniques.
4. Provide hands-on knowledge of cloud attack and security tools through case studies.
5. Enable students to understand VM, hypervisor, and container security and apply suitable defense strategies.

Course Outcomes (COs)

CO1. Explain cloud computing fundamentals, service/deployment models, architectures, vulnerabilities, cloud security concepts and standards.

CO2. Analyze cloud security goals and privacy issues including CIA triad, access control, identity management, SLA, and compliance requirements.

CO3. Identify threat models, cloud attack taxonomies, and classify intrusion detection systems and techniques used in cloud environments.

CO4. Apply cloud attack and security tools, including VM, VMM, and network-level tools, and analyze real-world case studies such as LibVMI.

CO5. Evaluate virtual machine, hypervisor, and container security mechanisms and propose suitable defense strategies for practical cloud scenarios.

CO/PO Mapping Table

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	*	*	*		*	*	*
CO2	*	*	*		*	*	*
CO3	*	*	*		*		*
CO4	*	*	*	*	*		*
CO5	*	*	*	*	*	*	*

UNITS	Teaching Hours
Unit 1: Cloud Computing and Security Fundamentals	**12 Hrs**
Introduction to Cloud Computing, SPI Framework, Cloud Service models, Cloud Deployment Models, Cloud Service Platforms, Cloud Reference Architecture. Vulnerabilities present in cloud, Need of cloud security, **Cloud Security Concepts:** Multi-tenancy, Virtualization, Data outsourcing, Trust management, Metadata security. **Cloud Security Standards:** Control objectives for information and related technology (COBIT), ISO/IEC20000, Cloud security alliance (CSA) cloud controls matrix, CSA Cloud Reference Model, NIST Cloud Reference Model.	
Unit 2 : Cloud Security and Privacy Issues	**12Hrs**
Cloud Security Goals / Concepts: Confidentiality, Integrity, Availability (CIA triad), Authentication, Authorization, Auditing and Access control. **Cloud Security Issues:** Application-level security issues, Network level security issues, Virtualization level security issues, Data security, Identity management and access control, Improper cryptographic keys management, Service level agreement (SLA), Regular audit and compliances, Cloud and CSP migration, SLA and trust level issues, and Hardware-level security issues. **Security Requirements for Privacy:** Fine-grained access control, Privacy-preserving and Collision resistance. Privacy Issues in Cloud: Defining roles to actors, Compliance, Legal issues and multi-location issues, Privacy issues on CIA, Protection of the data, User control lacking and Data movement.	
Unit-3: Threat Model, Cloud Attacks, and Intrusion detection Techniques	**12 Hrs**
Threat Model: Type of attack entities, Attack surfaces with attack scenarios. A Taxonomy of Attacks: VMAT, VMMAT, HWAT, VSWAT and TENAT **Classification of Intrusion Detection Systems in Cloud:** TVM-based Intrusion Detection System. Hypervisor-based Intrusion Detection System. Network-based Intrusion Detection System. Distributed Intrusion Detection Systems. **Intrusion Detection Techniques in Cloud:** Misuse detection techniques, Anomaly detection techniques, Virtual machine introspection (VMI) techniques, Hypervisor introspection-based techniques, and Hybrid techniques. **Case Study:** Description of Features for Attack Analysis Based on Dataset including Fuzzers, Analysis, Backdoor, Exploits, Generic, Reconnaissance, Shellcode, and Worms.	
UNIT-4: Cloud Attack and Security Tools	**12 Hrs**

Attack Tools: Network-level attack tools, VM-level attack tools, and VMM attack tools. **Security Tools :** Network security tools, VM security tool, and VMM security tools. **Case Study of LibVMI:** A Virtualization-Specific Tool including Check the system configurations, Install KVM and necessary dependencies, Creating a virtual machine, Install LibVMI tool and necessary dependencies	
UNIT–5: Virtual Machine, Hypervisor and Container Security	**12 Hrs**
Virtual Machine Introspection (VMI): VM hook based, VM-state information based, Hypercall verification based, Guest OS kernel debugging based, and VM interrupt analysis based. Hypervisor Introspection (HVI): Nested virtualization, Code integrity checking using hardware-support, Memory integrity checking using hardware/software support, Revisiting the VMM design, and VM-assisted hypervisor introspection. **Container Security:** Threat Model in Containerized Environment including Attacks in containers. Defense Mechanisms. **Case Study:** SQL Injection Attack in Containers.	
Text Book: 1. Cloud Security: Attacks, Techniques, Tools and Challenges, Preeti Mishra, Emmanuel S. Pilli, R. C. Joshi, *1st Edition*, CRC Press, 2024.	
Reference Books: 1. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, *1st Edition*, Wiley Publishing, 2010. 2. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, *1st Edition*, O'Reilly Media, 2009. 3. Cloud Computing Security: Strategies and Best Practices, N. Agrawal, R. Kumar, S. Tapaswi, *1st Edition*, CRC Press / Routledge, 2024.	

Course Code: CSAI8426	**Course Title: Machine Learning Essentials**
Course Credits: 04	**Hours/Week: 04**
Total Hours: 60	**Formative Assessment Marks: 50**
Exam Marks: 50	**Exam Duration: 2 Hrs**

Course Objectives:

1. To understand the basic concepts of Machine Learning.
2. To understand and build the supervised and unsupervised learning models.
3. To learn and understand the concept of neural networks and deep learning.

Course Outcomes:

1. Identify the basic concepts of Machine Learning and Training model.
2. Identify and apply the appropriate machine learning techniques for classification.
3. Analyze and apply unsupervised learning techniques for solving real time problems.
4. Analyze the concept of Neural Network.
5. Analyze and apply the concepts of Markov Decision Process in HMM and Reinforcement Learning.

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	*							
CO2	*	*	*					
CO3	*	*	*					
CO4	*	*	*					
CO5	*	*	*	*		*		

Unit 1	Introduction to Machine Learning	12 hours
Fundamentals of Machine Learning – Applications -Types of Machine Learning – Challenges of Machine Learning – Testing and Validating. Training a ML model-End-to-End Machine Learning Project – Working with Real Data – Get the Data – Explore and Visualize the Data – Prepare the Data for Machine Learning Algorithms, Cross-Validation and Resampling Method,Measuring Classifier Performance.		
Unit II	**Classification and Regression**	**12 hours**
Supervised Learning- Regression –Linear Regression – Logistic Regression -Support Vector Machine – Naive Bayes – Decision Tree – KNN algorithm . **Case Study:** Loan Approval Prediction Using Decision Tree.		
Unit III	**Unsupervised Learning**	**12 hours**
Introduction to clustering- Categories of Clustering-K-means clustering - Limits of K-means–Hierarchical clustering- agglomerative Clustering-Divisive Clustering-Expected maximization Algorithm. Dimensionality Reduction: Introduction, subset selection, principal component analysis(PCA) . **Case Study**: Customer Segmentation Using K-Means Clustering.		
Unit IV	**Artificial Neural Network**	**12 hours**
Biological to Artificial Neurons – Logic Computations with Neurons – Perceptron - Multilayer Perceptron and Back propagation. Case Study: Handwritten Digit Recognition Using Multilayer Perceptron (MLP).		
Unit V	**Hidden Markov Models and Reinforcement Learning**	**12 hours**
Definition of Markov Processes,The Hidden Markov Model Structure-Hidden states vs observed variables, Transition and emission probabilities, Initial state distributions,Key Problems in HMMs- Computing the probability of an observed sequence, Inferring most likely hidden state sequences (e.g., Viterbi algorithm), Estimating model parameters (e.g., Baum-Welch algorithm/MM algorithm) . Introduction to Reinforcement Learning- Formalizing agents, environments, actions, rewards, Distinction from supervised/unsupervised learning. **Case Study:** Speech Recognition Using Hidden Markov Models (HMMs).		
Text Book(s)		
1.	Ethem Alpaydin, “Introduction to Machine Learning”, 2020, Fourth Edition, MIT Press.	
2.	Aurelien Geron, “Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow”, 2019, 2nd Edition,O'Reilly Media, Inc.	
Reference Books		
1.	Stephen Marsland, “Machine Learning: An Algorithmic Perspective “,2014, Second Edition”, CRC Press.	

2.	Tom M. Mitchell,"Machine Learning",2021,by McGraw-Hill.

Course Code: CSAI8526	Course Title: Big Data Analytics
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 50
Exam Marks: 50	Exam Duration: 2 Hrs

Course Objectives:

• Define big data and its workflows with Hadoop

• Understanding HDFS and MapReduce

• Applying Python for data streaming and MapReduce, Counting Bigrams and Overview of Pig for Machine Learning techniques.

• Data Analysis with Hive, Hbase, MySQL, FLume

• Analyze case studies for big data analytics with Sparks

Course Outcomes:

• Understand the concept of big data and its challenges, describe the role of Hadoop in handling big data, explain the basic workflow of processing big data using Hadoop.

• Comprehend the architecture and components of Hadoop Distributed File System (HDFS), understand the MapReduce programming model and its role in processing large datasets.

• Apply Python for streaming data processing in a Hadoop environment, Implement MapReduce jobs using Python for data analysis tasks, demonstrate the ability to count bigrams in a dataset using MapReduce, provide an overview of Pig for implementing machine learning techniques in big data analytics.

• Use Hive for querying and analyzing structured data in Hadoop, describe the role of HBase in storing and retrieving large volumes of data in Hadoop, utilize MySQL for integrating relational databases with Hadoop, explain the use of Flume for ingesting streaming data into Hadoop for analysis.

• Analyze real-world case studies to understand the application of Spark in big data analytics, demonstrate the ability to use Spark for processing and analyzing large datasets, apply machine learning algorithms in Spark for predictive analytics.

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	*							
CO2	*	*	*					
CO3	*	*	*					
CO4	*	*	*					
CO5	*	*	*	*		*		

Unit 1

Introduction to Big Data and Hadoop **12 hours**

Introduction To Big Data Platform, Challenges Of Conventional Systems, Web Data, Evolution of Analytic Scalability, Analytic Processes, and Tools. Data Product, Building Data Products at Scale with Hadoop, Leveraging Large Datasets, Hadoop for Data Products, The Data Science Pipeline and the Hadoop Ecosystem, Big Data Workflows.

Unit 2

Hadoop Distributed File System (HDFS) **12 hours**

Introduction to HDFS, Hadoop Architecture, A Hadoop Cluster - HDFS and YARN, Working with a Distributed File System, Basic File System Operations, File Permissions in HDFS, Other HDFS Interfaces. MapReduce: A Functional Programming Model, MapReduce implemented on a Cluster, Submitting a MapReduce Job to YARN.

Unit 3

A Framework for Python and Hadoop Streaming **12 hours**

Hadoop Streaming, Computing on CSV Data with Streaming, Executing Streaming Jobs, Framework for MapReduce with Python, Counting Bigrams, Other Frameworks, Advanced MapReduce, Combiners, Partitioners, and Job Chaining.
Pig, Pig Latin - Data Types, Relational Operators, User-Defined Functions. Machine learning - Collaborative Filtering, Classification, Clustering.

Unit 4

Structured Data Queries with Hive **12 hours**

The Hive Command-Line Interface (CLI), Hive Query Language (HQL), Data Analysis with Hive, HBase, NoSQL and Column-Oriented Databases, Real-Time Analytics with HBase, Importing Relational Data with Sqoop, Importing from MySQL to HDFS, Importing from MySQL to Hive, Importing from MySQL to HBase, Ingesting Streaming Data with Flume, Flume Data Flows, Ingesting Product Impression Data with Flume.

Unit 5

Big Data with Spark **12 hours**

Spark Introduction - Why Spark?, The Spark Stack, Understanding Spark Cluster Modes on YARN, RDDs (Resilient Distributed Datasets), RDD lineage, General RDD Operations, Building a Spark Application (Java, Python), The Spark Application Web UI, Configuring Spark Properties, Running Spark on Cluster, RDD Partitions, Spark SQL and DataFrames, Common Spark Use Cases(Data Cleaning (Movielens))

Textbooks:

Data Analytics with Hadoop by Benjamin Bengfort, Jenny Kim, 2016, O'Reilly Media, Inc.

Reference books:

- Taming The Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, Bill Franks, Thomas H. Davenport, 2012.
- Practical Data Science with Hadoop and Spark: Designing and Building Effective Analytics at Scale, 1st edition, Addison-Wesley Professional, 2017.
- E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012.
- Alan Gates, "Programming Pig", O'Reilley, 2011.

Course Code	CSAI8P126

Course Title	Cryptography Techniques Lab
Course Credit	02

Course Outcomes (COs)

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

- **CO1:** Implement classical cryptographic algorithms for secure data transformation.
- **CO2:** Design and implement symmetric key cryptographic algorithms and modes of operation.
- **CO3:** Implement public key cryptographic algorithms and key exchange mechanisms.
- **CO4:** Apply cryptographic hash functions, MACs, and digital signatures for authentication.
- **CO5:** Demonstrate secure communication protocols and real-world cryptographic applications.

Program Outcomes (POs)

- PO1: Apply knowledge of mathematics, science, and computing fundamentals.
- PO2: Analyze problems and identify appropriate computing solutions.
- PO3: Design, implement, and evaluate computing-based solutions.
- PO4: Use modern tools, techniques, and platforms for computing practices.
- PO5: Understand professional, ethical, legal, and security responsibilities.
- PO6: Communicate effectively and function efficiently in teams.
- PO7: Engage in lifelong learning and adapt to emerging technologie

List of Experiments / Programs

1. Program to implement Caesar Cipher (encryption and decryption).

2. Program to implement Monoalphabetic Substitution Cipher.
3. Program to implement Transposition Cipher.

4. Program to implement DES encryption and decryption.
5. Program to implement AES encryption and decryption.
6. Program to demonstrate Block Cipher Modes of Operation (ECB, CBC).
7. Program to perform file encryption using a symmetric key algorithm.
8. Program to implement RSA algorithm (key generation, encryption, decryption).
9. Program to implement Diffie–Hellman Key Exchange.
10. Program to implement ElGamal Cryptosystem.
11. Program to demonstrate Elliptic Curve Cryptography (ECC) basics.
12. Program to generate hash values using MD5 and SHA-256.
13. Program to implement Message Authentication Code (MAC).
14. Program to implement HMAC using SHA-256.
15. Program to demonstrate Digital Signature creation and verification.
16. Program to demonstrate Secure File Transfer using Cryptography.
17. Program to simulate Public Key Infrastructure (PKI) concepts.
18. Program to demonstrate the SSL/TLS handshake process (simulation).
19. Case study / mini project on Secure Communication System

Course Code: CSAI8P226	**Course Title: Machine Learning Lab**
Course Credits: 02	**Hours/Week: 04**
Total Contact Hours: 44	**Formative Assessment Marks: 50**
Exam Marks: 50	**Exam Duration: 2 Hrs**

Course Objectives:

1. To equip students with the knowledge about python libraries.
2. To equip students with the knowledge about machine learning algorithms.
2. To provide experience in applying machine learning algorithms to practical problems.

Course Outcomes:

1. Understand the library functions in python for data preprocessing
2. Use the python machine learning models.
3. Understand complexity of Machine Learning algorithms and their limitations
4. Apply appropriate algorithms for problem solving.
5. Capable of performing experiments in Machine Learning using real-world data

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	*							
CO2	*							
CO3	*							
CO4	*	*	*					
CO5	*	*	*	*		*		

	Program List	
	Python Libraries: Numpy, Pandas, MatplotLib and Scikit	

1.	Creating Arrays and Array Operations: a) Create a 1D array with elements from 0 to 9. b) Create a 2D array with shape (3, 4) filled with random numbers. c) Calculate the mean and standard deviation of a given array. d) Normalize the values of an array (subtract the mean and divide by the standard deviation).	
2	Generating Array, Reshaping and Stacking: a) Create a 1D array of 10 evenly spaced values between 0 and 1. b) Generate a 3x3 identity matrix. c) Reshape a 1D array into a 2D array with shape (2, 5). d) Stack two arrays vertically and horizontally.	
3	Indexing and Slicing & Matrix Operations: a) Extract the third column from a 2D array. b) Reverse the order of elements in a 1D array. c) Create two matrices (2x3 and 3x4) and perform matrix multiplication. d) Find the determinant of a 3x3 matrix.	
4	Statistical Operations, Broadcasting: a) Generate a random 2D array and calculate the mean along each axis. b) Find the minimum and maximum values in a given array. c) Create a 1D array and add a constant value to each element without using a loop. d) Multiply each row of a 2D array by a different constant.	
5	Develop a python program to create pandas data frame from list of data.	
6	Develop a python program to analyze the dataset using pandas and matplotlib library	
7	Develop a program to compute Mean, Median, Mode, Variance and Standard Deviation using Datasets.	
8	Plot histogram for the given dataset. 10-15 | 15-20 | 20-25 | 25-30 | 30-35 5 | 6 | 9 | 8 | 2	
9	Draw box-and-whisker plot for the data set {3, 7, 8, 5, 12, 14, 21, 13, 18}.	

<table>
<tr><td>10</td><td>Draw Line Plot and Bar chart for the following data.

<table><tr><td>Elapsed time (s)</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td></tr><tr><td>Speed(m/s)</td><td>0</td><td>3</td><td>7</td><td>12</td><td>20</td><td>30</td><td>45.6</td></tr></table></td><td></td></tr>
<tr><td>11</td><td>Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the
training data from a .CSV file</td><td></td></tr>
<tr><td>12</td><td>Develop a python program to implement Simple linear regression and plot
the graph</td><td></td></tr>
<tr><td>13</td><td>Develop a python program to implement single layer perceptron.</td><td></td></tr>
<tr><td>14</td><td>Implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering
few test data sets.</td><td></td></tr>
<tr><td>15</td><td>Demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge
to classify a new sample.</td><td></td></tr>
<tr><td>16</td><td>Implement k-Nearest Neighbor algorithm to classify the iris data set. Print
both correct and wrong predictions.</td><td></td></tr>
<tr><td>17</td><td>Apply k means algorithm to cluster a set of data stored in a .CSV file.</td><td></td></tr>
<tr><td>18</td><td>Build an Artificial Neural Network by implementing the Back propagation
algorithm and test the same using appropriate data sets.</td><td></td></tr>
<tr><td>19</td><td>Write a Python program to Read a CSV file,Handle missing values,

Apply StandardScaler,Split the data into training and testing sets</td><td></td></tr>
<tr><td>20</td><td>Implement a program to split a dataset into 70:30 ratio and compute: Accuracy,Precision,Recall,F1-score</td><td></td></tr>
<tr><td>21</td><td>Write a Python program to compute and display a confusion matrix for a given classifier.</td><td></td></tr>
<tr><td>22</td><td>Implement SVM for a linearly separable dataset and visualize the decision boundary.</td><td></td></tr>
<tr><td>23</td><td>Implement PCA to reduce the dataset to two dimensions and visualize the transformed data.</td><td></td></tr>
<tr><td>24</td><td>Apply Agglomerative Hierarchical Clustering and plot the dendrogram.</td><td></td></tr>
</table>