DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.M.S. COLLEGE OF ENGINEERING, BENGALURU

Autonomous Institute affiliated to VTU

Master of Technology

in

Computer Science and Engineering

Scheme and Syllabus 2024-2026

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.M.S. COLLEGE OF ENGINEERING, BENGALURU

Autonomous Institute affiliated to VTU

VISION OF INSTITUTE:

Promoting Prosperity of mankind by augmenting human resource capital through Quality Technical Education & Training

MISSION OF INSTITUTE:

Accomplish excellence in the field of Technical Education through Education, Research and Service needs of society.

VISION OF DEPARTMENT:

To be a model center for education and training in the frontier areas of Computer Science and Engineering.

MISSION OF DEPARTMENT:

M1:To Educate and Empower the students in the area of Computer Science by providing best practices of Teaching Learning Process for successful professional career.

M2: To enhance skills of the students to pursue higher studies and research.

M3: To foster the students to innovate and nurture them towards Entrepreneurship.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.M.S. COLLEGE OF ENGINEERING, BENGALURU

Autonomous Institute affiliated to VTU

Programme Educational Objectives (PEO)

PEO1: Able to excel professionally by adapting to the dynamic needs of Industry, Academia and Research in the field of Computer Science and Engineering.

PEO2: Able to practice and promote computing technologies for societal needs.

PEO3: Able to contribute to advancement of computer technology by means of research and lifelong learning.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.M.S. COLLEGE OF ENGINEERING, BENGALURU

Autonomous Institute affiliated to VTU

Program Outcomes (POs)

The Program Outcomes (POs) define the attributes to be developed through the program and are measured at the time of graduation. The POs are significantly addressed through the curriculum (Content, delivery and assessments) and further strengthened through co-curricular and extra-curricular activities.

The POs are listed below:

PO No.	Program Outcomes
PO 1	**An ability to independently carry out Research / Investigation and development work to solve practical problems**
PO 2	**An ability to write and present a substantial technical Report / Document**
PO 3	**An ability to demonstrate mastery in the area of Computer Science and Engineering, at a level higher than undergraduate program in Computer Science and Engineering.**

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

M.Tech. In Computer Science and Engineering

I SEMESTER

Sl. No	Course	Course Code	Course Title		Teaching Hours per Week		Examination				Credits
				Theory	Tutorial/ Skill Development Activities	Practical/Seminar	Duration in hours	CIE Marks	SEE Marks	Total Marks	
				L	T/SDA	P					
1	PCC	MCS101	Artificial Intelligence	03	00	00	03	50	50	100	3
2	PCC	MCS102	Data Science and Management	03	00	00	03	50	50	100	3
3	PCC	MCS103	Data Structures & Algorithms for Problem Solving	03	00	00	03	50	50	100	3
4	PCC	MCS104A	Multicore Architecture & Programming	03	00	00	03	50	50	100	3
5	IPCC	MCS105B	Forensic Security	02	00	02	03	50	50	100	4
6	PCCL	MCSL106	Algorithms & AI Laboratory	00	00	02	03	50	50	100	2
7	NCMC	MRMI107	Research Methodology and IPR (**Online**)		Online courses (online.vtu.ac.in)						0
								300	**300**	**600**	**18**

Note: **BSC**-Basic Science Courses, **PCC**: Professional Core Courses. **IPCC**-Integrated Professional Core Courses, **PCC(PB):** Professional Core Courses (Project Based), **PCCL**-Professional Core Course lab **, NCMC**- Non Credit Mandatory Course, ,**L**-Lecture**, P**-Practical**, T/SDA-**Tutorial **/** Skill Development Activities (Hours are for Interaction between faculty and students)

MRMI107- Research Methodology and IPR (**Online**) for the students who have **not studied** this course in the Undergraduate level. This course is not counted for vertical progression, students have to qualify for the award of the master's degree.

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

M.Tech. In Computer Science and Engineering

II SEMESTER

Sl. No	Course	Course Code	Course Title	Teaching Hours / Week			Examination				Credits
				Theory	Tutorial/ Skill Development Activities	Practical/Seminar	Duration in hours	CIE Marks	SEE Marks	Total Marks	
				L	T/SDA	P					
1	PCC	MSCS201	DevOps and MLOps	02	02	00	04	50	50	100	3
2	PCC	MSCS202	Deep Learning and Generative AI	03	00	00	03	50	50	100	3
3	IPCC	MSCS203	Cloud Computing and Infrastructure Management	03	00	01	05	50	50	100	4
4	PEC1	MSCS214x	Professional Elective-1	03	00	00	03	50	50	100	3
5	PEC2	MSCS215x	Professional Elective-2	03	00	00	03	50	50	100	3
6	PBLC	MSCS286	Mini Project/Technology-Based Societal Project	00	00	03	06	50	50	100	3
7	PCCL	MSCSL207	Deep Learning and Generative AI Lab	00	00	02	04	50	50	100	2
8	AEC/SEC	MSCS258x	Ability/Skill Enhancement Course (Offline/Online)	00	00	02	04	50	50	100	1
TOTAL								**450**	**350**	**800**	**22**

Note: **PCC:** Professional Core Courses. **IPCC**-Integrated Professional Core Courses, **PCC(PB):** Professional Core Courses (Project Based), **PCCL**-Professional Core Course lab, **PEC**- Professional Elective Courses, **MDC**- Multi-Disciplinary Courses, **L-Lecture, P-Practical, T/SDA-Tutorial / Skill Development Activities** (Hours are for Interaction between faculty and students), **PBLC**: Project Based Learning Course.

Ability / Skill Enhancement Courses

Course Code	Course Title
MSCS258A	Advanced Web Technology
MSCS258B	Augmented and Virtual Reality
MSCS258C	Competitive Coding

Professional Elective-1		Professional Elective-2	
MSCS214A	Advanced Cryptography	MSCS215A	Big Data Technologies
MSCS214B	Pattern Classification	MSCS215B	System and Network Security
MSCS214C	Edge and Fog Computing	MSCS215C	Applied Natural Language Processing
MSCS214D	Quantum Computing	MSCS215D	High-Performance Computer Architectures

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

I SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MCS101
Course Name	**Artificial Intelligence**	**Total No. of Lecture Hours**	**40**
CIE Marks	**50**	**SEE Marks**	**50**

Unit 1:

Introduction: What is AI? The Foundations of Artificial Intelligence. 8 hrs.

Problem-Solving: Problem-Solving Agents, Search algorithms, Uninformed search strategies, Informed (Heuristic) search strategies.

Unit 2:

Knowledge, reasoning and planning: Knowledge-based agents, Logic, Propositional logic: A very simple logic, Propositional theorem proving, Syntax and semantics of First-order logic, Using First- order logic, Propositional vs First-order inference, Unification and first-order inference. 8 hrs.

Unit 3:

Uncertain knowledge and reasoning: Acting under uncertainty, Basic Probability Notation, Inference using full joint distribution, Independence, Baye’s rule and its use, Naïve Bayes models, Representing knowledge in an uncertain domain, The semantics of Bayesian networks. 8 hrs.

Unit 4:

Machine learning: Forms of learning, Supervised learning, Learning decision trees, Model selection and optimization, Linear regression and classification, Nonparametric models. 8 hrs.

Unit 5:

Natural Language Processing: Language models, Grammar, Parsing, Augmented grammars, Complications of real natural language, Natural language tasks. 8 hrs.

The Ethics of AI, The Future of AI.

Text Book:

1 Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig, 4th Edition (2021).

References:

1 "Pattern Recognition and Machine Learning" by Christopher M. Bishop Edition: 4th Edition (2020)

2 "Artificial Intelligence: Foundations of Computational Agents" by David L. Poole and Alan K. Mackworth Edition: 3rd Edition (2021).

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Understand the foundational concepts of artificial intelligence, types, ethics and key problem-solving techniques.			**3**
CO2:	Apply knowledge representation and reasoning techniques to solve complex problems in AI systems.			**3**
CO3:	Implement machine learning algorithms and evaluate their performance in real-world applications.	**2**		
CO4:	Explore the principles and applications of natural language processing to enhance human-computer interaction.			**3**

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

I SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MCS102
Course Name	**Data Science and Management**	**Total No. of Lecture Hours**	**40**
CIE Marks	**50**	**SEE Marks**	**50**

Unit 1:
Introduction to Data Science: The Data Science Process: Roles in a data science project, Stages of a data science project, Setting expectations. Starting with Rand data: Starting with R, Working with data from files, Working with relational databases. Exploring data: Using Summary Statistics to spot problems, Spotting problems with graphics and visualization. 8 hrs.

Unit-2:
Managing data: Cleaning data, Data transformations, Sampling for modeling and validation. Choosing and evaluating models: Mapping problems to machine learning tasks, evaluating models, Local interpretable model-agnostic explanations (LIME) for explaining model predictions. 8 hrs.

Unit-3:
Modeling Methods: Supervised methods Linear Regression, Logistic Regression. Unsupervised methods: Cluster Analysis, Association rules. Exploring Advanced Methods: Bagging and random forest, Generalized additive models, Kernel methods to increase data separation. 8 hrs.

Unit-4:
Documentation, Deployment and Presentation: Predicting buzz, Using R markdown to produce milestone documentation, using comments and version control for running documentation, Deploying models, Producing Effective Presentations. 8 hrs.

Unit-5:
Text Mining and Sentiment analysis: The tidy text format, Sentiment analysis with Tidy data: The sentiments datasets, Comparing the three sentiment dictionaries, Most common positive and negative words, Word clouds, Looking at units beyond just words, Analyzing word and document frequency. 8 hrs.

Text Book:

1. Zumel, N., Mount, J., & Porzak, J., "Practical data science with R", 2nd edition. Shelter Island, NY: Manning, 2019.
2. Julia Silge and David Robinson. "Text Mining with R: A Tidy Approach", 1st. edition. O'Reilly Media, Inc., 2017

References:

1. Eric Pimpler, Data Visualization and Exploration with R, GeoSpatial Training service, 2017.

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO 1**	**PO 2**	**PO 3**
CO1:	Apply the concepts of modeling methods			**3**
CO2:	Analyze and formulate relationships of data			**3**
CO3:	Implement various models on different datasets using R	**3**		

I SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MCS103
Course Name	Data Structures & Algorithms for Problem Solving	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1: 8 hrs.

Search Trees: Two Models of Search Trees. General Properties and Transformations. Height of a Search Tree. Basic Find, Insert, and Delete. Returning from Leaf to Root. Dealing with Non unique Keys. Queries for the Keys in an Interval. Building Optimal Search Trees. Converting Trees into Lists. Removing a Tree.

Unit 2: 8 hrs.

Balanced Search Trees: Height-Balanced Trees. Weight-Balanced Trees. (a, b)- and B-Trees. Red-Black Trees and Trees of Almost Optimal Height. Top-Down Rebalancing for Red-Black Trees.

Unit 3: 8 hrs.

Heaps: Balanced Search Trees as Heaps. Array-Based Heaps. Heap-Ordered Trees and Half Ordered Trees. Leftist Heaps. Skew Heaps. Binomial Heaps. Changing Keys in Heaps. Fibonacci Heaps. Heaps of Optimal Complexity.

Unit 4: 8 hrs.

Graph Algorithms: Bellman - Ford Algorithm; Single source shortest paths in a DAG; Johnson's Algorithm for sparse graphs. Flow networks and Ford-Fulkerson method. Maximum bipartite matching.

Unit 5: 8 hrs.

String-Matching Algorithms: Naïve string Matching. Rabin - Karp Algorithm. String matching with Finite Automata. Knuth-Morris-Pratt Algorithm. Boyer – Moore Algorithms.

Data Structures for Strings: Tries and Compressed Tries, Suffix Trees.

Text Books:

1. Advanced Data Structures, Peter Brass, Cambridge University Press, 2008.
2. Kenneth A. Berman. Algorithms. Cengage Learning. 2002.
3. T. H Cormen, C E Leiserson, R L Rivest and C Stein. Introduction to Algorithms. PHI, 3rd Edition, 2010

Reference Books:

1. Data Structures and Algorithm Analysis in C++, Mark Allen Weiss, 4 th Edition, 2014, Pearson.
2. Data structures with Java, Ford and Topp, Pearson Education.

3. Ellis Horowitz, Sartaj Sahni, S.Rajasekharan. Fundamentals of Computer Algorithms. Universities press. 2nd Edition, 2007
4. Data structures and Algorithms in Java, M.T.Goodrich, R.Tomassia, 3rd edition, Wiley India Edition.

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1	Analyze fundamental data structures and algorithms to solve complex computational problems effectively			**3**
CO2	Apply various searching algorithm to optimize algorithm performance.			**3**
CO3	Design and analyze advanced tree and graph algorithms, including balanced search trees and graph traversal methods, to address real-world applications	**3**		

I SEMESTER

Credits	3 [L-T-P: 3-0-0]	Course Code	MCS104A
Course Name	Multicore Architecture & Programming	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1:

Introduction to Multi-core Architecture: Motivation for Concurrency in software, Parallel Computing Platforms, Parallel Computing in Microprocessors, Differentiating Multi-core Architectures from Hyper- Threading Technology, Multi-threading on Single-Core versus Multi-Core Platforms Understanding Performance, Amdahl's Law, Growing Returns: Gustafson's Law. System Overview of Threading: Defining Threads, System View of Threads, Threading above the Operating System, Threads inside the OS, Threads inside the Hardware, What Happens When a Thread Is Created, Application Programming Models and Threading, Virtual Environment: VMs and Platforms, Runtime Virtualization, System Virtualization. 8 hrs.

Unit 2:

Fundamental Concepts of Parallel Programming: Designing for Threads, Task Decomposition, Data Decomposition, Data Flow Decomposition, Implications of Different Decompositions, Challenges You'll Face, Parallel Programming Patterns, A Motivating Problem: Error Diffusion, Analysis of the Error Diffusion Algorithm, An Alternate Approach: Parallel Error Diffusion, Other Alternatives. Threading and Parallel Programming Constructs: Synchronization, Critical Sections, Deadlock, Synchronization Primitives, Semaphores, Locks, Condition Variables, Messages, Flow Control- based Concepts, Fence, Barrier, Implementation-dependent Threading Features 8 hrs.

Unit 3:

Threading APIs: Threading APls for Microsoft Windows, Win32/MFC Thread APls, Threading APls for Microsoft. NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signalling, Compilation and Linking. 8 hrs.

Unit 4:

Open MP: A Portable Solution for Threading: Challenges in Threading a Loop, Loop-carried Dependence, Data-race Conditions, Managing Shared and Private Data, Loop Scheduling and Portioning, Effective Use of Reductions, Minimizing Threading Overhead, Work-sharing Sections, Performance-oriented Programming, Using Barrier and No wait, Interleaving Single-thread and Multithread Execution, Data Copy-in and Copy-out, Protecting Updates of Shared Variables, Intel Task queuing Extension to Open MP, Open MP Library Functions, Open MP Environment Variables, Compilation, Debugging, performance. 8 hrs.

Unit 5:

Solutions to Common Parallel Programming Problems: Too Many Threads, Data Races, Deadlocks, and Live Locks, Deadlock, Heavily Contended Locks, Priority Inversion, Solutions for Heavily Contended Locks, Non-blocking Algorithms, ABA Problem, Cache Line Ping-ponging, Memory Reclamation Problem, Recommendations, Thread-safe Functions and Libraries, Memory Issues, Bandwidth, Working in the Cache, Memory Contention, Cache-related Issues, False Sharing, Memory Consistency, Current IA-32 Architecture, Itanium Architecture, High-level Languages, Avoiding Pipeline Stalls on IA-32,Data Organization for High Performance. 8 hrs.

Text Book:

1. Shameem Akhter and Jason Roberts: Multicore Programming, Increased Performance through Software Multi-threading, Intel Press, 2006.

Reference:

1. Calvin Lin, Lawrence Snyder: Principles of Parallel Programming, Pearson Education, 2009.
2. Michael J. Quinn: Parallel Programming in C with MPI and OpenMP, Tata McGraw Hill, 2004.

Course Outcomes (Cos):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Demonstrate scholarship of knowledge to analyze various types of Multithreading/Multicore architectures/techniques & Estimate their Performance			**3**
CO2:	Implement Parallel Solutions using multi-threading with the help of appropriate tools and present a technical report/document.	**2**		
CO3:	Apply threading API's to solve practical problems to exploit parallelism in Multicore Environment			**3**

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

I SEMESTER

Credits	4[L-T-P: 2-0-2]	Course Code	MCS105B
Course Name	**Forensic Security**	**Total No. of Lecture Hours**	**40**
CIE Marks	**50**	**SEE Marks**	**50**

Unit 1:

Introduction to Cybercrime: Introduction, Cybercrime, Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals?, Classifications of Cybercrimes, Cybercrime: The Legal Perspectives, Cybercrimes: An Indian Perspective, Cybercrime and the Indian ITA 2000 , A Global Perspective on Cybercrimes. 8 hrs.

Cyber offenses: Introduction, How Criminals Plan the Attacks, Social Engineering, Cyberstalking, Cybercafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector.

Unit-2:

Cybercrime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era, Laptops. 8 hrs.

Unit-3:

Tools and Methods Used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks. 8 hrs.

Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft).

Unit-4:

Understanding Computer Forensics: Introduction, Historical Background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber forensics and Digital Evidence, Forensics Analysis of E-Mail, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation. 8 hrs.

8 hrs.

Unit-5:

Cybercrime and Cyberterrorism: Social, Political, Ethical and Psychological Dimensions, Intellectual Property in the Cyberspace, The Ethical Dimension of Cybercrimes, The Psychology, Mindset and Skills of Hackers and Other Cybercriminals, Sociology of Cybercriminals.

Real-Life Examples: Indian Banks Lose Millions of Rupees, Justice vs. Justice- Software Developer Arrested for Launching Website Attacks.

Cybercrime: Online Scams: - Purchasing Goods and Services Scams, Malware Scams.

Text Books:

1 Sunit Belapure and Nina Godbole, “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, WILEY INDIA, 2011.

References:

1 Marjie T. Britz, Computer Forensics and Cyber Crime, Pearson, 2012.

2 David Cowen, Computer Forensics: A Beginners Guide, Mc Graw Hill Education, 2013.

3 Bill Nelson, Amelia Phillips, Christopher Steuart, Guide to Computer Forensics and Investigations, 4^{th} Edition, 2014.

Course Outcomes (Cos):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Exhibit advanced expertise and understanding in cybersecurity and forensics by applying comprehensive knowledge and skills acquired.			**3**
CO2:	Analyze cybersecurity threats, vulnerabilities, and forensic techniques.			**3**
CO3:	Execute detailed investigative procedures and develop innovative solutions to address complex challenges in cybersecurity and forensics.	**2**		
CO4:	Write and present substantial technical reports articulating complex cybersecurity and forensics concepts, strategies, and findings effectively.		**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

I SEMESTER

Credits	2[L-T-P: 0-0-2]	Course Code	MCSL106
Course Name	Algorithms & AI Lab	Total No. of Lecture Hours	-
CIE Marks	50	SEE Marks	50

List of Experiments:

1. Implement Naive Bayes models and Bayesian networks. (Demonstrate the diagnosis of heart patients using standard heart disease data set etc)

2. Implement a simple linear regression algorithm to predict a continuous target variable based on a given dataset.

3. Develop a program to implement a Support Vector Machine for binary classification. Use a sample dataset and visualize the decision boundary.

4. Write a program to demonstrate the ID3 decision tree algorithm using an appropriate dataset for classification.

5. Implement a KNN algorithm for regression tasks instead of classification. Use a small dataset, and predict continuous values based on the average of the nearest neighbors.

6. Implement the k-Nearest Neighbor algorithm to classify the Iris dataset, printing both correct and incorrect predictions.

7. Develop a program to implement the non-parametric Locally Weighted Regression algorithm, fitting data points and visualizing results.

8. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test it with suitable datasets.

9. Implement a Q-learning algorithm to navigate a simple grid environment, defining the reward structure and analyzing agent performance.

10. Write a python program
a. to perform tokenization by word and sentence using nltk.
b. to eliminate stop words using nltk.
c. to perform stemming using nltk.
d. to perform Parts of Speech tagging using nltk.

Note: Instructions to Students to be followed in each lab:

1. Each student should bring the observation book for each lab and write the programs and output completed in the previous week and get it evaluated by the faculty in charge.
2. In the observation book, students should
   a) Handwrite the Program/scenarios with topology and procedure
   b) Paste the printout of the Output or Handwrite the Output (Output should be written for all the cases).
3. Hard copy (PDF file) of all the experiments and programs along with the output needs to be submitted before the lab test.
4. Each Student should practice the extra exercise given in each lab.

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

I SEMESTER

Credits	PP	Course Code	MRMI107
Course Name	**Research Methodology and IPR**	**Total No. of Lecture Hours**	-
CIE Marks	-	**SEE Marks**	-

Unit 1:
Research Methodology: Introduction, Meaning of Research, Objectives of Research, Motivation in Research, Types of Research, Research Approaches, Significance of Research, Research Methods versus Methodology, Research and Scientific Method, Importance of Knowing How Research is Done, Research Process, Criteria of Good Research, and Problems Encountered by Researchers in India. Defining the Research Problem: Research Problem, Selecting the Problem, Necessity of Defining the Problem, Technique Involved in Defining a Problem, An Illustration

Unit-2
Reviewing the literature: Place of the literature review in research, Bringing clarity and focus to your research problem, Improving research methodology, Broadening knowledge base in research area, Enabling contextual findings, How to review the literature, searching the existing literature, reviewing the selected literature, Developing a theoretical framework, Developing a conceptual framework, Writing about the literature reviewed. Research Design: Meaning of Research Design, Need for Research Design, Features of a Good Design, Important Concepts Relating to Research Design, Different Research Designs, Basic Principles of Experimental Designs, Important Experimental Designs.

Unit-3
Design of Sampling: Introduction, Sample Design, Sampling and Non-sampling Errors, Sample Survey versus Census Survey, Types of Sampling Designs. Measurement and Scaling: Qualitative and Quantitative Data, Classifications of Measurement Scales, Goodness of Measurement Scales, Sources of Error in Measurement Tools, Scaling, Scale Classification Bases, Scaling Technics, Multidimensional Scaling, Deciding the Scale. Data Collection: Experimental and Surveys, Collection of Primary Data, Collection of Secondary Data, Selection of Appropriate Method for Data Collection, Case Study Method.

Unit-4
Testing of Hypotheses: Hypothesis, Basic Concepts Concerning Testing of Hypotheses, Testing of Hypothesis, Test Statistics and Critical Region, Critical Value and Decision Rule, Procedure for Hypothesis Testing, Hypothesis Testing for Mean, Proportion, Variance, for Difference of Two Mean, for Difference of Two Proportions, for Difference of Two Variances, P-Value approach, Power of Test, Limitations of the Tests of Hypothesis. Chi-square @#@#@ Test: Test of Difference of more than Two Proportions, Test of Independence of Attributes, Test of Goodness of Fit, Cautions in Using Chi Square Tests.

Unit-5
Interpretation and Report Writing: Meaning of Interpretation, Technique of Interpretation, Precaution in Interpretation, Significance of Report Writing, Different Steps in Writing Report, Layout of the Research Report, Types of Reports, Oral Presentation, Mechanics of Writing a Research Report, Precautions for Writing Research Reports. Intellectual Property: The Concept, Intellectual Property System in India, Development of TRIPS Complied Regime in India, Patents Act, 1970, Trade Mark Act, 1999,The Designs Act, 2000, The Geographical Indications of Goods (Registration and Protection) Act1999, Copyright Act,1957,The Protection of Plant Varieties and Farmers' Rights Act, 2001,The Semi-Conductor Integrated Circuits Layout Design Act, 2000, Trade Secrets,

Utility Models, IPR and Biodiversity, The Convention on Biological Diversity (CBD) 1992, Competing Rationales for Protection of IPRs, Leading International Instruments Concerning IPR, World Intellectual Property Organisation (WIPO),WIPO and WTO, Paris Convention for the Protection of Industrial Property, National Treatment, Right of Priority, Common Rules, Patents, Marks, Industrial Designs, Trade Names, Indications of Source, Unfair Competition, Patent Cooperation Treaty (PCT), Advantages of PCT Filing.

Text Books:

1 Research Methodology: Methods and Techniques, C.R. Kothari, Gaurav Garg, New Age International, 4th Edition, 2018.
2 Research Methodology a step-by-step guide for beginners.RanjitKumar, SAGE Publications, 3rd Edition, 2011.

References:

1 Research Methods: the concise knowledge base, Trochim, Atomic Dog Publishing, 2005.

2 Conducting Research Literature Reviews: From the Internet to Paper, Fink A, Sage Publications, 2009.

Web links:

1. https://www.youtube.com/watch?v=A7oioOJ4g0Y&list=PLVf5enqoJ-yVQ2RXUl6mCfLPf3J_JUfoc

Course Outcomes (Cos):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Identify and Conduct research independently in suitable research field.			**3**
CO2:	Choose research designs, sampling designs, measurement and scaling techniques and also different methods data collection.			**3**
CO3:	Explore the Precautions in interpreting the data and drawing inferences.			**3**

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 2-2-0]	Course Code	MSCS201
Course Name	**DevOps and MLOps**	**Total No. of Lecture Hours**	**40**
CIE Marks	**50**	**SEE Marks**	**50**

Unit 1: **Introduction to DevOps and MLOps:** Understanding DevOps and MLOps, Importance of DevOps in software development, Overview of MLOps and its role in machine learning, Difference between DevOps, MLOps, and DataOps, Key principles and best practices, CI/CD in traditional software vs. ML pipelines, Model development to deployment lifecycle	8 hrs.
Unit 2: **Data Management and Versioning in MLOps:** Managing data and model versioning (Git, DVC, MLflow), Handling large datasets for ML, Data versioning tools: DVC, Pachyderm, Integrating data quality checks into workflows, Experiment tracking and model reproducibility (MLflow, TensorBoard	8 hrs.
Unit 3: **CI/CD and Automation in MLOps:** Automation in DevOps & MLOps, Automating ML model training processes, Scalable training techniques (distributed training with Horovod, Ray), CI/CD for Machine Learning and DevOps, GitHub Actions, Jenkins, GitLab CI/CD for automated pipelines, Continuous training (CT) for model retraining, Evaluation metrics & validation strategies	8 hrs.
Unit 4: **Model Deployment and Production Monitoring:** DevOps & MLOps Deployment Strategies: Blue-Green, Canary, A/B Testing, Online, batch, and streaming deployments**,** Model Serving and Scaling: TensorFlow Serving, TorchServe, FastAPI**,** Model optimization techniques (Quantization, ONNX)**,** Monitoring & Model Performance Management: Prometheus & Grafana**,** Detecting model drift, data drift, and concept drift**,** Strategies for retraining and updating ML models	8 hrs.
Unit 5: **Infrastructure, Orchestration & Security:** Containerization & Orchestration: Docker, Kubernetes, Kubeflow, KServe, Managing compute resources with Kubeflow, Feature Store and Workflow Orchestration: Feast, Tecton, Apache Airflow, Kubeflow Pipelines, Security & Compliance in DevOps & MLOps, DevSecOps: Security in CI/CD pipelines, Model security risks & GDPR compliance	8 hrs.

Text Book:

1 “Practical MLOps: Operationalizing Machine Learning Models”, Noah Gift, Alfredo Deza, O'Reilly Media, 1st Edition, 2021.
2 “Machine Learning with Kubernetes: Deploy, Manage, and Scale Models with Kubernetes”, Oleg Iegorov, Faisal Masood, Krishnaram Kenthapadi, O'Reilly Media, 1st Edition, 2021.
3 “Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications”, Chip Huyen, O'Reilly Media, 1st Edition, 2022.
4 The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations" Gene Kim, Jez Humble, Patrick Debois, and John Willis, IT Revolution Press, 2nd Edition (2021).

References:

1 “Introducing MLOps”, Mark Treveil & The Dataiku Team, O'Reilly Media, 1st Edition, 2020

2 “Building Machine Learning Pipelines”, Hannes Hapke, Catherine Nelson, O'Reilly Media, 1st Edition, 2020

3 “The DevOps Toolkit: Automating the Continuous Deployment Pipeline with Containers and Kubernetes”, Viktor Farcic, Packt Publishing, 1st Edition, 2019

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply DevOps and MLOps principles, CI/CD pipelines and infrastructure orchestration techniques to software and machine learning workflows.			**3**
CO2:	Analyze the impact of versioning, orchestration, and security strategies on the reliability and performance of ML systems			**3**
CO3:	Implement end-to-end DevOps and MLOps workflows	**3**		**3**

Tutorial Plan

The students should review the relevant theoretical concepts before the tutorial class. The followings are the prerequisites for conducting hands on classes.

- Ensure that Git, Jenkins, MLflow, and necessary dependencies are installed before the session.
- Have a basic understanding of Python, SQL, and cloud platforms.

The students should maintain softcopy of code and document created in each tutorial session

S.No	Name of the experiment
1	Version Control using GIT Creating repositories in GIT, Exercises to demonstrate the use of GIT operations and commands (Push, pull, clone etc.,) Creating branches and merging branches using GIT
2	Installation of Jenkins and Configuration of Jenkins to work with different version control, build and testing tools
3	Create jobs and projects in Jenkins
4	Demonstration of continuous integration with Jenkins through source code polling and build triggers
5	Demonstrate continuous inspection with Jenkins using sonarqube to ensure code quality

6	Demonstration of continuous deployment/delivery to staging/production environment with Jenkins.
7	Construct an end-to-end Machine Learning Pipeline using MLflow (MATS?) ● Stages a) Problem understanding (aka business understanding) b) Data collection c) Data annotation d) Data wrangling e) Model development, training and evaluation f) Model Validation g) Local Model deployment h) Prediction ● Tech-Stack a) RDBMS / Real time Source b) SQL/Python c) Dbt d) Feast e) DVC f) Python/Scikit-Learn g) MLFlow h) GitHub i) REST
8	Manage Machine Learning Model Metadata using MLFlow / Neptune (Continuous Integration) ● Components a) Projects b) Experiments c) Metadata d) Model tracking / logging e) Model Registry ● Tech-Stack a) Python b) Jupyter Notebooks c) MLFlow / Neptune d) GIT?
9	Deploy and serve the ML model as Microservices (Continuous Delivery) ● Stages a) Triggers for deployment b) Local Deployment using containers c) Cloud deployment using AWS services (Sagemaker + S3 etc.) d) Offline (batch) serving e) Online serving ● Tech-stack a) Python b) Containers c) AWS d) MLflow Model Registry

	e) API
10	Monitor the Performance of deployed predictive model ● Stages a) Monitoring Data and feature drift b) Monitoring target drift c) Monitoring model performance ● Tech-Stack a) MLflow b) Model Server c) Evidently
11	Manage MLOps lifecycle using Cloud services ● Stages ● Tech-Stack a) Azure Machine Learning b) Azure DevOps

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS202
Course Name	Deep Learning and Generative AI	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1:

Introduction: Neural Networks: Building Intelligent Systems, Limitations of Traditional Computer Programs, Mechanics of Machine Learning, Neuron, Expressing Linear Perceptrons as Neurons, Feed-Forward Neural Networks, Linear Neurons and Limitations, Sigmoid, Tanh and ReLU Neurons, Softmax Output Layers, Training Feed-Forward Neural Networks: Gradient Descent, Delta Rule and Learning Rates, Gradient Descent with Sigmoidal Neurons, Backpropagation Algorithm, Stochastic and Minibatch Gradient Descent, Overfitting, Preventing Overfitting in Deep Neural Networks. 8 hrs.

Unit-2:

Convolutional Neural: Introduction, Filters and Feature Maps, Max Pooling, Popular CNN Architectures: LeNet, AlexNet, VGG, ResNet, Accelerating Training with Batch Normalization, Group Normalization for Memory Constrained Learning Tasks, Residual Learning and Skip Connections for very Deep Networks, Transfer Learning, CNN Applications 8 hrs.

Unit-3:

Recurrent Neural Networks: Introduction, Backpropagation in RNNs, Regularization Techniques: Dropout, L1/L2 Regularization, LSTMs and GRUs, Video Understanding using RNNs, Attention in Vision Models: An Introduction, Soft and Hard Attention: Image Captioning, Self-Attention and Transformers, Vision Transformers and Applications 8 hrs.

Unit-4:

Introduction To Gen AI: Historical Overview of Generative modeling, Difference between Gen AI and Discriminative Modeling, Types of Generative models – GANs, VAEs, autoregressive models and Vector quantized Diffusion models, Challenges of Generative Modeling, Future of Gen AI, Ethical Aspects of Gen AI. 8 hrs.

Unit-5:

Generative Models for Text: Language Models Basics, Building blocks of Language models, Transformer Architecture, Encoder and Decoder, Attention mechanisms, Generation of Text: Models like BERT and GPT models, Exploring ChatGPT, Prompt Engineering: Designing Prompts, Issues of LLM like hallucination. 8 hrs.

Text Book:

1. Nikhil Buduma, Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithm, O’Reilly, 2017
2. Josh Patterson & Adam Gibson, Deep Learning: A Practitioner's Approach, O’Reilly, First Edition, 2017.
3. Rajalingappaa Shanmugamani, Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras, Packt Publishing, 2018
4. David Foster, Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play, 2019
5. Altaf Rehmani, Generative AI for everyone: Understanding the essentials and applications of this breakthrough technology. BlueRose One, 2024
6. Research Papers on Relevant Topics and Internet Resources

References:

1. Goodfellow I, Bengio Y, Courville A, Deep learning, MIT press, 2016
2. Giancarlo Zaccone, Md. Rezaul Karim, Ahmed Menshawy, Deep Learning with TensorFlow: Explore neural networks with Python, Packt Publisher, 2017
3. Umberto Michelucci , Applied Deep Learning. A Case-based Approach to Understanding Deep Neural Networks, Springer, 2018
4. Charu C. Aggarwal, Neural Networks and Deep Learning: A Textbook, Springer, 2018
5. Francois Chollet, Deep Learning with Python, Manning Publications Co, First Edition, 2018
6. Josh Kalin, Generative Adversarial Networks Cookbook: Over 100 recipes to build generative models using Python, TensorFlow, and Keras, Packt Publishing; 1st edition, December 2018
7. Andrew Glassner, Deep learning a visual approach, No starch press, 2021
8. Adrian Rosebrock. Deep Learning for Computer Vision with Python, PyImageSearch, 2021
9. Numa Dhamani and Maggie Engler, Introduction to Generative AI, Manning, 2024
10. Bernad Marr, Generative AI in Practice, Wiley, 2024
11. Denis Rothman, Transformers for Natural Language Processing and Computer Vision, Third Edition, Packt publishing, 2024
12. Jay Alammar, Maarten Grootendorst Hands-On Large Language Models, O'Reilly Media, 2024

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Recognize the mathematics behind functioning of ANN, CNN, RNN and Generative Models.	**3**		
CO2:	Build and train ANNs, CNNs, RNN for computer vision tasks, such as image classification.	**3**		
CO3:	Demonstrate the application of transfer learning and fine-tuning techniques for pre-trained CNNs.			**2**
CO4:	Demonstrate the application of Generative AI models for Generating Texts			**2**

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 2-0-2]	Course Code	MSCS203
Course Name	**Cloud Computing and Infrastructure Management**	**Total No. of Lecture Hours**	**40**
CIE Marks	**50**	**SEE Marks**	**50**

Unit 1: **Fundamental Concepts of Cloud Computing:** Goals and Benefits, Risks and Challenges, Cloud Computing Service and Deployment Models: Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud, Multi-Cloud.	8 hrs.
Unit-2: **Cloud-Enabling Technology**: Cloud Data Center Technology, Modern Virtualization, Multitenant Technology, Service Technology and Service APIs, Fundamental of Containerization, Containers, Container Images, Multi-Container Types. **Cloud Infrastructure Mechanisms**: Logical Network Perimeter, Virtual Server, Hypervisor, Cloud Storage Device, Cloud Usage Monitor, Resource Replication, Ready-Made Environment.	8 hrs.
Unit-3: **Specialized Cloud Mechanisms**: Automated Scaling Listener, Load Balancer, SLA Monitor, Pay Per-Use Monitor, Audit Monitor, Failover System, Resource Cluster, Multi-Device Broker, State Management Database **Cloud Management Mechanisms**: Remote Administration System, Resource Management System, SLA Management System, Billing Management System.	8 hrs.
Unit-4: **Cloud Computing Architectures**: Workload Distribution Architecture, Elastic Resource Capacity Architecture, Multi Cloud Architecture, Hypervisor Clustering Architecture, Cloud Balancing Architecture **Specialized Cloud Architectures**: Edge Computing Architecture, Fog Computing Architecture, Metacloud Architecture, Federated Cloud Application Architecture.	8 hrs.
Unit-5: **Cloud Computing Security**: Threat Agents, Common Threats, **Cloud Security and Cybersecurity Access-Oriented Mechanisms**: Cloud-Based Security Groups, Hardened Virtual Server Image, Identity and Access Management (IAM) System, **Cloud Security and Cybersecurity Data-Oriented Mechanisms**: Data Loss Prevention (DLP) System, Trusted Platform Module (TPM). **Cloud Delivery Model Considerations**: Case Study on Cloud Provider and Consumer Perspective.	8 hrs.

Text Book:

1. Thomas Erl, Eric Barceló Monroy, “Cloud Computing: Concepts, Technology, Security, and Architecture”, 2^{nd} Edition, 2023, Pearson, ISBN: 9780138052287.

References:

1. "Rajkumar Buyya, Christian Vecchiola, and S. Thamarai Selvi, “Cloud Computing: Principles and Practice", 2020.
2. Comer, D, “The Cloud Computing Book: The Future of Computing Explained”, 1st edition, Chapman and Hall/CRC, 2021. https://doi.org/10.1201/9781003147503.

3. Sean Howard, "Edge Computing with Amazon Web Services: A practical guide to architecting secure edge cloud infrastructure with AWS", 1st Edition, ISBN: 9781835081082, Packt Publishers, 2024

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply the knowledge of cloud computing concepts and cloud management mechanisms for different applications.			**3**
CO2:	Analyze cloud enabled technologies and potential risks for cloud deployments of various applications.			**3**
CO3:	Conduct experiments to demonstrate the different cloud computing technologies.	**3**		

Experiment list:

1. Demonstration of SaaS application (Demonstrate CRM application suitable platform)
2. Demonstration of PaaS application (Create a Trailblazer app on sales force trailhead platform)
3.
    a. Create a virtual machine using VMWare Workstation PRO12 (VMware PlayStation)
    b. Install Python on created a virtual machine and run a simple program.
4. Installation of CloudSim Simulator
5. Demonstration and program implementation on CloudSim Simulator
6. Demonstration of Dockers and Kubernetes.
7. Introduction to Cloud Course on AWS Educate
8. Getting Started with Security on AWS Educate
9. Getting Started with Compute on AWS Educate
10. Complete Getting Started with Serverless on AWS Educate
11. Create an application for data analysis (Use any front-end tool and NoSQL Database as backend), Dockerize the same and host it on a chosen cloud (AWS, Google cloud etc)
12. Create an application for data analysis (Use any front-end tool and NoSQL Database as backend), Dockerize the same and host it on a chosen cloud (AWS, Google cloud etc)

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS214A
Course Name	Advanced Cryptography	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1:
Introduction, Information security and cryptography, Basic terminology and concepts, Symmetric-key encryption, Digital signatures, Public-key cryptography, Hash functions, Protocols and mechanisms, Key establishment, management, and certification, Pseudorandom numbers and sequences, Classes of attacks and security models. 8 hrs.

Unit 2:
Probability theory , Information theory, Complexity theory, Number theory, Abstract algebra, Finite fields, The integer factorization problem, The RSA problem, The Diffie-Hellman problem, Composite moduli. 8 hrs.

Unit 3:
Fields, Characteristic of a field, prime fields, Arithmetic of polynomials over fields. Field extensions, Galois group of a field extensions, Fixed field and Galois extensions. Minimum polynomial, Construction of fields with the help of an irreducible polynomial. 8 hrs.

Unit 4:
Splitting field of a polynomial, Separable polynomial and Separable extensions. Construction of finite fields and their structure. Enumeration of irreducible polynomials over finite fields. Fundamental theorem of Galois Theory. Cyclotomic extensions, Geometric constructions and Galois theory of Equations (Statement only of Abel Ruffini), Solving Cubic and Bi-quadratic polynomials using radicals. 8 hrs.

Unit 5:
Introduction to Key Establishment Protocols, Key transport based on symmetric encryption, Key agreement based on symmetric techniques, Key transport based on public-key encryption, Key agreement based on asymmetric techniques, Secret sharing, Key Management Techniques, Techniques for distributing public keys, Techniques for controlling key usage, Key management involving multiple domains. 8 hrs.

Text Book:

1. William Stallings, “Cryptography and Network Security: Principles and Practice”, 7th Edition, 2017
2. Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, “Handbook of Applied Cryptography” CRC Press, 1997

References:

1. Stinson. D., “Cryptography: Theory and Practice”, Chapman & Hall/CRC, 3rd edition, 2012
2. Atul Kahate, “Cryptography and Network Security”, Tata McGraw-Hill, 2003

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply number theory concepts to the field of cryptography.			**3**
CO2:	Analyze various symmetric and asymmetric cryptosystems and types of attacks on the cryptosystems.			**3**
CO3:	Demonstrate the usage of cryptographic algorithms in the field of security.	**2**	**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P:3-0-0]	Course Code	MSCS214B
Course Name	**Pattern Classification**	**Total No. of Lecture Hours**	**40**
CIE Marks	**50**	**SEE Marks**	**50**

Unit 1:

Introduction to Pattern Classification: Pattern recognition systems, Design Cycle, Learning and adaptation. **Bayesian Decision Theory:** Continuous Features, Minimum-error-rate classification, Classifiers, Discriminant functions, Decision surfaces, Normal density and discriminant functions, Discrete features — 8 hrs.

Unit-2:

Parameter Estimation Methods: Maximum-Likelihood estimation: Gaussian case, Bayesian estimation, Bayesian parameter estimation: Gaussian case — 8 hrs.

Unit-3:

Nonparametric techniques for density estimation: Density estimation, Parzen-window method, K-Nearest Neighbour method — 8 hrs.

Unit-4:

Unsupervised learning and clustering: Mixture densities and Identifiability, Maximum-Likelihood estimates, Applications of normal mixtures, Unsupervised Bayesian Learning, Data description and clustering, Criterion functions for clustering, Hierarchical clustering. — 8 hrs.

Unit-5:

Linear discriminant functions: Linear discriminant functions and decision surfaces**,** Generalised linear discriminant functions, Two category linearly separable case, Minimising the perceptron criteria function, Non separable behaviour**.** — 8 hrs.

Text Book:

1. R.O.Duda, P.E.Hart and D.G.Stork., " Pattern Classification ", 2nd edition John Wiley, 2001

References:

1. Christopher M. Bishop, “Pattern Recognition and Machine Learning”, Springer, 2009

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply the concepts Pattern Classification using Machine Learning Classifiers			**3**
CO2:	Analyze and identify the data patterns.			**3**
CO3:	Implement pattern recognition using supervised /unsupervised classifiers.	**2**	**2**	

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS215D
Course Name	FOG AND EDGE COMPUTING	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit-1

Fog computing requirements when applied to IoT: Scalability, Interoperability, Fog- IoT architectural model, Challenges on IoT Stack Model via TCP/IP Architecture, Data Management, filtering, Event Management, Device Management, cloudification, virtualization, security and privacy issues. Integrating IoT, Fog, Cloud Infrastructures: Methodology, Integrated C2F2T Literature by Modelling Technique by Use-Case Scenarios, Integrated C2F2T Literature by Metrics. 8 hrs.

Unit-2

Exploiting Fog Computing in Health Monitoring: An Architecture of a HealthMonitoring IoT- Based System with Fog Computing, Fog Computing Services in Smart E-Health Gateways, Discussion of Connected Components. Fog Computing Model for Evolving Smart Transportation Applications: Introduction , Data-Driven Intelligent Transportation Systems , Fog Computing for Smart Transportation Applications Case Study: Intelligent Traffic Lights Management (ITLM) System. 8 hrs.

Unit-3

Software Defined Networking and application in Fog Computing: Open Flow Protocol, Open Flow Switch, SDN in Fog Computing, Home Network using SDN. Security and Privacy issues: Trust and privacy issues in IoT Network, web Semantics and trust Management for Fog Computing, Machine Learning based security in Fog Computing, Cyber- Physical Energy Systems over Fog Computing 8 hrs.

Unit-4

Introduction to Edge Computing Scenarios and Use cases - Edge computing purpose and definition, Edge computing use cases, Edge computing hardware architectures, Edge platforms, Edge vs Fog Computing, Communication Models - Edge, Fog, and M2M. 8 hrs.

Unit-5

IoT Architecture and Core IoT Modules -A connected ecosystem,IoT versus machine-to-machine versus, SCADA, The value of a network and Metcalfe's and Beckstrom's laws, IoT and edge architecture, Role of an architect, Understanding Implementations with the examples- Edge computing with RaspberryPi, Industrial, and Commercial IoT and Edge, and Edge computing and solutions. 8 hrs.

Text Book:

1. Fog Computing: Theory and Practice by Assad Abbas, Samee U. Khan, Albert Y. Zomaya
2. IoT and Edge Computing for Architects - Second Edition, by Perry Lea, Publisher: Packt Publishing, 2020, ISBN: 9781839214806
3. David Jensen, “Beginning Azure IoT **Edge Computing**: Extending the Cloud to the Intelligent **Edge**, MICROSOFT AZURE

References:

1. Fog and Edge Computing: Principles and Paradigms (Wiley Series on Parallel and Distributed Computing) by RajkumarBuyya and Satish Narayana Srirama
2. FlavioBonomi, Rodolfo Milito, Jiang Zhu, SateeshAddepalli, - Fog Computing and Its Role in the Internet of Things‖, MCC'12, August 17, 2012, Helsinki, Finland. Copyright 2012 ACM 978- 1-4503-1519-7/12/08... $15.00.
3. Shanhe Yi, Cheng Li, Qun Li, ―A Survey of Fog Computing: Concepts, Applications and Issues‖, Mobidata'15, ACM 978-1-4503-3524-9/15/06, DOI: 10.1145/2757384.2757397, June 21, 2015, Hangzhou, China.
4. Raspberry Pi Cookbook, 3rd Edition, by Simon Monk, Publisher: O'Reilly Media, Inc., 2019, ISBN: 978149204322

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply the concept of edge computing, distributed computing and IoT for real time applications.			**3**
CO2:	Analyze data and application requirements and pertaining issues.			**3**
CO3:	Implement the concepts of fog and cloud computing using modern tools for deploying relevant infrastructures.	**2**	**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS214D
Course Name	Quantum Computing	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1:

Foundation: Overview of traditional computing, Church-Turing thesis, circuit model of computation, reversible computation, quantum physics, quantum physics and computation, Dirac notation and Hilbert Spaces, dual vectors, operators, the spectral theorem, functions of operators, tensor products, Schmidt decomposition theorem. 8 hrs.

Unit 2:

Qubits And Quantum Model of Computation: State of a quantum system, time evolution of a closed system, composite systems, measurement, mixed states and general quantum operations, quantum circuit model, quantum gates, universal sets of quantum gates, unitary transformations, quantum circuits. 8 hrs.

Unit 3:

Quantum Algorithms-I: Superdense coding, quantum teleportation, applications of teleportation, probabilistic versus quantum algorithms, phase kick-back, the Deutsch algorithm, the Deutsch- Jozsa algorithm, Simon's algorithm, Quantum phase estimation and quantum Fourier Transform, eigenvalue estimation. 8 hrs.

Unit 4:

Quantum Algorithms – II: Order-finding problem, eigenvalue estimation approach to order finding, Shor's algorithm for order finding, finding discrete logarithms, hidden subgroups, Grover's quantum search algorithm, amplitude amplification, quantum amplitude estimation, quantum counting, searching without knowing the success probability. 8 hrs.

Unit 5:

Quantum Computational Complexity and Error Correction: Computational complexity, black-box model, lower bounds for searching, general black-box lower bounds, polynomial method, block sensitivity, adversary methods, classical error correction, classical three-bit code, fault tolerance, quantum error correction, three-qubit and nine-qubit quantum codes, fault-tolerant quantum computation. 8 hrs.

Text Book:

1. "An introduction to Quantum Computing" by P. Kaye, R. Laflamme, and M. Mosca, 1st Edition, Oxford University Press (2007).
2. Chris Bernhardt, Quantum Computing for Everyone, The MIT Press,Cambridge, (2020)

References:

1. "Quantum Computing" by V. Sahni, 1st Edition McGrawHill Edition (2007)
2. Quantum Computing, A Gentle Introduction, Eleanor G. Rieffel and Wolfgang H. Polak MIT press (2014)

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply the concepts of quantum computing and quantum mechanism.			**3**
CO2:	Analyze the computational model in quantum computing.			**3**
CO3:	Demonstrate the result of recent research paper on quantum computing.	**2**	**2**	

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS215A
Course Name	Big Data Technologies	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1:
Big Data Processing Frameworks: MapReduce programming model. Apache Hadoop ecosystem - HDFS, YARN, Hive and Pig. Batch vs. Stream Processing. 8 hrs.

Unit 2:
Introduction to Data Analysis with Spark: Introduction to Apache Spark, A unified Spark, Users of Spark, Storage layers for Spark. Programming with RDDs: RDD Basics, Creating RDDs, RDD Operations, Passing functions to Spark, Common Transformations and Actions, Persistence. 8 hrs.

Unit 3:
Data Ingestion and Integration: Data ingestion tools: Apache Kafka, Flume, Sqoop. ETL (Extract, Transform, Load) processes. Real-time data streaming. Data quality and preprocessing. 8 hrs.

Unit 4:
Big Data Analytics and Visualization: Descriptive, predictive, and prescriptive analytics. Tools: Apache Zeppelin, Tableau, Power BI. Data visualization principles. Time-series analysis. 8 hrs.

Unit 5:
Big Data Security and Ethics: Data privacy and GDPR compliance. Security in distributed systems. Ethical considerations in big data (bias, fairness). Anonymization techniques. 8 hrs.

Text Book:

1. Karau, H., et al. (2015). Learning Spark: Lightning-Fast Big Data Analysis. O’Reilly Media
2. White, T. (2015). Hadoop: The Definitive Guide (4^{th} ed.). O’Reilly Media.

References:

1. Kreps, J. (2014). Kafka: The Definitive Guide. O’Reilly Media.
2. Casters, M., et al. (2010). Pentaho Kettle Solutions: Building Open Source ETL Solutions. Wiley.
3. Provost, F., & Fawcett, T. (2013). Data Science for Business. O’Reilly Media.
4. Few, S. (2012). Show Me the Numbers: Designing Tables and Graphs to Enlighten. Analytics Press.
5. Solove, D. J. (2011). Nothing to Hide: The False Tradeoff between Privacy and Security. Yale University Press.
6. Mayer-Schönberger, V., & Cukier, K. (2013). Big Data: A Revolution That Will Transform How We Live, Work, and Think. Houghton Mifflin Harcourt.

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply the foundational concepts of Big Data, Processing, Solutions, Security and Governance.			**3**
CO2:	Analyze knowledge of Big Data to solve complex real-world problems			**3**
CO3:	Implement Case Studies to find solutions for real time Big Data scenarios.	**2**		

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS215B
Course Name	**System and Network Security**	**Total No. of Lecture Hours**	**40**
CIE Marks	**50**	**SEE Marks**	**50**

Unit 1:

Fundamentals of Computer Security: Objectives of Computer Security, Issues Involved in Computer Security, Privacy and Ethics. **Linux Security Basics:** Users and Groups, Permissions and Access Control List, Running Command with Privileges, Authentication. **Set-UID Programs:** The Need for Privileged Programs, The Set-UID Mechanism, What Can Go Wrong: What Happened to Superman, Attack Surfaces of Set-UID Programs, Invoking Other Programs, Principle of Least Privilege. **Shellcode:** Introduction, Writing Assembly Code, Writing Shellcode: the Basic Idea, Approach 1: The Stack Approach, Approach 2: the Code Segment Approach, Writing 64-bit Shellcode, A Generic Shellcode. **Reverse Shell:** Introduction, File Descriptor and Redirection, Redirecting Input/Output to a TCP Connection, Reverse Shell

8 hrs.

Unit 2:

Network Security Network Security Basics: The Organization of the Network-Security Module, IP Address and Network Interface, The Life-Cycle of Packet and Protocol Layers, Packet Sniffing, Packet Spoofing, More About Scapy, Containers and Networks. **Attacks on the TCP Protocol:** Introduction, How the TCP Protocol Works, SYN Flooding Attack, TCP Reset Attack, TCP Session Hijacking Attack, The Mitnick Attack

8 hrs.

Unit 3:

Firewall: Introduction, Types of Firewalls, Implementing a Simple Firewall Using Netfilter, Configuring Linux Firewall Using iptables, Connection Tracking and Stateful Firewall, Application/Proxy Firewall and Web Proxy. **Virtual Private Network:** Introduction, An Overview of How TLS/SSL VPN Works, Creating and Using the TUN Interface, Implementing the IP Tunnel, Pulling the Return Packets Back to Client, Testing VPN, Connecting Two Private Networks Using TUN, Bridging Two Private Networks Using TAP

8 hrs.

Unit 4:

Web Security Basics: The Web Architecture, Web Browser, Web Server: HTTP Server and Web Applications, Browser-Server Communication: The HTTP Protocol, Cookies and Sessions, Sandboxing JavaScript, Ajax Request and Security. **Cross-Site Request Forgery:** Cross-Site Requests and Its Problems, Cross-Site Request Forgery Attack, CSRF Attacks on HTTP GET Services, CSRF Attacks on HTTP POST Services, Countermeasures. **Cross-Site Scripting Attack:** The Cross-Site Scripting Attack, XSS Attacks in Action, Achieving Self-Propagation, Preventing XSS attacks, JavaScript Code Injection Attacks in General.**SQL Injection Attack:** A Brief Tutorial of SQL, Interacting with Database in Web Application, Launching SQL Injection Attacks, The Fundamental Cause, Countermeasures

8 hrs.

Unit 5:

Risk Analysis: Theory, Risk Analysis in Computer Security, Terminology, Application, **Developing Secure Computer Systems:** External Security Measures, Structure of a Computer System, Secure Computer System Issues, **Security Models:** Specification and Verification,
Security Models, TCSEC.

8 hrs

.

Text Book:

1. Gregory B. White (SecureLogix, San Antonio, Texas, USA) (Author) , Eric A. Fisch (KPMG LLP, Plano, Texas, USA) (Author) , Udo W. Pooch (Texas A & M University, College Station, Texas, USA) (Author), CRC Press Inc; 1st edition (10 August 1995)
2. Computer Internet Security: A hands-on approach by Wenliang Du; 3rd ed. edition (1 May 2022)

References:

1. Network Security Essentials: Applications and Standards by William Stalling, Pearson Education; 6th edition (27 February 2018); Pearson Education
2. Firewalls and Internet Security: Repelling the Wily Hacker by William R. Cheswick, Steven M. Bellovin, and Aviel D. Rubin, Addison-Wesley; 2nd edition (4 March 2003)
3. Applied Network Security Monitoring by Chris Sanders and Jason Smith, Syngress; Illustrated edition (20 January 2014)
4. NETWORK AND SYSTEM SECURITY SECOND EDITION by John R. Vacca, Syngress Media,U.S.; 2nd edition (8 October 2013)

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply Software, Network, and web security to secure real-time applications.			**3**
CO2:	Analyze the risk involved in the Software, Network, and Web application			**3**
CO3:	Demonstrate the Software, Network, and Web security attacks and protection.	**2**	**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS215C
Course Name	Applied Natural Language Processing	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1:
Introduction to Computational Linguistics, Word Meaning, Distributional Semantics, Word Sense Disambiguation, Sequence Models, N-gram Language Models, Feed forward Neural Language Models, Word Embedding, Recurrent Neural Language Models, Tokenization, Lemmatization, Stemming, Sentence Segmentation, POS Tagging and Sequence Labeling, Structured Perceptron, Viterbi – Loss, Augmented Structured Prediction. 8 hrs.

Unit 2:
Information Extraction from Text, Sequential Labeling, Named Entity Recognition, Semantic Lexicon Induction, Relation Extraction, Paraphrases Inference Rules, Summarization, Event Extraction, Opinion Extraction, Temporal Information Extraction, Open Information Extraction, Knowledge based Population, Narrative Event Chains and Script Learning. 8 hrs.

Unit 3:
Machine Translation, Encoder-Decoder Models, Beam Search, Attention Models, Multilingual Models, Syntax, Trees, Parsing, Transition based Dependency Parsing, Graph based Dependency Parsing, Transfer Learning, Deep Generative Models for Natural Language Data, Information Extraction with AQL-Conversational AI. 8 hrs.

Unit 4:
Application: Spelling Correction, Sentiment Analysis, Word Sense Disambiguation, Text Classification, Machine Translation, Question Answering System, Intent Detection, False Fact Detection. 8 hrs.

Unit 5:
Real World NLP Challenges – Information Extraction, Question Answering, Dialog Engines. Self-Determined Learning (SDL): Long Short-Term Memory (LSTM) networks, Sequence to sequence models and attention 8 hrs.

Text Book:

1. Dan Jurafsky, James H. Martin, "Speech and Language Processing", Prentice Hall, 2008
2. Emily Bender, "Linguistics Fundamentals for NLP", Morgan Claypool Publishers, 2013

References:

1. Jacob Eisenstein, "Natural Language Processing", The MIT Press 2019 (2020)
2. Natural Language Processing in Action, Hobson Lane, Cole Howard, Hannes Max Hapke, Manning, 2019.
3. Chris Manning, HinrichSchutze, "Foundations of Statistical Natural Language Processing", The MIT Press 1999
4. Pushpak Bhattacharyya, "Machine Translation", CRC Press,2015

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply various text processing techniques for NLP applications			**3**
CO2:	Analyze linguistic structures, text extraction and language translation techniques for real-world NLP applications.			**3**
CO3:	Implement natural language processing applications using various libraries.	**2**	**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 3-0-0]	Course Code	MSCS215D
Course Name	High-Performance Computer Architectures	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	50

Unit 1:

Fundamentals of computer design: Introduction; Classes computers; Defining computer architecture; Trends in Technology; Trends in power in Integrated Circuits; Trends in cost; Dependability, Measuring, reporting and summarizing Performance attributes; Quantitative Principles of computer design. 8 hrs.

Unit 2:

Introduction to Parallel Programming: Motivation, Scope of Parallel Computing, Principles of Parallel Algorithm design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for containing Interaction Overheads, Parallel Algorithms Models. 8 hrs.

Unit 3:

Programming Using the Message Passing Paradigm: Principles of Message Passing Programming, Building Blocks, MPI, Topologies and Embedding, Overlapping Communication with computation, Collective Communication and computation operations, Groups and Communicators. 8 hrs.

Unit 4:

GPU Architectures: Introduction to Graphics Processing Units, Detecting and Enhancing Loop-Level Parallelism, Mobile versus Server GPUs and Tesla versus Core i7, GPU programming using CUDA 8 hrs.

Unit 5:

Heterogeneous Computing: From Serial To Parallel Programming Using OpenAcc, a simple data-parallel loop, the OpenAcc kernels and parallel construct, the various forms of OpenAcc parallelism.

Intel FPGAs: Introduction to Intel FPGAs and Intel Quartus Prime Design Software- FPGA design and implementation. 8 hrs.

Intel SoC FPGAs: Introduction to Intel SoC FPGAs - IP design and Platform designer, Embedded System design using Cyclone V and ARM -SoC Design Flow

Text Book:

1. Computer Architecture: A Quantitative Approach, John L Hennessy, David A Patterson, Elsevier, 5th edition; 2011, ISBN: 9780123838728.
2. Introduction to Parallel Computing, Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar : 2nd edition, Pearson Education, 2007

References:

1. Parallel Programming with Open ACC, Rob Farber 1st edition, 2016, ISBN :9780124103979
2. Using Open MP Portable Shared Memory Parallel Programming, Barbara Chapman, Gabriele Jost, Ruud van der Pas, 2008, The MIT Press, ISBN: 978-0-262-53302-7.

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Apply the fundamental concepts of parallel computer architecture			**3**
CO2:	Analyze the performance of parallel programming			**3**
CO3:	Design and Demonstrate parallel computing concepts for suitable applications	**3**		

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	3[L-T-P: 0-0-3]	Course Code	MSCS286
Course Name	Mini Project/Technology-Based Societal Project	Total No. of Lecture Hours	40
CIE Marks	50	SEE Marks	--

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Conduct literature survey and formulate a real-world problem to address Sustainable Development Goals.	**3**		
CO2:	Design and develop prototype and/or techniques that contribute to the software / hardware solution of the project using appropriate tools.	**2**		
CO3:	Test and validate the conformance of the developed prototype against the original requirements of the problem and exhibiting integrity and ethical behaviour in engineering practice.	**3**		
CO4:	Use formal and informal communications, make presentations and prepare technical document		**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	**2[L-T-P: 0-0-2]**	**Course Code**	**MSCSL207**
Course Name	**Deep Learning and Generative AI Lab**	**Total No. of Lecture Hours**	**--**
CIE Marks	**50**	**SEE Marks**	**50**

Evaluation Criteria:

I. CIE: 50 Marks

20 marks	**Weekly Evaluation:** For ten weeks, 20 Marks each. At the end of lab hour, each student should upload the code with output in their respective GitHub account. The code should be appropriately commented and suitable variable names should be provided. Should always look forward to write optimized code for every lab program. Should not copy the code from peers.
30 Marks	**Lab Test:** Open Lab test. Can use internet resources. Should build and train models using ANN, CNN, RNN, Transformers or Gen AI Models for the given dataset. If required pre-trained models can be used. Datasets can downloaded from internet resources. **Note:** During Lab test hour i. Should not use any AI tools (Like ChatGPT, DeekSeek etc.,) for designing the models and code generation. ii. Should not use any online Chat tools or Emails for exchange of answers with peers.

II. SEE: 50 Marks

i. Demonstration of Capstone project for Image classification or text Generation tasks for large size datasets to solve real world problems. Using publicly available datasets/by creating new datasets.

ii. Project report preparation and presentation.

iii. Final Project work evaluation by External examiners.

List of Lab program the students should implement over ten weeks:

Week-1: Single Unit Perceptron

- Design a single unit perceptron for classification of a linearly separable binary dataset without using pre-defined models. Use the Perceptron() from sklearn.
- Identify the problem with single unit Perceptron. Classify using Or-, And- and Xor-ed data and analysis the result.

Week-2: Artificial Neural Network

- Implementing the Backpropagation algorithm and test the same using IRIS and Ionosphere data sets.

- Take training data percentage 80%, test data percentage 20%. Number of epochs as 100 and 300. Use number of hidden layers as two.

- Note down the accuracies for the following set of experiments on the given ANN and compare the results.

i. Using activation functions Sigmoid, ReLu, Tanh

ii. Using optimizers SGD, ADAM

iii. Varying learning rate in SGD as 0.0001, 0.0005, 5.

iv. Using loss mean squared error and categorical cross entropy.

Week-3: Design Deep forward Neural Network for image classification

- Design and implement an Image classification model to classify a dataset of images using Deep Feed Forward NN.

- Use the MNIST datasets.

- Record the accuracy corresponding to the number of epochs 5, 50, 100.

- Repeat for CIFAR10 datasets. Note down the changes made and the accuracies obtained.

Week-4: Building a CNN for Image Classification

- Design and implement a CNN model (with 4+ layers of convolutions) to classify multi category image datasets.

- Use the Fashion MNIST dataset.

- Set the number of Epoch as 5, 10 and 20.

- Make the necessary changes whenever required. Record the accuracy corresponding to the number of epochs.

- Record the time required to run the program, using CPU as well as using GPU in Colab.

Week-5: Building a CNN for Image Classification

- Design and implement a CNN model (with 4+ layers of convolutions) to classify multi category image datasets. Use the concept of regularization and dropout while designing the CNN model. U

- Use CIFAR-10 dataset.

- Record the Training accuracy and Test accuracy corresponding to the following architectures:

a. Base Model

b. Model with L1 Regularization

c. Model with L2 Regularization

d. Model with Dropout

e. Model with both L2 (or L1) and Dropout

Week-6: Transfer Learning

- Implement the standard LeNet, AlexNet, VGG CNN architecture model to classify multicategory image dataset.
- MNIST handwritten digits (0-9)
- Note down accuracies obtained for epochs 5, 50, 250.

Week-7: Sentiment Analysis

- Implement RNN, LSTM, Bi-directional and GRU's for sentiment analysis
- Use movie reviews dataset
- Use the concept of Embedding layer.

Week-8: Image Generation

- Implement Generative Adversarial Networks to generate realistic Images.
- Use MNIST dataset.
- Use Fashion MNIST

Week-9: Text Generation using Transformers

- Implement a transformer model for text generation using the GPT-2 architecture.
- Fine-tune the model on a custom text dataset (e.g., Shakespearean text).
- Experiment with different prompt designs and analyze the generated text.

Week-10: Large Language Models (LLMs) and Prompt Engineering

- Fine-tune a pre-trained GPT model (e.g., GPT-3 or GPT-4) for a specific text generation task.
- Experiment with prompt engineering techniques to improve model outputs.

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Analyze the given dataset for ANN, CNN, RNN and Generative Models.	**3**		
CO2:	Design and implementation of ANNs, CNNs, RNN and transfer learning for computer vision tasks.	**3**		
CO3:	Implementation of Generative AI models for Generating Texts	**3**		
CO4:	Demonstration of developing Deep Learning and Generative AI models for real-world computer vision and text generation problems using publicly available datasets/by creating new datasets.	**2**	**2**	**3**

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	1[L-T-P: 0-0-2]	Course Code	MSCS258A
Course Name	**ADVANCED WEB TECHNOLOGY**	**Total No. of Lecture Hours**	**--**
CIE Marks	**50**	**SEE Marks**	**50**

The Course content is not limited to the following Topics:

- Introduction: What's MERN, MERN Component, Why MERN. Hello World, Project Setup, Express, Separate Script file, JSX Transform, Older Browser Support. React Components: Issue Tracker, React Classes, Composing Components, Passing Data using properties, passing data using children, Dynamic composition.
- React State: Initial State, Async state Initialization, Updating State, Lifting State up, Event handling, Stateless components, designing components, Express, REST API, Graph QL, Graph QL Schema file, Input validation, displaying error. MongoDB.
- Architecture and ESLint, Modularization and Webpack, React Router.
- React Forms, React Bootstrap, Server Rendering.
- Advanced Features, Authentication, Deployment.

Text Books:

1. Vasan Subramanian Pro MEAN Stack (Full Stack Web App Development with Mongo, Express, React and Node. Apress.

References:

1 Full Stack Developer Your Essential Guide To The Everyday Skills Expected Of A Modern Full Stack Web Developer by Northwood, Apress.

2 Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, and Docker Paperback.

3 Manu Sharma, Full Stack development with MongoDB, bpb Publications, 2022.

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Independently develop a solution for a given website specification using web technologies	**2**		
CO2:	Prepare a technical document for full stack development		**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	1[L-T-P: 0-0-2]	Course Code	MSCS258B
Course Name	**AUGMENTED & VIRTUAL REALITY**	**Total No. of Lecture Hours**	**--**
CIE Marks	**50**	**SEE Marks**	**50**

The content is not limited to the following Topic:

- Introduction to Augmented Reality - Part 1: History of AR, AR Scenarios, the future of AR, Applications of AR. Calibration and Registration: Transformations, Coordinate Systems.
- Introduction to Augmented Reality - Part 2: Projections, Image formation in a pinhole camera, camera calibration, camera calibration techniques, camera calibration tools.
- Pose Estimation and Tracking: Pose Estimation; Pose Tracking in AR, Classification of Tracking, Stationary Tracking Systems, Mobile Sensor-Based Tracking, Optical Tracking, Hybrid Tracking, Marker-Based Tracking and AR, Diminished Reality, Markerless Tracking and AR.
- Display Data with Core UI components in Unity, Responding to user Events for Interactive UIs, Inventory and Advanced Uis.

  Playing and Manipulating sounds, Creating 3D Objects, Terrains, Textures and Materials

Text Books:

1 Matt Smith, Shaun Ferns, Unity 2021 Cookbook, Fourth Edition, Packt Publishing, 2021.

2 Chetankumar G Shetty, "Augmented Reality: Theory, Design and Development", McGrawHill Publications 2020.

References:

1. Mike Gieg, Sams Teach Yourself Unity 2018 Game Development in 24 Hours, Pearson, 2018
2. Jonathan Linowes, Krystian Babilinski, "Augmented Reality for Developers: Build practical augmented reality applications with Unity, ARCore, ARKit and Vuforia", Paperback – Import, Packt Publishing Limited, 9 October 2017.

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Independently develop a solution for a given application using AR and VR aspects	**2**		
CO2:	Prepare a technical document using Unity3D for chosen application		**2**	

B.M.S. COLLEGE OF ENGINEERING, BENGALURU
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Autonomous Institute affiliated to VTU

II SEMESTER

Credits	1[L-T-P: 0-0-2]	Course Code	MSCS258C
Course Name	**Competitive Coding**	**Total No. of Lecture Hours**	**--**
CIE Marks	**50**	**SEE Marks**	**50**

The Course content is not limited to the following Topics:

- Revisit Algorithms to deepen problem solving skills which have learnt in the previous courses such as Data Structures & Algorithms for Problem Solving.
- Competitive Programming Strategies such as Problem formulation, Time Complexity and Space Complexity analysis, Handling input/output efficiently especially for large datasets.
- Practice hands-on problem-solving skill on various platforms like LeetCode, HackerRank etc. which will help in preparing for interviews, Hackathons, Google Talent Search, ACM ICPC.

Text Books:

1 Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein 3rd MIT Press 2009

2 Competitive Programming 4 - Book 1: The Lower Bound of Programming Contests in the 2020 Steven Halim, Felix Halim, Suhendry Effendy 1st Lulu.com 2020

3 Competitive Programming 4 - Book 2: The Lower Bound of Programming Contests in the 2020 Steven Halim, Felix Halim, Suhendry Effendy 1st Lulu.com 2020

References:

1 Elements of Programming Interviews: The Insiders' Guide Adnan Aziz, Tsung-Hsien Lee, Amit Prakash 2nd Amazon Digital Services 2016

2 Cracking the Coding Interview: 189 Programming Questions and Solutions, Gayle Laakmann McDowell, 6th HYESHOM 2019

Course Outcomes (COs):				
On Completion of the course, the students will be able to		**PO1**	**PO2**	**PO3**
CO1:	Independently develop a solution for a given problem of different levels of complexity using efficient algorithms and data structure	**2**		
CO2:	Prepare a technical document for the developed solution		**2**	