



Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Information Technology, Pune-48
(An Autonomous Institute affiliated to Savitribai Phule Pune University)

T. Y. B. TECH CSE-DS
SEMESTER-VI SYLLABUS
(PATTERN 2023-NEP)



Bansilal Ramnath Agarwal Charitable Trust's
Vishwakarma Institute of Information Technology, Pune-48
(An Autonomous Institute affiliated to Savitribai Phule Pune University)

Vision of Department

"Excellence in Data Science to empower the future of technology with holistic development."

Mission of Department

- **M1:** To impart quality education with contemporary industry needs using emerging Machine Learning & Data Science techniques.
- **M2:** To cultivate a research-oriented mindset and comprehensive professional skills.
- **M3:** To equip learners with interdisciplinary skill sets to cater the needs of the industry and society.

Program Specific Outcomes (PSO's)

PSO-1: Apply Data Science techniques to extract valuable insights from real-world data for informed decision-making.

PSO-2: Apply Machine Learning and AI concepts for predictive modelling and intelligent system development.

Program Outcomes (PO's)

At the end of program, students should be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety and the cultural, social and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and

research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage:** Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings.

10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning:** Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Contents

Sr. No.	Title	Page No.
1.	Vision, Mission, PSO's, PO's	2
2.	Course Structure –	5
3.	CD32231	DEEP LEARNING 6
4.	CD32232	COMPILER DESIGN 9
5.	CD32233	AGENTIC AI 12
6.	MDM32236	RISK ANALYSIS AND MANAGEMENT WITH STANDARDS 14
7.	CD32234A	DEVOPS 16
8.	CD32234B	NATURAL LANGUAGE PROCESSING 19
9.	ECI 32234H	ADVANCED MAINFRAME TECHNOLOGIES 22
10.	CD32235	* COURSERA COURSES
11.	CD32235A	IBM FULL STACK SOFTWARE DEVELOPER
12.	CD32235B	IBM CYBERSECURITY ANALYST
13.	CD32235C	MICROSOFT CYBERSECURITY ANALYST
14.	CD32235D	GOOGLE CYBERSECURITY
15.	CD32235E	GOOGLE DIGITAL MARKETING AND E- COMMERCE
16.	CD32235F	IBM MAINFRAME DEVELOPER
17.	CD32235G	SALESFORCE SALES DEVELOPMENT REPRESENTATIVE
18.	CD32235H	SAP TECHNOLOGY CONSULTANT
19.	CD32237	VOCATIONAL SKILL-PROJECT II
20.	CD3223M3	LRQA



Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Information Technology, Pune-48

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

T.Y. B. TECH. (CSE-DS), SEMESTER-VI (PATTERN 2023)


AY 2025-26

	Course Name	Course Type	Teaching Scheme (Hrs/Week)			Assessment Scheme (100-mark scale)										Credits
			Th	L	T	ISA					ESA				Total	
						HA	SCE	PPT	GD	CIE	ESE	PR	OR	TW		
CD32231	DEEP LEARNING	TH	3	2	-	-	20	-	-	20	40	20	-	-	100	4
CD32232	COMPILER DESIGN	TH	2	2	-	-	20	-	-	20	40	-	20	-	100	3
CD32233	AGENTIC AI	TH	2	2	-	-	20	-	-	20	40	-	20	-	100	3
CD32234X	PROGRAM ELECTIVE	TH	3	-	1	-	20	-	-	20	40	-	-	20	100	4
MDM32236	RISK ANALYSIS AND MANAGEMENT WITH STANDARDS	TH	2	-	1	10	20	-	-	10	-	-	40	20	100	3
CD32235	COURSERA COURSES	TH	2	-	-	-	-	-	-	-	100	-	-	-	100	4
M3	MANDATORY COURSE	AU	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CD32237	VOCATIONAL SKILL-PROJECT II	CE	-	4	-	-	-	-	-	-	-	-	-	50	50	1
	TOTAL		14	10	1	10	100	-	-	90	260	20	80	90	650	22

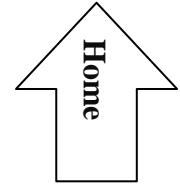
PROGRAM ELECTIVES			
Course Code	Course Name	Course Code	Course Name
CD32234A	DEVOPS	CD32234B	NATURAL LANGUAGE PROCESSING
ECI 32234H	ADVANCED MAINFRAME TECHNOLOGIES		

COURSERA COURSES			
Course Code	Course Name	Course Code	Course Name
CD32235A	IBM FULL STACK SOFTWARE DEVELOPER	CD32235E	GOOGLE DIGITAL MARKETING AND E-COMMERCE
CD32235B	IBM CYBERSECURITY ANALYST	CD32235F	IBM MAINFRAME DEVELOPER
CD32235C	MICROSOFT CYBERSECURITY ANALYST	CD32235G	SALESFORCE SALES DEVELOPMENT REPRESENTATIVE
CD32235D	GOOGLE CYBERSECURITY	CD32235H	SAP TECHNOLOGY CONSULTANT


BQS Chairman
Sh. Supriya Asri


Dean Academics
Namrata Wavalkar


Director
Dr. Vivek Deshpande



CD32231: Deep Learning

Teaching Scheme	Examination Scheme				
Credits: 04	CIE	SCE	ESE	PR	Total
Lecture (L): 03 hrs / week					
Practical (P): 02 hr / week	20	20	40	20	100
Prerequisites: Python programming , Linear Algebra , Machine Learning					
Course Objectives: <ol style="list-style-type: none"> 1. To understand the basics and evolution of deep learning. 2. To learn neural network architectures and training mechanisms. 3. To apply activation functions, loss functions, and optimization techniques. 4. To design CNN models for image-based applications. 5. To implement RNN models for sequential data. 6. To explore deep generative models and modern DL frameworks. 					
Course Outcomes: After completion of the course, student will be able to: <ol style="list-style-type: none"> 1. Explain the evolution of deep learning, differentiate it from traditional machine learning, and implement basic neural network models such as perceptron, MLP, and feedforward networks. 2. Apply forward and backpropagation algorithms, select appropriate activation and loss f f functions, and analyze training challenges such as vanishing and exploding gradients. 3. Implement gradient-based optimization algorithms, perform hyperparameter tuning, and apply regularization techniques to improve model generalization. 4. Design, train, and evaluate convolutional neural network architectures for image classification, feature extraction, transfer learning, and object detection tasks. 5. Develop and evaluate recurrent neural network models including RNN, LSTM, and Bidirectional RNN for sequence and time-dependent data applications. 6. Implement deep generative models such as autoencoders, GANs, and transformer-based models using modern deep learning frameworks. 					
Contents					
Unit I: Foundations of Deep Learning					(6 Hrs.)
Evolution from Machine Learning to Deep Learning, Need for Deep Learning in Modern Applications, Deep Learning vs. Traditional Machine Learning. Artificial Neural Networks (ANN): Biological Inspiration and Key Concepts. Perceptron Model: Introduction, Single Layer Perceptron (SLP), Multi-Layer Perceptron (MLP). Feedforward Neural Networks (FNN): Architecture and Working, Deep Feedforward Networks, Hidden Units, EX-OR Problem. Introduction to Neural Network Training: Forward Propagation, Cost Functions, Error Computation.					
Unit II: Training Mechanisms and Activation Functions					(6 Hrs.)
Backpropagation Algorithm: Derivation, Error Backpropagation, Weight Update Mechanism. Loss Functions: Notation and Importance, Loss Functions for Regression, Classification, and Reconstruction. Activation Functions: Need and Properties, Linear and Non-Linear Functions – Step, Sigmoid, Tanh, ReLU, Softmax.Challenges in Training Deep Networks: Vanishing and Exploding Gradient Problems.					

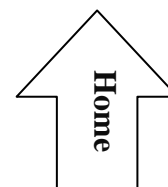
Unit III: Optimization and Regularization Techniques	(6 Hrs.)
<p>Gradient Descent Optimization: Batch Gradient Descent, Stochastic Gradient Descent (SGD). Advanced Optimization Algorithms: Momentum, RMSprop, Adam. Hyperparameter Tuning: Learning Rate, Batch Size, Number of Epochs. Regularization Techniques: L1 and L2 Regularization (Weight Decay), Dropout, Batch Normalization. Overview of Training Stability and Generalization.</p>	
Unit IV: Convolutional Neural Network(CNN)	(6 Hrs.)
<p>Motivation for CNNs in Image Processing ,CNN Architecture: Convolution & Pooling layers, Fully Connected Layers , Feature Extraction and Feature Maps , Types of Pooling: Max Pooling, Average Pooling ,CNN Architectures: AlexNet, VGG, ResNet Transfer Learning and Fine-Tuning ,Object Detection: YOLO, R-CNN, SSD ,Applications of CNNs in Image Classification, Medical Imaging, etc.</p>	
Unit V: Recurrent Network	(6 Hrs.)
<p>Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, Echo State Networks, Leaky Units and Other Strategies for Multiple Time Scales, The Long Short-Term Memory and RNNs, Optimization for LongTerm Dependencies.</p>	
Unit VI: Deep Generative Models	(6 Hrs.)
<p>Introduction to deep generative model: Autoencoders and Variational Autoencoders (VAE) ,Generative Adversarial Networks (GANs) – Architecture and Training, Boltzmann Machine, Deep Belief Networks, Transformers and Self Attention Mechanism ,Introduction to BERT. Overview of Deep Learning Frameworks: TensorFlow, PyTorch, and Keras</p>	
Textbooks:	
<ol style="list-style-type: none"> 1. Deep Learning , Goodfellow, I., Bengio, Y.Courville, A, MIT Press, 2016. 2. Deep Learning S. Sridhar & D. Narashiman — Pearson India, 2025, ISBN: 978-9367138663Charu Agarwal, “Neural Networks and deep learning”, A textbook 3. Neural Networks and Deep Learning: 2nd Edition Charu C. Aggarwal — Springer, 2023, ISBN: 978-3-031-29644-4 	
Reference Books:	
<ol style="list-style-type: none"> 1. Reinforcement Learning: An Introduction by Richard S. Sutton and Andrew G. Barto was published by MIT Press in 2018 (2nd Edition) with ISBN-13: 978-0262039246 and ISBN-10: 0262039249. 2. Deep Learning from Scratch: Building with Python from First Principles by Seth Weidman was published by O’Reilly Media in 2019 with ISBN-13: 978-1492041412 and ISBN-10: 1492041416. 3. Deep Learning for Beginners: Practical Guide with Python and TensorFlow by François Duval was published in 2017 via CreateSpace Independent Publishing Platform with ISBN-13: 978-1982027179 and ISBN-10: 1982027177. 	

List of Courses:

1. Deep Learning Specialization (Coursera) – Series of DL courses by DeepLearning.AI (Andrew Ng). <https://www.coursera.org/specializations/deep-learning>
2. Deep Learning Courses & Tutorials List – Comprehensive list of paid + free courses. <https://onlinecourseing.com/deep-learning-courses-and-tutorials>

List of Assignments:

1. Implement and compare different activation functions (Sigmoid, Tanh, ReLU) in a Multi-Layer Perceptron (MLP) to classify loan applications as approved or rejected based on applicant data. Take "Loan Prediction Dataset" from
2. Kaggle (<https://www.kaggle.com/datasets/itsmesunil/bank-loanmodelling>)
3. Implement and compare different loss functions, gradient descent optimizers, and regularization techniques in a deep learning model for medical image classification (e.g., detecting pneumonia from chest X-ray images).
4. Develop and train a Convolutional Neural Network (CNN) to classify different types of skin diseases using medical images and evaluate the effectiveness of feature extraction, pooling, and fine-tuning in deep learning.
5. Implement real-time object detection using the YOLO (You Only Look Once) model to identify vehicles, pedestrians, and traffic signs, enhancing autonomous vehicle systems.
6. Build and train a Recurrent Neural Network (RNN) for sentiment analysis on a real-world text dataset, applying different RNN architectures (Bidirectional RNN, LSTM, GRU) and evaluating their effectiveness in handling long-term dependencies.
7. Train a Generative Adversarial Network (GAN) to generate realistic images, exploring the impact of GAN architecture and training challenges.
8. Apply SHAP (SHapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations) to explain decisions made by a deep learning model.
9. Implement Tic-Tac-Toe RL Agent (train a bot to play against a human or another AI).



CD32232: Compiler Design

Teaching Scheme	Examination Scheme				
Credits 3.0	CIE	SCE	ESE	OR	Total
Lecture 2 hrs. / week					
Practical 2 hrs. /week	20	20	40	20	100
Prerequisites: Discrete Mathematics / Automata Theory					
Course Objectives: <ol style="list-style-type: none">1. To understand the structure, components, and functioning of a compiler.2. To study the lexical, syntax, and semantic phases of compilation.3. To learn the techniques for generating intermediate code during compilation.4. To understand code optimization methods to improve program performance.5. To learn basic concepts of target code generation for machine-level execution.					
Course Outcomes: <ol style="list-style-type: none">1. Explain the phases, components, and overall structure of a compiler.2. Apply lexical and syntax analysis techniques to design and implement lexical analyzers and parsers for a given grammar.3. Apply semantic analysis methods to generate intermediate code for a programming language construct.4. Analyze and apply basic code optimization techniques to improve the efficiency of compiled programs.					
UNIT 1: Introduction to Compilers & Lexical Analysis					(8 Hrs.)
Introduction to system software, Compiler vs Interpreter, Phases of a compiler (overview), Lexical Analysis: tokens, lexemes, patterns, Regular expressions, Finite automata (NFA, DFA), Lexical analyzer and its role, Introduction to Lex tool Case Study: Spell Checker and Text Editor Applications					
UNIT 2: Syntax Analysis					(6 Hrs.)
Context-free grammar (CFG), Parse tree, Ambiguity in grammar, Top-down parsing: Recursive, descent, LL (1) parsing, First and Follow sets, Bottom-up parsing: Shift-reduce parsing, LR parsing basics, Error handling in parsing. Case Study: Online Form Validation and Command Parsing					
UNIT 3: Semantic Analysis & Intermediate Code Generation					(8 Hrs.)
Syntax-directed translation, Semantic rules, Type checking, Symbol table and its operations, Intermediate code representations, Three-address code, Quadruples, triples, Control flow statements (if, while, for), Runtime environment: activation records Case Study: Banking and E-commerce Applications					
UNIT 4: Code Optimization & Code Generation					(6 Hrs.)
Basic block and flow graph, Machine-independent optimizations: Constant folding, Dead code elimination, Strength reduction, Loop optimization, Code generation process, Register allocation (concept), Introduction to target machine code					

Textbooks:

1. Y. N. Srikant and P. Shankar, Eds., *The Compiler Design Handbook: Optimizations and Machine Code Generation*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2023. **ISBN-13:** 978-1498771818
2. A. V. Aho, M. S. Lam, R. Sethi, and J. D. Ullman, *Compilers: Principles, Techniques, and Tools*, 2nd ed. (Updated). Boston, MA, USA: Pearson, 2023. **ISBN-13:** 978-9357054119

Reference Books:

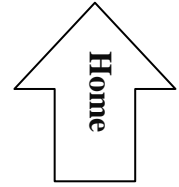
1. K. D. Cooper and L. Torczon, *Engineering a Compiler*, 3rd ed. Cambridge, MA, USA: Morgan Kaufmann, 2023. **ISBN-13:** 978-0128154120
2. N. Sandler, *writing a Compiler: Build a Real Programming Language from Scratch*. San Francisco, CA, USA: No Starch Press, 2024. **ISBN-13:** 978-1718500426

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc25_cs13/preview
2. <https://www.youtube.com/watch?v=rONyHy5amws>

Laboratory experiments

1. Design and implement a lexical analyzer to identify keywords, identifiers, numeric constants, and operators from a given program segment.
2. Develop a lexical analyzer using LEX/Flex to recognize arithmetic operators, integers, floating-point numbers, and parentheses from input expressions.
3. Construct a recursive descent parser to validate arithmetic expressions based on a given grammar.
4. Compute the FIRST and FOLLOW sets for a given grammar and construct an LL (1) predictive parsing table to validate input expressions.
5. Implement a shift–reduce parser to analyze arithmetic expressions and report whether the input string is syntactically correct.
6. Generate three-address code (TAC) for given arithmetic expressions and simple assignment statements.
7. Design and implement a symbol table to store variable information and perform type checking for given declarations and statements.
8. Apply basic code optimization techniques such as constant folding and dead code elimination on the given intermediate code.



CD32233: Agentic AI

Teaching Scheme	Examination Scheme				
Credits: 03	CIE	SCE	ESE(CP)	OR	Total
Lecture (L): 02 hrs / week	20	20	40	20	100
Practical (P): 02 hrs / week					
Prerequisites: Basic Programming Knowledge (preferably Python), Fundamentals of Artificial Intelligence , Data Structures and Algorithms.					
Course Objectives: <ol style="list-style-type: none">1. To understand the fundamentals of agentic AI, including intelligent agents, perception, reasoning, and action.2. To analyze and design agent architectures, planning, and multi-agent coordination strategies.3. To apply memory, context, and learning mechanisms in building autonomous agents.4. To evaluate and implement agentic AI systems in practical scenarios using tools and frameworks.					
Course Outcomes: <p>After completion of the course, student will be able to:</p> <ol style="list-style-type: none">1. Understand the basic concepts of agentic AI, types of agents, and their components.2. Apply planning, goal-based reasoning, and utility-based decision-making in building agents.3. Analyze multi-agent interactions, memory management, and learning mechanisms in autonomous agents.4. Create simple agentic AI systems for real-world scenarios using Python and agent frameworks.					
Contents					
Unit I: Foundations of Agentic AI					(7 Hrs.)
Introduction to Agentic AI and differences from traditional AI systems, Intelligent agents: perception, reasoning, action, and interaction, Types of agents: reactive, deliberative, hybrid, and learning agents, Large Language Models (LLMs) as agent cores, Memory and context handling in autonomous agents, Real-world examples of Agentic AI systems.					
Unit II: Architectures and Planning					(7 Hrs.)
Agent architectures: goal-based, utility-based, and belief-desire-intention (BDI), Planning and decision-making: MDP, POMDP basics, Utility and reward functions, Workflow orchestration and chaining for agent behavior, Multi-agent interaction and coordination, Frameworks and tools (LangChain, AutoGen, CrewAI).					

Unit III: Learning, Adaptation & Control	(7 Hrs.)
Agent learning strategies and feedback loops, Self-improving agents and reflection, Integration with APIs, external knowledge, and sensors, Adaptivity, robustness, and error handling, Prompt engineering for agent control, Evaluation metrics for autonomous agents.	
Unit IV: Applications and Future Trends	(7 Hrs.)
Industrial and research applications of agentic AI, Use cases: autonomous coding agents, scientific discovery, business process automation, Challenges and limitations, Integration with enterprise systems, Trends in agent research (AutoGPT, BabyAGI, CrewAI).	
Textbooks:	
<ol style="list-style-type: none"> 1. Matt R. Cole, Building Agentic AI Systems, Packt Publishing, 1st Edition, 2024. 2. Trevor Grant, AI Agents in Action, Manning Publications, 1st Edition, 2024. 3. Deepak Agarwal, Agentic Artificial Intelligence: Harnessing AI Agents to Reinvent Business, Work and Life, World Scientific Publishing, 1st Edition, 2024. 	
Reference Books:	
<ol style="list-style-type: none"> 1. Virgílio A. F. Almeida and Diogo Cortiz, Responsible Artificial Intelligence: Challenges for Sustainable Management, Springer, 1st Edition, 2023. 2. Alexandra Sunderland, Responsible AI, Kogan Page, 1st Edition, 2024. 3. Ankur Taly, Introduction to Responsible AI: Implement Ethical AI Using Python, Springer, 1st Edition, 2023. 4. Mark Coeckelbergh, Thomas Wischmeyer (Eds.), The Cambridge Handbook of Responsible Artificial Intelligence: Interdisciplinary Perspectives, Cambridge University Press, 1st Edition, 2023. 	
List of Courses:	
<ol style="list-style-type: none"> 1. Building AI Agents with OpenAI and LangChain (DeepLearning.AI – Coursera), Instructor: Andrew Ng & LangChain Team, Focus: Fundamentals of agentic AI, tool use, memory, planning, orchestration, and real-world applications. 2. Autonomous Agents with LLMs (DeepLearning.AI – short course), Instructor: Harrison Chase (Lang Chain Founder), Focus: Hands-on design of intelligent autonomous agents using LLMs and APIs. 	
List of Course Projects:	
<ol style="list-style-type: none"> 1. Personal Study Planner Agent- An autonomous agent that plans daily study schedules based on syllabus, deadlines, and student preferences. 2. Smart FAQ Answering Agent for College Website - Agent answers student queries by reasoning over college rules, notices, and FAQs. 3. Autonomous News Summarizer Agent- Agent fetches daily news, summarizes, categorizes, and highlights key trends. 4. Smart Route Recommendation Agent - Agent chooses optimal travel route considering time, cost, and traffic. 5. Hostel Room Allocation Agent - Agent assigns rooms by maximizing student satisfaction (preferences, distance, amenities,etc.). 	

6. Multi-Agent Task Planner - Planner–Executor–Reviewer agents collaboratively complete tasks.
7. Self-Improving Answer Writing Agent - Agent generates answers, self-evaluates, and refines responses.
8. API-Integrated Weather Decision Agent - Agent suggests actions (travel, clothing etc.) based on live weather data.
9. Research Paper Assistant Agent - Agent searches, summarizes, and extracts insights from research papers.
10. Autonomous Coding Agent - Agent writes, tests, and fixes simple code tasks autonomously.
11. Enterprise Report Generation System - Role-based agents generate business reports collaboratively.
12. AI Project Management Agent - Agent breaks tasks, assigns deadlines, and tracks progress.

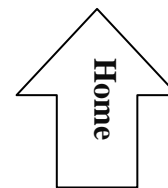


Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Information Technology, Pune-48

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

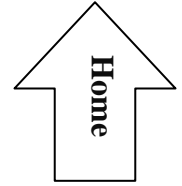
Department of Computer Science and Engineering-Data Science



MDM32236: Risk Analysis and Management with standards

Teaching Scheme	Examination Scheme					
Credits: 03	HA	SCE	CIE	OR	TW	Total
Lecture (L): 02 hrs / week						
Tutorial (P): 01 hrs / week	10	20	10	40	20	100
Prerequisite: Manufacturing Processes, Manufacturing Technology, Metrology and Quality Control, and Material Science, Quality Standards in Practice , Statistical Methods in Six Sigma						
Course Objectives:						
<ol style="list-style-type: none"> 1. Explain fundamental principles of risk, uncertainty, and their role in engineering decision-making. 2. Provide tools for identifying, analyzing, evaluating, and mitigating risks. 3. Introduce qualitative and quantitative approaches to risk analysis. 4. Develop understanding of international risk management frameworks (ISO 31000, IEC 31010). 5. Enable students to prepare a simple Risk Management Plan (RMP) through case-based learning. 						
Course Outcomes:						
After completion of the course, student will be able to:						
<ol style="list-style-type: none"> 1. Understand and classify various types of risks and their sources. 2. Identify and assess risks using structured qualitative and quantitative methods. 3. Develop appropriate risk mitigation and control strategies. 4. Integrate risk management principles in engineering systems or project 						
Contents						
Unit-I Fundamentals of Risk and Risk Management						(6Hrs.)
Concept of risk, hazard, uncertainty, exposure, consequence, and likelihood. Types of risks: operational, project, environmental, safety, financial, technological. Risk perception and risk appetite. The risk management process: identification, analysis, evaluation, treatment, monitoring, review. International standards and frameworks (ISO 31000, IEC 31010).						
Unit II – Risk Identification and Qualitative Assessment						(6Hrs.)
Techniques: brainstorming, checklist, HAZOP, SWOT, and Failure Mode and Effects Analysis (FMEA).Development of a risk register and risk breakdown structure (RBS).Qualitative tools: probability–impact matrix, risk ranking, heat maps.						

UnitIII –Quantitative Risk Analysis and Evaluation	(6Hrs.)
Probability theory basics and uncertainty quantification. Quantitative analysis techniques: Fault Tree Analysis (FTA), Event Tree Analysis (ETA), Sensitivity analysis, Monte Carlo simulation overview. Consequence and likelihood modeling. Risk evaluation: comparing options using cost-benefit and expected value analysis. Decision-making under uncertainty.	
Unit IV – Risk Mitigation, Treatment, and Emerging Issues	(6Hrs.)
Risk response strategies: avoid, accept, transfer, and mitigate. Design of mitigation and contingency plans. Monitoring, review, and feedback systems. Integration of risk management with project life cycle and quality management (ISO 9001 risk-based thinking).Emerging areas: cyber risk, sustainability and environmental risk, reliability-based design, and resilience engineering.	
A) List of experiments	
<ol style="list-style-type: none"> 1 .Case study: identifying risks in an engineering or project scenario. 2. Example: risk identification in mechanical/manufacturing or civil project. 3. Solving numerical examples on probability and quantitative risk. 4. Preparing a risk register and risk matrix. 5. Performing FMEA and FTA exercises. 6. Case study analysis (industrial/construction/software project). 7. Mini-project: Develop a basic Risk Management Plan for an engineering system. 8. Preparation of a Risk Management Plan (RMP) as a mini project. 	
B) Students Presentations	
Textbooks:	
<ol style="list-style-type: none"> 1. Varde, P. V., & Pecht, M. (2018). Risk-Based Engineering: Reliable Asset Management. Springer. 2. Rejda, G. E. (2017). Principles of Risk Management and Insurance. Pearson. 3. Kerzner, H. (2022). Project Management: A Systems Approach to Planning, Scheduling , and Controlling. Wiley. 	
Reference Books:	
<ol style="list-style-type: none"> 1.ISO 31000: Risk Management – Guidelines (latest edition). 2.PMBOK Guide, Project Management Institute (PMI). 3.NPTEL: Risk and Reliability of Engineering Systems (IIT Madras / BARC). 4.Kaplan, S., & Garrick, B. J. (1981). On the Quantitative Definition of Risk, Risk Analysis, 1(1), 11–27. 	



CD32234A: DevOps

Teaching Scheme	Examination Scheme				
Credits: 04	CIE	SCE	ESE	TW	Total
Lecture (L): 03 hrs / week					
Tutorial (T): 01hr / week	20	20	40	20	100
Prerequisites: Basics of AWS, Services of AWS					
Course Objectives: <ol style="list-style-type: none"> To understand Linux file permission models including ownership, access levels, and security policies. To understand the fundamental concepts, principles, and benefits of Continuous Integration in modern software development. To explore build automation and dependency management using Maven Project Object Model (POM). To analyze system monitoring concepts and apply them using Nagios for continuous monitoring. To understand and deploy applications using Helm charts and integrate with cloud components like AWS Load Balancer and ECR. To apply monitoring and logging techniques using metrics, SLOs/SLIs, and tools like Kibana for reliability and observability 					
Course Outcomes: After completion of the course, students will be able to: <ol style="list-style-type: none"> Explain the evolution, workflow, and technical challenges of DevOps and demonstrate Linux command-line operations. Configure and manage Continuous Integration pipelines using Jenkins for automated builds. Develop and manage Maven-based projects with dependencies and lifecycle management. Implement continuous monitoring of systems and services using Nagios plugins and architecture. Deploy containerized applications using Helm and integrate with AWS ECR and Load Balancer controllers. Evaluate system performance and reliability using SLOs, SLIs, and visualize logs using Kibana dashboards 					
Contents					
Unit I: Introduction to DevOps and Linux					(6 Hrs.)
Evolution of DevOps , Pipeline, trend of DevOps, DevOps Technical Challenges, History of Linux, Feature and Advantages of Linux, file system, Linux commands and its use, Access modes/Permissions. Case Studies: Configure server by providing security using CLI.					
Unit II: Continuous Integration					(6 Hrs.)
Essentials of Continuous Integration, Jenkins and its architecture, Jenkins tool Management, Installing Jenkins, User management in Jenkins Case Studies: Design an efficient application using Jenkin as a CI tool					

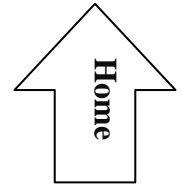
Unit III: Project Object Model (POM)	(6 Hrs.)
<p>Overview of Maven, Maven project structure, Maven plugins , fundamental unit of work in Maven project, Maven build lifecycle , adding external dependencies to maven pom.xml ,Maven build and test project.</p> <p>Case Study: Implement a structured and automated approach to building, testing, and deploying projects using Maven.</p>	
Unit IV: Continuous Monitoring	(6 Hrs.)
<p>Introduction to Nagios, Nagios Architecture , Monitoring Services in Nagios, Nagios Plugins , Monitoring System info using Nagios Plugins.</p> <p>Case Study: Set up a robust monitoring framework for infrastructure</p>	
Unit V: Deployment of Application	(6 Hrs.)
<p>Helm and its characteristic ,package manager, integration with AWS Load Balancer Controller & AWS Elastic Load Balancing, ECR repository</p> <p>Case Study :Develop Kubernetes application providing packaging, templating, and release management capabilities.</p>	
Unit VI: Monitoring & Logging	(6 Hrs.)
<p>Metrics, logs, and traces for end-to-end visibility of the system, Master Service Level Objectives (SLO's) and Service Level Indicators (SLIs) for reliability,Metric collection, storage, and alerting with a time-series database, Kibana for end-to-end log management.</p> <p>Case Study: Develop an application ensures high system reliability, better operational insights, and improved end-user experience</p>	
Textbooks:	
<ol style="list-style-type: none"> 1. Jim Bird, DevSecOps: A Leader's Guide First edition ISBN-10. 1781335028 ,June 2016. 2. Nicole Forsgren, Accelerate: Building and Scaling High Performing Technology Organizations First edition ISBN-13. 978-1942788331 March 2018. 	
Reference Books:	
<ol style="list-style-type: none"> 1. Gene Kim, Jez Humble, "The DevOps Handbook", First edition ISBN: 1942788002 Oct 2016 2. Creane , "Kubernetes Security and Observability", First edition ISBN-10: 1098107101 Oct 21 3. Kief Morris, "Infrastructure as Code",Third edition ISBN-13: 978-1491924358 March 2025. 	
List of Courses:	
<ol style="list-style-type: none"> 1. Open-DevSecOps Course (GitHub): A free, interactive course covering Git, CI/CD, and security checks. Open-devsecops https://open-devsecops.github.io/docs/course/?utm_source=chatgpt.com 2. Infosec Institute – DevSecOps Overview: Good for understanding models, principles, and SDLC. Infosec Institute https://www.infosecinstitute.com/skills/courses/devsecops-overview/?utm_source=chatgpt.com 3. Infosec Institute – DevSecOps Learning Path: A structured online path with modules on 	

SAST, SCA, container security, etc. Infosec Institute

https://www.infosecinstitute.com/skills/learning-paths/devsecops/?utm_source=chatgpt.com

List of Assignments:

1. Demonstrate Linux file permissions and command-line utilities.
2. Install Jenkins, configure a basic pipeline, and automate a simple build process.
3. Create a Maven project, define dependencies in pom.xml, and perform a build-test cycle.
4. Install and configure Nagios to monitor a local service (e.g., web server). Document the plugin usage and monitoring results.
5. Deploy a sample application using Helm charts integrated with AWS Load Balancer Controller and push images to ECR.
6. Collect metrics and logs, define SLOs/SLIs, and visualize system performance using Kibana dashboards.

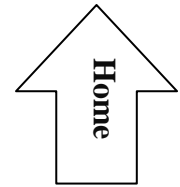


CD32234B: Natural Language Processing

Teaching Scheme	Examination Scheme				
Credits: 04	CIE	SCE	ESE	TW	Total
Lecture (L) 03 hrs / week					
Tutorial (T) 01 hr / week	20	20	40	20	100
Prerequisites: Python, Data Structure and Machine learning concepts.					
<p>Course Objectives:</p> <ol style="list-style-type: none"> To study some of the problems and solutions of NLP and their relation to linguistics and statistics. To understand and carry out proper experimental methodology for training and evaluating empirical NLP systems To estimate parameters using supervised and unsupervised training methods. To design, implement, and analyze NLP algorithms. Able to design different language modeling Techniques. 					
<p>Course Outcomes:</p> <p>After completion of the course, student will be able to:</p> <ol style="list-style-type: none"> Identify and explain key problems in Natural Language Processing (NLP) and evaluate various linguistic and statistical approaches used to solve them. Implement parameter estimation techniques using both supervised and unsupervised learning methods in NLP tasks. Design and develop NLP algorithms for core language tasks such as tokenization, parsing, named entity recognition, and sentiment analysis. Develop and evaluate different language modeling techniques (e.g., n-gram, neural models) to perform tasks such as text generation or speech recognition. 					
Contents					
Unit I: Natural Language processing (NLP)					(6 Hrs.)
Introduction, Applications or Use cases of NLP, Components of NLP, Steps in NLP, Finding the Structure of Words: Words and Their Components, Lexemes, Morphemes, Morphology, Problems in morphological processing, Typology, Morphological Typology, Natural Language Processing with python NLTK package: Word Tokenization, Sentence Tokenization, Filtering Stop words, Stemming, Tagging Parts of Speech, Lemmatization, Chunking, Chinking, Named Entity Recognition.					
Unit II: Syntax Analysis and Parsing					(6 Hrs.)
<p>Syntax Analysis: Parsing Natural Language, Tree banks: A Data-Driven Approach to Syntax, Syntax Analysis using Dependency Graph, Syntax Analysis using Phrase Structure Trees, Parsing Algorithms: Shift Reduce Parsing, Hyper Graphs and Chart Parsing (CYK Parsing), Models for ambiguity Resolution in Parsing: Probabilistic Context Free Grammar, Generative Models, Discriminative models for Parsing.</p>					

Unit III: Language Modeling	(6 Hrs.)
Language Modeling: Introduction, N-Gram Models, Language Model Evaluation, Parameter Estimation, Language Model Adaptation, Bag-of-Words model, Word Embeddings (Word2Vec, GloVe) BERT Modeling, Types of Language Models, Language-Specific Modeling Problems.	
Unit IV: Semantic Analysis	(6 Hrs.)
Semantic Analysis: Introduction, Semantic Interpretation, System Paradigms, Word Sense Systems, Software, Predicate-Argument Structure , Meaning Representation Systems, Software. Discourse Processing: Cohesion, Reference Resolution, Discourse Cohesion and Structure.	
Unit V: Sentiment Analysis and Text Classification	(6 Hrs.)
Understanding sentiment analysis tasks: Sentiment lexicons and rule-based approaches, Machine learning approaches for sentiment analysis, Text Classification: Supervised learning for text classification, Feature extraction techniques, Classification algorithms (Naive Bayes, Support Vector Machines, Neural Networks)	
Unit VI: Topic Modeling and Applications	(6 Hrs.)
Topic Modeling: LSA (Latent Semantic Analysis), Latent Dirichlet Allocation (LDA), Non-negative Matrix Factorization (NMF), Applications of topic modeling, Applications in machine translation, summarization, and question answering, Overview of real-world NLP applications, Ethical Considerations in NLP, Future trends and directions in NLP research and development	
Textbooks:	
<ol style="list-style-type: none"> 1. Multilingual Natural Language Processing Applications: From Theory to Practice–Daniel M. Bikel and Imed Zitouni, IBM Press Publication First Edition. ISBN-10: 9788131791370 2. Speech and Language Processing-Daniel Jurafsky & James H Martin, Pearson Publications. 2nd Edition. ISBN-10: 9789332518414 	
Reference Books:	
<ol style="list-style-type: none"> 1. Introduction to Large Language Models Generative AI by Tanmoy Chakraborty Wiley Publication, ISBN-10: 936386474X 2. Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning by Delip Rao, Brian McMahan, O'Reilly Publication. Greyscale Indian Edition ISBN 10: 935213768X 	
List of Courses:	
<ol style="list-style-type: none"> 1. https://onlinecourses.nptel.ac.in/noc25_cs51/preview 2. https://www.coursera.org/specializations/natural-language-processing 	
List of Tutorials:	
Implement any 7 of them.	
<ol style="list-style-type: none"> 1. Study the major applications and use cases of Natural Language Processing (NLP) in real-world systems and explain the components and steps involved in an NLP pipeline with suitable examples. 2. Analyze the structure of words in natural language by identifying lexemes, morphemes, and morphological processes, and discuss the challenges involved in morphological processing. 	

3. Implement basic text preprocessing techniques using Python NLTK, including word tokenization, sentence tokenization, stop-word removal, stemming, lemmatization, and Part-of-Speech tagging.
4. Perform syntactic analysis of sentences by constructing dependency graphs and phrase structure trees, and explain the differences between dependency-based and constituency-based parsing.
5. Implement parsing algorithms such as Shift-Reduce Parsing or CYK Parsing using Context-Free Grammar (CFG) and analyze syntactic ambiguity in natural language sentences.
6. Build and evaluate language models using unigram, bigram, and trigram approaches, and analyze their effectiveness in predicting word sequences.
7. Compare Bag-of-Words, TF-IDF, and word embedding models (Word2Vec, GloVe, BERT) with respect to representation capability, context handling, and applications.
8. Study semantic parsing techniques by implementing word sense disambiguation using WordNet and analyzing semantic similarity between words or sentences.
9. Design and implement a sentiment analysis system using rule-based and machine learning approaches (Naive Bayes or SVM), and evaluate its performance on a text dataset.
10. Apply topic modeling techniques such as LSA, LDA, or NMF on a document collection and analyze their applications, ethical considerations, and future trends in NLP.



CEI32234H: Advanced Mainframe Technologies

Teaching Scheme	Examination Scheme				
Credits: 03	CIE	SCE	ESE(CP)	OR	Total
Lecture (L): 02 hrs / week					
Practical (P): 02 hrs / week	20	20	40	20	100
Prerequisites: Basic Programming Knowledge (preferably Python), Fundamentals of Artificial Intelligence , Data Structures and Algorithms.					
Course Objectives:					
<ol style="list-style-type: none"> To introduce ISPF, JCL & GDG of IBM Z Mainframe Technology. To learn integration of z/OS with Java, Python, and Containerization on IBM Z. To learn z/OS REXX, COBOL Programming with DB2 Integration. To introduce the IBM Z16, Z17 Telum Processor. 					
Course Course Outcomes:					
After completion of the course, students will be able to					
<ol style="list-style-type: none"> Understanding of IDPF, JCL & GDG of IBM Z Mainframe Technology. Design and develop a program for Java, Python & Containerization on IBM Z using z/OS. Design and develop a program for REXX, COBOL with DB2 integration using z/OS. Understanding of IBM Z16, Z17 Telum Processor 					
Contents					
Unit I: Overview of ISPF, JCL & GDG					(7 Hrs.)
ISPF: Customizing ISPF Panels, Advanced DSLIST Operations, Batch Processing with ISPF JCL: Building Modular JCL, Error Handling and Recovery, JCL for Automation, Practice Various utilities of JCL Transferring Files Between Mainframe and Systems: Automating File Transfers (Using JCL), Data Format Conversion GDG : Overview of GDG, Defining GDG Base, Creating and Accessing GDG Generations, Operations on GDG, Advanced GDG Applications z/OS Unix System Services: Introduction to z/OS USS, Accessing and Navigating z/OS USS, File Systems in z/OS USS.					
#Exemplar/Case Studies	Designing a real-world scenario using mainframe tools for efficient enterprise operations.				
*Mapping of Course Outcomes for Unit I	CO1				
Unit II: Architectures and Planning					(6 Hrs.)
Overview of z/OS, Java on z/OS & its use cases with examples, Python on z/OS & its use cases with examples. Containerization on IBM Z: Key Technologies Supporting Containers on IBM Z, z/OS Container Extensions (zCX), Working architecture of z/OS Container Extensions(zCX), Key Features of IBM Z Containers, Benefits of Containers on IBM Z, Typical Use Cases for IBM Z Containers.					

#Exemplar/Case Studies	Developing applications using technology to modernize workloads, enhance efficiency, and support hybrid cloud deployments.
*Mapping of Course Outcomes for Unit II	CO2

Unit III: COBOL & REXX Programming with DB2 Integration	(6 Hrs)
--	-----------------

COBOL Fundamentals and Core Concepts, Advanced COBOL: Arithmetic, Functions, Conditional Processing, and Loops, Character and File Handling in COBOL, Working with Tables in COBOL, Developing CICS-COBOL Programs, Basic Program in REXX, Methods of Executing REXX Programs.

Role of DB2 Integration on z/OS, Advantages of DB2 in mainframe environments, accessing a DB2 database from a Cobol / Java program, DB2 for z/OS architecture (Subsystem, DB2 Catalog, Buffer Pools, and more), DB2 SQL support and relational databases, DB2 High Availability Disaster Recovery (HADR), DB2 Tools and Utilities, DB2 Command Line Processor (CLP), IBM Data Studio for DB2, DB2 Backup and Recovery processes

#Exemplar/Case Studies	Developing applications using programming tools to solve real-world challenges in data processing, automation, and modernization.
------------------------	---

*Mapping of Course Outcomes for Unit III	CO3
--	-----

Unit IV: IBM Z16, Z17 Telum Processor	(6 Hrs.)
--	-----------------

Introduction to IBM Z16 and Z17 Telum Systems, Comparison of Z15, Z16, and Z17 architectures, Telum's role in Artificial Intelligence (AI) and Machine Learning (ML) workloads.

Key Features Comparison: Performance, Scalability, Security, and Availability, Hardware and Software Integration.

#Exemplar/Case Studies	Developing applications for real-time analysis using Telum Processor's with AI inferencing capabilities enabled.
------------------------	--

*Mapping of Course Outcomes for Unit IV	CO4
---	-----

Learning Resources

Textbooks:

1. <https://www.ibm.com/training/>
2. <https://learn.ibm.com/>
3. <https://www.ibm.com/z/education>
4. <https://www.redbooks.ibm.com/redbooks.nsf/redbookabstracts/crse0304.html>

Reference Books:

e-Resources: :

1. https://ibmzxplore.influitive.com/users/sign_in
2. <https://www.ibm.com/z/education>

MOOC Courses:

1. <https://www.ibm.com/training/>
2. <https://learn.ibm.com/>
3. <https://www.ibm.com/z/education>
4. <https://www.redbooks.ibm.com/redbooks.nsf/redbookabstracts/crse0304.html>

List of Assignments:

Assignment 1: Enterprise Data Encoding

Weightage: 25%

Objective:

Students are required to create 10 data records following a specific format that will be provided by the BMC SMEs during the session. Basic training will also be conducted to guide students through the process.

Details:

- The records must include Binary Coded Decimal (BCD) and Packed Decimal formats, which are commonly used in Mainframe systems across various enterprises.
- Once the records are created, students must download the files in binary format and upload them to the designated Google Classroom.

Instructions for Faculty:

- Faculty members are responsible for collecting all submitted files and uploading them to the Mainframe system.
- The BMC SMEs will supply the necessary resources and guidance to facilitate automated evaluation of the assignments using COBOL, REXX, and JCL.

Assignment 2: JCL for Using IBM Utilities

Weightage: 15%

Objective:

BMC SMEs will guide students on how to generate JCL to use IBM/Non-IBM Programs, verify and resolve JCL errors during the training session.

Details:

- After the session, students will be providing 5 problem statements to generate JCL for using different IBM Utilities.
- Students are required to generate, analyze, correct, and verify each JCL.
- Once finalized, the JCLs and Job Logs must be downloaded in text format and uploaded to the designated Google Classroom.

Instructions for Faculty:

- Faculty members are responsible for collecting all submitted files and uploading them to the Mainframe system.
- The BMC SMEs will provide the necessary resources and guidance to enable automated evaluation using REXX and JCL.

Assignment 3: COBOL Program Development

Weightage: 35%

Objective:

This assignment is based on the data created in Assignment 1.

Details:

- Students are required to develop a COBOL program based on the data records created in Assignment 1.
- Once the program is complete, students must upload the source code and Output Job Log to the designated Google Classroom.

Instructions for Faculty:

- Faculty members are responsible for collecting all submitted files and uploading them to the Mainframe system.
- The BMC SMEs will provide the necessary resources and guidance to enable automated evaluation using COBOL, REXX, and JCL.

Assignment 4: REXX Program Development

Weightage: 25%

Objective:

This assignment is based on the data created in Assignment 1.

Details:

- Students are required to develop a REXX program based on the data records created in Assignment 1.
- REXX Should be submitted using JCL and not Interactive.
- Once the program is complete, students must upload the source code and Output Job Log to the designated Google Classroom.

Instructions for Faculty:

- Faculty members are responsible for collecting all submitted files and uploading them to the Mainframe system.
- The BMC SMEs will provide the necessary resources and guidance to enable automated evaluation using REXX and JCL.

Assignment 5: A Mini Project based on the concepts learned in this course.