SCHEME OF INSTRUCTION

Common to BE (CSE (AI&ML), CSE(AI), AI & ML)

SEMESTER- V

S. No.	Course Code	Course Title	Scheme of Instruction			Contact Hours/ Week	Scheme of Examination		Credits
			L	T	P		CIE	SEE	
THEORY									
1.	HS IO4 EG	Effective Technical Communication in English	3	0	0	3	30	70	3
2.	PC 501 AI	Design and Analysis of Algorithms	3	0	0	3	30	70	3
3.	PC 401 CS	Operating Systems	3	0	0	3	30	70	3
4.	PC 502 AI	Database Management Systems	3	0	0	3	30	70	3
5.	PC 503 AI	Automata Theory & Compiler Design	3	0	0	3	30	70	3
6.	PC 504 AI	Machine Learning	3	0	0	3	30	70	3
7.	**Professional Elective-II**		3	0	0	3	30	70	3
	PE 521 AI	Web Programming							
	PE 522 AI	Software Testing							
	PE 523 AI	Computer Graphics							
	PE 524 AI	Computer Vision							
Practicals									
8.	PC 551 AI	Database Management Systems Lab	0	0	2	2	25	50	1
9.	PC 452 CS	Operating Systems Lab	0	0	2	2	25	50	1
10.	PC 552 AI	Machine Learning Lab	0	0	2	2	25	50	1
Total			**21**	**0**	**6**	**27**	**285**	**640**	**24**

(KMEC, NAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

HS 104 EG	**EFFECTIVE TECHNICAL COMMUNICATION IN ENGLISH**					
Prerequisites			L	T	P	C
			3	0	0	3
Evaluation	CIE	30 Marks	SEE		70 Marks	

Course Objectives:	
1	Features of technical communication
2	Types of professional correspondence
3	Techniques of Report Writing
4	Basics of manual writing
5	Aspects of data transfer and Presentations.

Course Outcomes: On completion of this course, the student will be able to	
CO1	Handle technical communication effectively
CO2	Use different types of professional correspondence.
CO3	Use various techniques of report writing
CO4	Acquire adequate skills of manual writing
CO5	Enhance their skills of information transfer and presentation

UNIT – I
Definition and Features of Technical communication: Definition and features of technical communication (precision, relevance, format, style, use of visual aids), Differences between general writing and technical writing, Types of technical communication (oral and written)

UNIT – II
Technical Writing-I (Official correspondence): Emails, IOM, Business letters, Business Proposals

UNIT – III
Technical writing-Il (Reports): Project report, Feasibility report, Progress report, Evaluation report.

UNIT – IV
Technical writing- III (Manuals): Types of manuals, User manual, Product manual, Operations manual.

UNIT – V
Information Transfer and Presentations: Non-verbal (bar diagram, flow chart, pie chart, tree diagram) to verbal (writing), Verbal (written) to non-verbal, Important aspects of oral and visual presentations.

Suggested Reading:

1	Raman, Meenakshi & sharma, sangeeta. (2015). Technical communication: Principles and Practice, 3rd Edition, New Delhi
2	Rizvi, Ashraf, M. (20L7). Effective Technical Communication (2nd ed.). New Delhi, Tata McGraw Hill Education Security.
3	Sharma, R.C., & Mohan, Krishna. (2O17). Business Correspondence and Report Writing: A Practical Approach to Business & Technical Communication (4th ed.), New Delhi, Tata McGraw Hill Education.
4	Tyagi, Kavita & Misra, Padma.(2011). Advanced Technical Communication. New Delhi, PHI Learning.
5	Jungk, Dale.(20O4). Applied Writing for Technicians. New York, McGraw-Hill Higher Education.

(KMEC, NOAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC501 AI	**DESIGN AND ANALYSIS OF ALGORITHMS**					
Prerequisites	Data Structures		L	T	P	C
			3	1	0	3
Evaluation	CIE	30 Marks	SEE		70 Marks	

Course Objectives	
1	To review elementary data structures , order notation and algorithm analysis
2	To learn algorithm design strategies such as Divide-and-Conquer, greedy method, dynamic programming, backtracking and branch & bound technique
3	To understand the concepts of NP-hard and NP-complete
4	Apply important algorithmic design paradigms and methods of analysis.
5	Synthesize efficient algorithms in common engineering design situations.

Course Outcomes: On completion of this course, the student will be able to	
CO1	Understand algorithm complexity using Big-O notation and solve recurrence relations with Master's Theorem.
CO2	Design algorithms using Divide-and-Conquer and Brute Force methods.
CO3	Use Greedy and Dynamic Programming to solve problems like Knapsack, Shortest Path, and TSP.
CO4	Solve complex problems using Backtracking and Branch-and-Bound techniques.
CO5	Identify problem types like P, NP, NP-Complete, NP-Hard and learn the basics of parallel computing.

UNIT – I
Introduction & Elementary Data Structures: Introduction, Fundamentals of algorithm(Line Count, Operation Count), Analysis of algorithms(Best, Average, Worst case), Asymptotic Notations(O,Ω,Θ) Recursive Algorithms, Analysis using Recurrence Relations, Master's Theorem. Review of elementary data structures–Graphs: BFS, DFS, Bi-Connected Components. Sets: representation, UNION, FIND operations

UNIT – II
Divide-and-Conquer Method: The general method, Binary search, Finding maximum and minimum, Merge sort, Quick sort. Brute Force: Knapsack, Traveling salesman problem, Convex-Hull

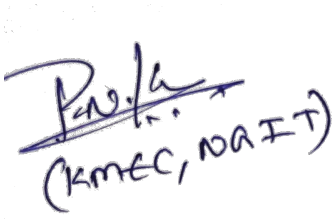
UNIT – III
Greedy Method: Knapsack problem, Minimum spanning trees, Single source shortest path, Job sequencing with deadlines, Optimal storage on tapes, Optimal merge pattern Dynamic programming method: All pairs shortest paths, Optimal binary search tress, 0/1 Knapsack problem, Reliability design, Traveling salesman problem,

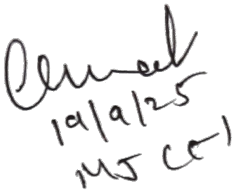
UNIT – IV
Back tracking: N-queens problem, Graph coloring, Hamiltonian cycles Branch-and-bound: FIFO & LC branch and Bound methods, 0/1 Knapsack problem, Traveling salesperson

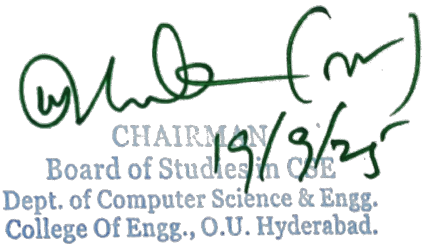
UNIT – V
Computational Complexity: Non Deterministic algorithms, The classes: P, NP, NP Complete, NP Hard, Satisfiability problem, Proofs for NP Complete Problems: Clique, Vertex Cover. Parallel Algorithms: Introduction, models for parallel computing, computing with complete binary tree.

Suggested Reading:

1	Horowitz E, Sahni S, Fundamentals of Computer Algorithms, 2nd Edition, Universities Press, 2007.
2	Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L. (1990). Introduction to Algorithms, 4th Edition, MIT Press and McGraw-Hill, April 2022.
3	Michael T. Goodrich, Roberto Tamassia, Algorithm Design: Foundations, Analysis and InternetExamples, John Wiley & Sons,2002.







PC 401CS	OPERATING SYSTEMS					
Prerequisites	**Programming in C and Data Structures**		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To learn the basic concepts, types, and structure of operating systems.
2	To explain process management, threads, and CPU scheduling.
3	To understand synchronization and deadlock handling.
4	To learn memory and virtual memory management techniques.
5	To gain knowledge on file systems, I/O operations, and disk scheduling.

Course Outcomes	
On completion of this course, the student will be able to	
CO1	Understand the basic concepts, types, services, and structure of operating systems.
CO2	Explain processes, threads, and CPU scheduling methods.
CO3	Apply synchronization techniques and solve deadlock issues.
CO4	Apply memory and virtual memory management strategies.
CO5	Evaluate file systems, I/O operations, and disk scheduling methods

UNIT – I

Introduction: Concept of Operating Systems, Generations of Operating Systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS – Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.

UNIT – II

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching.
Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,
Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers,Scheduling Criteria, Scheduling algorithms, multiprocessor scheduling.

UNIT – III

Process Synchronization: Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Peterson‘s Solution, classical problems of synchronization: The Bounded buffer problem, Producer\Consumer Problem, reader‘s & writer problem, Dining philosopher‘s problem. Semaphores, Event Counters, Monitors, Message Passing,
Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Methods for Handling: Deadlocks: Deadlock prevention, Deadlock Avoidance: Banker‘s algorithm, Deadlock detection and Recovery.

UNIT – IV

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation, fragmentation, and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, structure of page table, Protection and sharing, Disadvantages of paging.
Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms, Trashing.

UNIT – V

I/O Hardware: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods, Free-space management, directory implementation, efficiency, and performance. Secondary-Storage Structure: Disk structure, Disk scheduling algorithms, Disk Management, RAID structure.

Suggested Reading:

1	Avi Silberschatz, Peter Galvin, Greg Gagne, *Operating System Concepts Essentials*, 10th Edition, Wiley Asia Student Edition, 2018.
2	William Stallings, *Operating Systems: Internals and Design Principles*, 9th Edition, Pearson Education, 2018
3	Andrew S. Tanenbaum, Herbert Bos, *Modern Operating Systems*, 4th Edition, Pearson Education, 2015.

(KMEC, NDAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC 502 AI	DATABASE MANAGEMENT SYSTEMS					
Prerequisites	**Data Structures**		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To understand the basics of DBMS and learn how to design databases using ER diagrams.
2	To learn how data is stored in tables and apply rules to keep data correct and complete.
3	To write SQL queries to retrieve and manage data, and organize tables using normalization.
4	To understand how transactions work and how to manage multiple users safely.
5	To use triggers and stored procedures, and learn basic tasks of a database administrator.

Course Outcomes On completion of this course, the student will be able to	
CO1	Recall DBMS basics and draw ER diagrams.
CO2	Understand relational tables and apply data rules.
CO3	Write SQL queries and modify data.
CO4	Analyze transactions and concurrency issues.
CO5	Create triggers, procedures, and manage users.

UNIT – I

Database System Applications: File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS, Advantages of DBMS.

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model.

UNIT – II

Introduction to the Relational Model: Data Definition Language, Integrity constraint over relations, Types of Integrity Constraints: Domain Constraint - String, Character, Integer, Date, Entity Integrity Constraint, Primary Key, Referential Integrity Constraint - Foreign Key, Other Key Constraints – NULL, NOT NULL, CHECK, etc. Querying relational data, logical database design, introduction to views, destroying/altering tables and views.

UNIT – III

Introduction to SQL: Select Queries, Constraints, Data Manipulation Language – Insert, Delete, Update. Form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, Co-related Queries, Aggregation operators, NULL values, complex integrity constraints in SQL.

Concept of Joins: Join, Outer Join, Left Outer Join, Right Outer Join, Self Join.

Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies. FIRST, SECOND, THIRD normal forms, BCNF, Lossless join decomposition, Multi-valued dependencies, FOURTH normal form, FIFTH normal form

UNIT – IV
Introduction to Transactions: Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation. Testing for Serializability, Lock-Based Protocols, Timestamp-Based Protocols, Validation-Based Protocols, Multiple Granularity. Recovery and Atomicity, Log-Based Recovery, Recovery with Concurrent Transactions. TCL Commands – Savepoint, Commit, Rollback

UNIT – V
Overview of Triggers, Stored Procedures: Triggers – Row-level, table-level, and active databases. Stored Procedures – IN, OUT parameters, Execution of Stored Procedure from Java. **DBA:** Introduction to DBA, Creating Users, Grant/Revoke Permissions on tables using DML Commands.

Suggested Reading:

1	Raghurama Krishnan, Johannes Gehrke, **Database Management Systems**, Tata McGraw Hill, 3rd Edition, 2003 (Units I, II, III)
2	Silberschatz, Korth, **Database System Concepts**, McGraw Hill, 7th Edition, 2019 (Units IV, V)
3	Elmasri & Navathe, **Fundamentals of Database Systems**, Pearson Education, 7th edition , 2015

(KMEC, NGIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.
19/9/25

PC503AI	AUTOMATA THEORY & COMPILER DESIGN					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	Provide information to some of the central ideas of theoretical computer science from perspective of formal languages
2	Introduce fundamental concepts of Formal languages, grammars, and types of automata
3	Introduce the major concepts of language translation and compiler design.
4	Impart the knowledge of practical skills for constructing a compiler.
5	Introduce various phases of compiler in detail with examples

Course Outcomes On completion of this course, the student will be able to	
CO1	Identify and understand the concept of abstract machines and their power to recognize the languages
CO2	Explore context free grammar for formal languages and learn basics of compiler to implement LL parsers
CO3	Implement LR parsers and understanding semantic analysis functionalities
CO4	Explore different forms of intermediate code to design and develop machine code
CO5	Understanding code optimization techniques for effective programming and generate assembly code

UNIT – I

Introduction to finite automata: The central concepts of automata theory, Structural representation of FA, Types of FA, Conversion of NFA to DFA,NFA with epsilon to NFA without epsilon conversion.
Regular Expression: Introduction to Regular language, Algebraic laws for regular expressions, Conversion of FA to RE, Conversion of RE to FA, Pumping lemma for Regular language

UNIT – II

Grammar: Definition of Grammar, Types of grammars, Derivation, Types of derivations, Derivation tree, Ambiguity, Left recursion elimination.
Push Down Automata: Definition of PDA, Structural representation of PDA, Construction of PDA

UNIT – III

Turing Machine: Definition of TM, Structural representation of TM, Construction of TM.
Compiler: Definition of Compiler, Phases of Compiler, Lexical Analysis, Input Buffering.
Syntax Analysis: Types of Parsing, Recursive Descent parsing, Predictive Parsing, Bottom-up Parsing SLR, CLR, and LALR

UNIT – IV

Semantic Analysis: Introduction to Syntax Directed Definition, Syntax Directed Translation, Attributes, Types of Attributes, Bottom-up evaluation of attributes.
Intermediate code generation: Types of Intermediate codes, Types of Three address codes

UNIT – V

Runtime Environment: Storages organization, Storage allocation strategies: Static, Stack, Heap allocations, Activation Record. **Code Optimization:** Introduction, Principal sources of optimization, basic block, partition algorithm of basic block, flow graph, techniques of loop optimizations. **Code generation:** Issues in code generation, DAG, Simple code generator

Suggested Reading:

1	Compilers: Principles, Techniques and Tools - Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, 2nd Edition, Pearson.
2	Compiler Construction - Principles and Practice, Kenneth C Louden, Cengage Learning.
3	Modern compiler implementation in C - Andrew W Appel, Revised Edition, Cambridge University Press
4	The Theory and Practice of Compiler writing - J. P. Tremblay and P. G. Sorenson, TMH.

(KMEC, NRIIT)

19/9/25
MJ CE-I

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.
19/9/25

PC 504 AI	MACHINE LEARNING					
Prerequisites	Artificial Intelligence		L	T	P	C
			3	0	0	3
Evaluation	CIE	30 Marks	SEE		70 Marks	

Course Objectives:	
1	To learn the basics of machine learning and its main types.
2	To understand how to use probability and regression for making predictions.
3	To study the support vector machines and analyze learning with statistics.
4	To explore how to evaluate and improve models using neural networks and optimization.
5	To learn how to build models using clustering, active learning, and reinforcement learning.

Course Outcomes: On completion of this course, the student will be able to	
CO1	Understand the basic machine learning types and algorithms like perceptron and PCA.
CO2	Apply probabilistic models and regression methods for data classification.
CO3	Analyze SVMs, statistical measures, and graphical models for learning tasks.
CO4	Evaluate learning models using regularization, neural networks, and genetic algorithms.
CO5	Develop models using clustering, active learning, and reinforcement learning techniques.

UNIT – I

Introduction: Learning, Types of Machine Learning, Machine Learning Applications.
Concept learning: Introduction, Version Spaces and the Candidate Elimination Algorithm.
Linear Discriminants: Learning Linear Separators,The Perceptron Algorithm, Margins, Principal component analysis.

UNIT – II

Estimating Probabilities from Data, Bayes Rule, MLE, MAP, **Naive Bayes:** Conditional Independence, **Naive Bayes:** Why and How, Bag of Words, **Logistic Regression:** Maximizing Conditional likelihood, Linear Regression, **Kernels:** Kernalization Algorithm, Kernalizing Perceptron, **Learning with Trees:** Decision Tree Learning.

UNIT – III

Support Vector Machines: Geometric margins, Primal and Dual Forms, Kernalizing SVM **Generalization & Overfitting:** Sample Complexity, Finite Hypothesis classes, VC Dimension Based Bounds, **Some Basic Statistics:** Averages, Variance and Covariance, The Gaussian, The Bias-Variance Tradeoff . **Graphical Models:** Bayesian networks, Approximate Inference, Making Bayesian Networks.

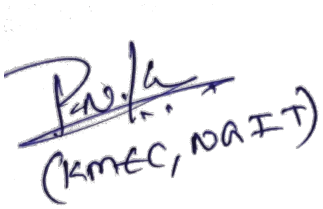
UNIT – IV

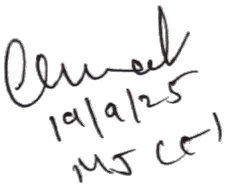
Model Selection & Regularization: Structural Risk Minimization, Regularization, k-Fold Cross Validation.
Neural Networks: Back Propagation. **Evolutionary Learning**: (Genetic Algorithm)
Perceptrons, Linear Separability, MLP: Going Forward, Backwards, Gradient Descents, Deriving Back Propagation, Back Propagation Algorithm. Different types of Activation Functions

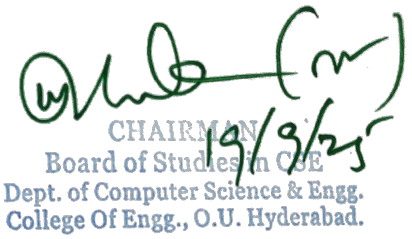
UNIT – V
Clustering: Introduction, Similarity and Distance Measures, Outliers, Hierarchical Methods, Partitional Algorithms, Clustering Large Databases, Clustering with Categorical Attributes, Comparison. **Interactive Learning:** Active Learning, Common heuristics, Sampling bias, Safe Disagreement Based Active Learning Schemes. **Reinforcement Learning:** Markov Decision Processes, Value Iteration, and Q-Learning.

Suggested Reading:

1	Tom M. Mitchell, *Machine Learning*, Mc Graw Hill, 1997
2	Chistopher Bishop, *Pattern recognition and Machine Learning*, Springer 2006.
3	Uma N. Dulhare , Khaleel Ahmad , Khairol Amali Bin Ahmad , Machine Learning and Big Data: Concepts, Algorithms, Tools and Applications, Scrivener Publishing,Wiley,2020
4	Stephen Marsland, *Machine Learning - An Algorithmic Perspective*, CRC Press, 2009







PROFESSIONAL ELECTIVE-II

PE 521 AI	WEB PROGRAMMING					
Prerequisites	**Programming Concepts**		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To learn the basics of the web and create web pages using HTML5.
2	To understand XML and use it for storing and displaying data.
3	To use JavaScript and Ajax for making interactive web pages.
4	To develop dynamic web applications using Java Servlets and JSP.
5	To build web applications using PHP and connect them to databases.

Course Outcomes On completion of this course, the student will be able to	
CO1	Understand the basics of the World Wide Web and create web pages using HTML5.
CO2	Work with XML documents, schemas, and style them using CSS and XSLT.
CO3	Use JavaScript and Ajax to create dynamic and interactive web applications.
CO4	Develop server-side programs using Java Servlets and Java Server Pages (JSP).
CO5	Build web applications using PHP and connect them to databases.

UNIT – I

Introduction to World Wide Web: Web Browsers, Web Servers, Uniform Resource Locators, HTTP. HTML5: Introduction, Links, Images, Multimedia, Lists, Tables, Creating Forms, Styling Forms.

UNIT – II

Introduction to XML: XML document structure, Document Type Definition, Namespaces, XML Schemas, Displaying raw XML documents, Displaying XML documents with CSS, XPath Basics, XSLT, XML Processors.

UNIT – III

Introduction to Java Script: Java Script and Forms Variables, Functions, Operators, Conditional Statements and Loops, Arrays DOM, Strings, Event and Event Handling, Java Script Closures. Introduction to Ajax, Pre-Ajax Java Script Communication Techniques, XML HTTP Request Object, Data Formats, Security Concerns, User Interface Design for Ajax. Introduction to Python, Objects and Methods, Flow of Control, Dynamic Web Pages.

UNIT – IV

Java Servlets: Java Servlets and CGI Programming, Benefits of Java Servlet, Life Cycle of Java Servlet, Reading data from client, HTTP Request Header, HTTP Response Header, working with Cookies, Tracking Sessions. Java Server Pages: Introduction to JSP, JSP Tags, Variables and Objects, Methods, Control Statements, Loops, Request String, User Sessions, Session Object, Cookies.

UNIT – V

Introduction to PHP: Overview of PHP, General Syntactic Characteristics, Primitives, Operations, Expressions, Control Statements, Arrays, Functions, Pattern matching, Form handling, Files, Cookies, Session Tracking. Database access through Web: Architectures for Database Access Database access with Perl - Database access with PHP-Database access with JDBC.

Suggested Reading:

1	Robert W.Sebesta, *Programming the World Wide Web*, 3rd Edition, Pearson Education, 2006.
2	Wendy Willard, *HTML5*, McGraw Hill Education, 2013
3	Thomas Powell, *The Complete Reference: Ajax*, Tata-McGraw-Hill, 2011
4	John Pollock, *Java Script*, 4th Edition, McGraw Hill Education, 2013.
5	Jim Keogh, *J2EE : The Complete Reference*, Tata-McGraw-Hill, 2002

(KMEC, NGIIT)

19/9/25
MJ CC-1

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.
19/9/25

PROFESSIONAL ELECTIVE-II

PE 522 AI	SOFTWARE TESTING					
Prerequisites			L	T	P	C
			3	0	0	3
Evaluation	CIE	30 Marks	SEE		70 Marks	

Course Objectives	
1	To Learn basic concepts, goals, and methods of software testing.
2	To Understand types and levels of testing like black-box, white-box, unit, integration, and system testing.
3	To Identify and fix bugs using different testing levels.
4	To Apply transaction flow and data flow testing techniques.
5	To Use domain, path, and logic-based testing to improve test effectiveness.

Course Outcomes: On completion of this course, the student will be able to	
CO1	Identify the various bugs and correcting them after knowing the consequences of the bug.
CO2	Use the program's control flow as a structural model, which is the corner stone of testing.
CO3	Performing functional testing using control flow and transaction flow graphs.
CO4	Apply the process of testing and various methodologies in testing for developed software.
CO5	Apply suitable testing tools, techniques and methods to enhance system effectiveness during test planning and execution process.

UNIT – I

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs.

Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT – II

Transaction Flow Testing: transaction flows, transaction flow testing techniques.

Dataflow testing: Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing. **Domain Testing:** domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT – III

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

UNIT – IV

State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.

UNIT – V

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like JMeter or Win-runner)..

Suggested Reading:

1	Software Testing techniques – Boris Beizer, Dreamtech, second edition.
2	Software Testing Tools – Dr. K.V.K.K. Prasad, Dreamtech.
3	Foundations of Software Testing, D. Graham & Others, Cengage Learning.
4	Introduction to Software Testing, P.Ammann & J.Offutt, Cambridge Univ. Press

(KMEC, NDAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PROFESSIONAL ELECTIVE-II

PE 523 AI	COMPUTER GRAPHICS					
Prerequisites	Data Structures and Discrete Mathematics		L	T	P	C
			3	0	0	3
Evaluation	CIE	30 Marks	SEE		70 Marks	

Course Objectives	
1	To introduce the fundamentals of computer graphics and shape-drawing algorithms.
2	To teach 2D transformations, coordinate systems, and clipping techniques.
3	To explain 3D object modeling using curves and surface representations.
4	To understand 3D transformations and viewing concepts.
5	To learn animation techniques and methods to detect visible surfaces in 3D scenes.

Course Outcomes On completion of this course, the student will be able to	
CO1	Understand the basics of computer graphics systems and draw basic shapes using standard algorithms.
CO2	Apply 2D transformations and clipping algorithms for graphics operations.
CO3	Represent 3D objects using curves and surfaces and apply basic lighting models.
CO4	Perform 3D transformations and projections for viewing 3D objects.
CO5	Understand computer animation techniques and methods for visible surface detection

UNIT – I

Introduction: Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random scan systems, graphics monitors and work stations and input devices

Output primitives: Points and lines, line drawing algorithms (Bresenham's and DDA Algorithm), mid-point, Circle and ellipse algorithms, Polygon Filling: Scan-line algorithm, boundary-fill and flood-fill algorithms.

UNIT – II

2-D geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems.

2-D viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland algorithms, Sutherland – Hodgeman polygon clipping algorithm.

UNIT – III

3-D object representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon rendering methods.

UNIT – IV

3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations. viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

UNIT – V
Computer Animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications Visible surface detection methods: Classification, back-face detection, depth-buffer, BSP-tree methods and area sub-division methods.

Suggested Reading:

1	"*Computer Graphics C version*", Donald Hearn and M. Pauline Baker, Pearson Education, 2nd Edition, 1997.
2	"*Computer Graphics Principles & Practice*", 2nd Edition in C, Foley, Van Dam, Feiner and Hughes, Pearson Education, 1996.
3	*Computer Graphics- A programming Approach*, Steven Harrington, TMH, 1987.

(KMEC, NAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.
19/9/25

PE 524 AI	COMPUTER VISION					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives:	
1	To explain how digital images are formed and processed using filtering and transforms.
2	To teach techniques for detecting features, segmenting images, and aligning images.
3	To introduce motion estimation and structure-from-motion concepts.
4	To develop skills in stitching images and enhancing them through computational photography.
5	To explore rendering techniques and object recognition in computer vision.

Course Outcomes On completion of this course, the student will be able to	
CO1	Identify the basic principles of image formation and preprocessing techniques.
CO2	Explain key concepts in feature detection, segmentation, and image alignment.
CO3	Apply motion estimation and structure-from-motion techniques to visual data.
CO4	Analyze image stitching and computational photography methods for enhancing image quality.
CO5	Evaluate and design image-based rendering and recognition systems for real-world vision tasks.

UNIT – I

Image Formation: Geometric primitives and transformations, Photometric image formation, The digital camera.

Image Processing: Image Preprocessing-Noise reduction-spatial filtering, frequency filtering, Point Operators, Linear filtering, More neighbourhood operators, Fourier Transforms, Pyramids and Wavelets.

UNIT – II

Feature detection and matching: Points and matches, Edges, Lines.

Segmentation: Active contours, Split and merge, mean shift and mode finding, Normalized cuts.

Feature-based alignment: 2D and 3D feature-based alignment, Pose estimation, Geometric intrinsic calibration.

UNIT – III

Morphological Operations: Structure from motion: Triangulation, Two-frame structure from motion, Factorization, Bundle adjustment. **Dense motion estimation:** Translational alignment, Parametric motion, Spline-based motion, Layered motion.

UNIT – IV

Image stitching: Motion Models, Global alignment, Compositing.

Computational photography: Photometric calibration, High dynamic range imaging, super-resolution and blur removal, image matting and compositing.

UNIT – V

Image-based rendering: View interpolation, Layered depth images, Light fields and Lumigraphs, Environment mattes, Video-based rendering.

Recognition: Object detection, Face recognition, Instance recognition, context and scene understanding

Suggested Reading:

1	"*Computer Vision: Algorithms and Applications*" Richard Szeliski, Springer, 2010
2	"*Numerical Methods for Computer Vision, Machine learning, and Graphics*", Justin Solomon, CRC Press, 2020.
3	*Digital Image Processing*, Rafael C. Gonzalez and Richard E. Woods, 3rd Edition, Prentice Hall 2008.

(KMEC, NAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC 551 AI	DATABASE MANAGEMENT SYSTEMS LAB					
Prerequisites			**L**	**T**	**P**	**C**
			0	**0**	**2**	**1**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	To understand and design database schemas using E-R and relational models.
2	To apply DDL and DML commands for creating and manipulating database objects.
3	To develop skills in querying databases using SQL including joins, functions, and subqueries.
4	To implement advanced SQL features like stored procedures, views, and triggers.
5	To demonstrate database administration concepts using DCL commands.

Course Outcomes On completion of this course, the student will be able to	
CO1	Recall and describe basic DBMS commands and database objects.
CO2	Understand relational schemas and apply DDL/DML commands to define and modify data.
CO3	Write SQL queries using operators, joins, and subqueries to retrieve required information.
CO4	Analyze the use of views, triggers, and stored procedures for managing complex operations.
CO5	Demonstrate user control using DCL commands for secure database access.

List of Experiments:

Scenario 1:

Product-Sales database: SouthWind

Southwind database is a sample database used by Organization. The database contains the sales data for SouthWind Traders, a foods export-import company. Using this schema to demonstrate how customers can choose and order products, how orders are placed and how those products get delivered to the customer.

Products: This Entity will have all the product details where suppliers will supply products based on customers demand.

Supplies: This Entity will supply the products demanded by the customers. Shippers: This Entity will take the orders from suppliers and deliver to customers. Employees: Employees will monitor the orders placed by customers.

Invoices: This Entity will take care of the billing process based on customer order.
Etc..identify some more entities and find out the relationship between them.

A product-sales the above process involves many steps like

1. Analyzing the problem and identifying the Entities and Relationships,
2. E-R Model
3. Relational Model
4. Normalization
5. Creating the database
6. Querying

Experiment 1: E-R Model

Analyze and come up with the entities in it. Identify what data has to be persisted in the database.

This contains the entities, attributes etc.

Identify the primary keys for all the entities. Identify the other keys like Foreign Key and constraints like NULL, NOT NULL, CHECK etc.

Example to create for **products, customers, suppliers, orders, employees, order details, categories**, among others.
Students should submit E-R diagrams using the above tables.

Experiment 2: Installation & DDL
Installation of Mysql and practicing DDL commands.

Creating databases, How to create tables, altering the database or tables, dropping tables and databases if not required. You will also try truncate, rename commands etc.

Data Definition Language (DDL) : create, alter, drop.

Experiment 3: DML
Data Manipulation Language Commands (DML) commands are used to for managing data within schema objects.

Exercising the commands using **DML** : insert, delete, update on the following tables: **products, customers, suppliers, orders, employees, order details, categories.**

- INSERT – insert data into a table.
  UPDATE – updates existing data within a table.
  DELETE – deletes single or all records from a table.

Data Query Language – Select

Populate all the tables designed in experiment : 2 with appropriate data.

Experiment 4: Querying
Practice queries on **Aggregate functions** like count, max, min ,avg ,sum Practice queries like nested queries/co-related queries using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSECT, groupby ,having etc.
Joins: Join, Left Outer Join, Right Outer Join, Self Join

Experiment 5: Querying (continued...)
Some example to practice the queries:

- Display all the order details given to a customer.
- Display all the products.
- Get the highest sold product from given supplier I
- List all products grouped by category
- List the products, whose product unit price is greater than all the products on average.
- List Details of order and customer of each order
- List the products which were sold in year 1997
- Display the total amount for each order
- Display Order Details for given an order ID

Order Details: product name and unit price for given order ID Exercising Simple to complex Queries using joins, nested and correlated queries.

Experiment 6: Stored Procedures
Create a stored procedure, Alter and Drop a procedure, IN, OUT, IN & OUT parameters

- Create a Procedure to display order details of given customer ID like ordered, order Date

, Required Date, Shipped Date

- Create a procedure to accept a customer ID and display the customer order history (product name and how much quantity ordered for that particular product) Ex: product name, Total quantity he/she ordered.
- Create a procedure to display Ten Most Expensive Products Columns should be displayed Product name & Unit price

Experiment 7: Views

Create a view to display the current product list which are available (not discontinued)

Create a view to display the products by category

Display product name, quantity Per Unit, units In Stock, Discontinued

Create a view as "Invoices" to display all the information from order, customer, shipper for each "Order Details"

Experiment 8: Triggers

Demonstrate Create Trigger, Alter Trigger, Drop Trigger, Row Level, Table Level triggers, Before Insert, After Insert, Before Update, After Update, Before Delete, After Delete

Experiment 9:

Demonstrate the role of DBA using DCL commands

Suggested Reading:

1	Raghurama Krishnan, Johannes Gehrke, "Database Management Systems", Tata McGraw Hill, 3rd Edition, 2008.
2	Silberschatz, Korth, "Database System Concepts", McGraw Hill, V edition, 2005.
3	M. Mc Laughlin, "Oracle Database 11g PL/SQL Programming", TMH, 2017

(KMEC, NAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC 452 CS	OPERATING SYSTEMS LAB					
Prerequisites			**L**	**T**	**P**	**C**
			0	**0**	**2**	**1**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	Learn different types of CPU scheduling algorithms.t
2	Demonstrate the usage of semaphores for solving synchronization problem
3	Understand memory management techniques and different types of fragmentation
4	That occur in them and various page replacement policies
5	Understand Banker‘s algorithm used for deadlock avoidance.
6	Learn various disk scheduling algorithms.

Course Outcomes	
On completion of this course, the student will be able to	
CO1	Evaluate the performance of different types of CPU scheduling algorithms.
CO2	Implement Producer consumer problem, Readers Writers problem, Dining Philospere’s problem
CO3	Simulate Banker’s algorithm for deadlock avoidance
CO4	Critically analyze the performance of the various Memory management algorithms
CO5	Use different system calls for writing application programs.

1. CaseStydy: Perform a case study by installing and exploring various types of operating systems on a physical or logical (virtual) machine.

List of Experiments (preferred programming language is C)

1. Write a C programs to implement UNIX system calls and file management
2. Write C programs to demonstrate various process related concepts.
3. Write C programs to demonstrate various thread related concepts.
4. Write C programs to simulate CPU scheduling algorithms: FCFS, SJF, Round Robin
5. Write C programs to simulate Intra & Inter-Process Communication (IPC) techniques: Pipes, Messages Queues, Shared Memory.
6. Write C programs to simulate solutions to Classical Process Synchronization Problems: Dining Philosophers, Producer-Consumer, Readers-Writers
7. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.
8. Write C programs to simulate Page Replacement Algorithms: FIFO, LRU
9. Write C programs to simulate implementation of Disk Scheduling Algorithms: FCFS, SSTF.

Software Required:

StarUML/Umbrello, NetBeans/Eclipse IDE, XAMPP/MEAN stack, JUnit, JMeter, Selenium, Bugzilla

(KMEC, NAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC 552 AI	MACHINE LEARNING LAB					
Prerequisites			**L**	**T**	**P**	**C**
			0	**0**	**2**	**1**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	To Learn how to set up Python and use basic libraries for machine learning.
2	To Understand and apply different classification algorithms to predict outcomes.
3	To Use clustering techniques to group data without labels.
4	To Compare traditional models with ensemble methods like boosting and bagging.
5	To Evaluate model performance and apply advanced methods like CNN and Genetic Algorithms.

Course Outcomes On completion of this course, the student will be able to	
CO1	Set up Python and perform basic data preprocessing using ML libraries.
CO2	Build and use classification models to solve prediction problems.
CO3	Apply clustering algorithms to analyze and group data.
CO4	Compare single models with ensemble techniques to improve accuracy.
CO5	Use performance metrics and apply CNN or GA for real-world problems.

List of Experiments:

1. Basic Data Preprocessing
   a. Installation of python environment/Anaconda IDE for machine learning installing python modules/Packages like scikit-learn, Keras and Tensorflow.
   b. Programs involving pandas, Numpy and Scipy libraries.
2. Programs for classification
   a. Build models using linear regression and logistic regression and apply it to classify a new instance.
   b. Write a program to demonstrate the following classifiers. Use an appropriate data set for building the model. Apply the model to classify a new instance.
      i) Decision tree
      ii) K nearest neighbour
      iii) Naïve bayes
      iv) Support vector machine
3. Demonstration of Clustering algorithms using
   a. K-means
   b. Hierarchical algorithms
4. Demonstrate ensemble techniques like boosting, bagging, random forests
5. Build a classifier, compare its performance with an ensemble technique like random forest.
6. Evaluate various classification algorithms performance on a dataset using various measures like True Positive rate, False positive rate, precision, recall.
7. Demonstrate GA for optimization (minimization or maximization problem)
1. Case study on supervised/unsupervised learning algorithm:
   a) Handwritten digits classification using CNN
   b) Text classification using python libraries.

(KMEC, NGAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

SCHEME OF INSTRUCTION

Common to BE (CSE (AI&ML), CSE(AI), AI & ML)

SEMESTER- VI

S. No.	Course Code	Course Title	Scheme of Instruction			Contact Hours / Week	Scheme of Examination		Credits
			L	T	P		CIE	SEE	
Theory									
1	PC 601 AI	Deep Learning	3	0	0	3	30	70	3
2	PC 602 AI	Computer Networks	3	0	0	3	30	70	3
3	PC 603 AI	Cloud Computing	3	0	0	3	30	70	3
4	**Professional Elective-III**		3	0	0	3	30	70	3
	PE 631 AI	Internet of Things							
	PE 632 AI	Quantum Computing							
	PE 633 AI	Natural language processing							
	PE 634 AI	Distributed Databases							
5		**Open Elective-I**	3	0	0	3	30	70	3
Practicals									
6	PC651AI	Deep learning Lab	0	0	2	2	25	50	1
7	PC553CS	Computer Networks Lab	0	0	2	2	25	50	1
8	PW651AI	Mini Project	0	0	2	2	25	50	1
9	SI651AI	Summer Internship	Six Weeks during summer vacation Evaluation will be done in VII-Semester						
Total			**15**	**0**	**6**	**21**	**225**	**500**	**18**

(KMEC, NOAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC 601 AI	DEEP LEARNING					
Prerequisites	Artificial Intelligence		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To introduce the concepts of AI, Machine Learning, and Deep Learning.
2	To explain the structure and components of neural networks.
3	To teach how to implement neural networks using TensorFlow.
4	To explore different deep learning architectures and their functions.
5	To understand the applications of deep learning in real-world problems.

Course Outcomes On completion of this course, the student will be able to	
CO1	Understand the basics of AI, Machine Learning, and Deep Learning.
CO2	Explain how neural networks work with layers and activation functions.
CO3	Build simple neural networks using TensorFlow.
CO4	Compare deep learning models like CNNs, RNNs, and others.
CO5	Identify real-world uses of deep learning in areas like vision and speech.

UNIT – I

What is deep learning? Artificial intelligence, Machine learning, and Deep learning - Artificial intelligence -Machine learning – Learning representations from data - The “deep” in deep learning - Understanding how deep learning works, in three figures -What deep learning has achieved so far- The promise of AI.

UNIT – II

Getting started with neural networks - Anatomy of a neural network - Layers: the building blocks of deep learning - Models: networks of layers - Loss functions and optimizers: keys to configuring the learning process The Neural Network-Building Intelligent Machines, The Limits of Traditional Computer Programs, The Mechanics of Machine Learning, The Neuron, Expressing Linear Perceptron’s as Neurons, Feed-Forward Neural Networks, Linear Neurons and Their Limitations, Sigmoid, Tanh.

UNIT – III

Implementing Neural Networks in Tensor Flow - What is Tensor Flow? - How Does Tensor Flow Compare to Alternatives?- Installing Tensor Flow - Creating and Manipulating Tensor Flow Variables – Tensor Flow Operations - Placeholder Tensors - Sessions in Tensor Flow - Navigating Variable Scopes and Sharing Variables - Managing Models over the CPU and GPU.

UNIT – IV

Introduction to Major Architectures of Deep Networks–Unsupervised Pretrained Networks (UPNs), Convolutional Neural Networks (CNNs), Recurrent Neural Networks, Recursive Neural Networks.

UNIT – V
Deep Learning Applications - Large Scale Deep Learning - Computer Vision - Speech Recognition - Natural Language Processing - Other Applications.

Suggested Reading:

1	Nikhil Buduma and Nicholas Locascio - Fundamentals of Deep Learning : Designing Next-Generation Machine Intelligence Algorithms – First Edition - O'Reilly , 2017
2	Francois Chollet-Deep Learning with Python-Second Edition,Manning Publications, 2017.
3	Uma N Dulhare, Essam Houssain- Deep Learning and Computer Vision: Models and Biomedical Applications:Vol1 & Vol 2 - Springer Publication 2025
4	Ian Goodfellow, Yoshua Bengio, Aaron Courville - Deep Learning – Second Edition- MIT Press , 2016.
5	Josh Patterson and Adam Gibson- Deep Learning: A Practitioner's Approach - First Edition - O'Reilly , 2017

(KMEC, NGIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.
19/9/25

PC 602 AI	COMPUTER NETWORKS					
Prerequisites	**Data Structures and Programming Concepts**		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To Understand data communication basics, transmission media, and network models.
2	To Learn error detection, correction methods, and access control protocols.
3	To Study addressing, switching, and inter-networking in the network layer.
4	To Explore transport layer protocols, socket programming, and QoS techniques.
5	To Understand application layer protocols and basic network security concepts.

Course Outcomes On completion of this course, the student will be able to	
CO1	Describe communication components, OSI/TCP-IP models, and bandwidth usage techniques.
CO2	Explain error control and multiple access methods in data transmission.
CO3	Apply addressing, switching, and internetworking concepts in networks.
CO4	Compare transport protocols and implement basic socket programming.
CO5	Identify application protocols and use basic cryptography for secure communication.

UNIT – I

Data Communications Components : Representation of data and its flow, Networks, Layered architecture, OSI and TCP/IP model, Transmission Media.

Techniques for Bandwidth utilization: Line configuration, Multiplexing – Frequency division, Time division and Wave division, Asynchronous and Synchronous transmission, XDSL, wireless LAN

UNIT – II

Data Link Layer and Medium Access Sub Layer: Error Detection and Error Correction – Fundamentals, Block coding, Hamming Distance, CRC;

Flow Control and Error control protocols: Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, and Piggybacking.

Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA

UNIT – III

Network Layer: Switching techniques (circuit and packet), Logical addressing – IPV4, IPV6, subnetting concepts.

Inter-networking: Tunnelling, Fragmentation, congestion control, Internet control

UNIT – IV

Transport Layer: Process to Process Communication, Elements of transport protocol, Introduction Socket Programming.

Internet Transport Protocols: UDP, TCP, SCTP; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

UNIT – V
Application Layer: Domain Name Space (DNS), EMAIL, SNMP, Bluetooth, VOIP. **Basic concepts of Cryptography:** Network Security Attacks, firewalls, symmetric encryption, Data encryption standards, Public key Encryption (RSA), Hash function, Message authentication, Digital Signature.

Suggested Reading:

1	*Computer Networks*, 5th Edition, Andrew S. Tanenbaum , David J. Wetherall , Pearson Education, 2021
2	*Computer Networks: A Systems Approach*, Larry Peterson and Bruce Davie, Elsevier , 5th Edition, 2021
3	*Computer Networking: A Top-Down Approach*, 6th Edition, James F. Kurose , Keith W. Ross , Pearson , 2022

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC 603 AI	**CLOUD COMPUTING**					
Prerequisites			L	T	P	C
			3	0	0	3
Evaluation	CIE	30 Marks	SEE		70 Marks	

Course Objectives	
1	To Understand the basics, benefits, and challenges of cloud computing and virtualization.
2	To Learn about cloud scaling, resource planning, load balancing, and storage systems.
3	To Explore cloud data handling, databases, security models, and privacy issues.
4	To Study interoperability, cloud management tools, and popular cloud platforms.
5	To Understand enterprise cloud systems, software architecture, and business processes.

Course Outcomes On completion of this course, the student will be able to	
CO1	Describe cloud computing concepts, services, and virtualization techniques.
CO2	Explain cloud scalability, capacity planning, and load balancing methods.
CO3	Analyze data management, security, and compliance in cloud environments.
CO4	Identify issues in cloud portability and evaluate cloud management solutions.
CO5	Apply cloud computing to enterprise systems and business process integration.

UNIT – I
Introduction, Benefits and challenges, Cloud computing services, Resource Virtualization, Resource pooling sharing and provisioning.

UNIT – II
Scaling in the Cloud, Capacity Planning , Load Balancing, File System and Storage

UNIT – III
.Multi-tenant Software, Data in Cloud , Database Technology, Content Delivery Network, Security Reference Model , Security Issues, Privacy and Compliance Issues

UNIT – IV
Portability and Interoperability Issues, Cloud Management and a Programming Model Case Study, Popular Cloud Services

UNIT – V
Enterprise architecture and SOA, Enterprise Software , Enterprise Custom Applications, Workflow and Business Processes, Enterprise Analytics and Search, Enterprise Cloud Computing Ecosystem

Suggested Reading:

1	Cloud Computing , Sandeep Bhowmik, Cambridge University Press, 2017
2	Enterprise Cloud Computing - Technology, Architecture, Applicatios, Gautam Shroff, Cambridge University Press, 2016
3	Kai Hwang, Geoffrey C.Fox, Jack J.Dongarra, Distributed and Cloud Computing From Parallel Processing to the Internet of Things, Elsevier, 2012.

(KMEC, NDAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PROFESSIONAL ELECTIVE-III

PE 631 AI	INTERNET OF THINGS					
Prerequisites	**Computer Organization and Microprocessors**		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To Understand the basics, applications, and security of IoT.
2	To Learn about internet communication protocols used in IoT.
3	To Gain skills in prototyping and programming IoT devices.
4	To Explore cloud services and data analytics for IoT systems.
5	To Understand IoT product development, business models, and ethics.

Course Outcomes On completion of this course, the student will be able to	
CO1	Describe IoT concepts, applications, and device-level challenges.
CO2	Explain communication protocols like IP, TCP, UDP, and HTTP.
CO3	Develop simple IoT projects using Arduino or Raspberry Pi.
CO4	Use cloud platforms and data analytics tools for IoT solutions.
CO5	Identify steps in IoT product manufacturing and address ethical issues.

UNIT – I

Introduction to Internet of Things: IOT vision, Strategic research and innovation directions, Iot Applications, Related future technologies, Infrastructure, Networks and communications, Processes, Data Management, Security, Device level energy issues

UNIT – II

Internet Principles and Communication Technology: Internet Communications: An Overview – IP, TCP, IP protocol Suite, UDP. IP addresses – DNS, Static and Dynamic IP addresses, MAC Addressess, TCP and UDP Ports, Application Layer Protocols – HTTP,HTTPS, Cost Vs Ease of Production, Prototypes and Production, Open Source Vs Closed Source

UNIT – III

Prototyping and programming for IoT : Prototyping Embedded Devices – Sensors, Actuators, Microcontrollers, SoC, Choosing a platform, Prototyping, Hardware platforms – Arduino, Raspberry Pi. Prototyping the physical design – Laser Cutting, 3D printing, CNC Milling.
Techniques for writing embedded C code: Integer data types in C, Manipulating bits - AND,OR,XOR,NOT, Reading and writing from I/ O ports. Simple Embedded C programs for LED Blinking, Control of motor using switch and temperature sensor for arduino board.

UNIT – IV

Cloud computing and Data Analytics: Introduction to Cloud storage models -SAAS, PAAS, IAAS. Communication APIs, Amazon web services for IoT, Skynet IoT Messaging Platform.
Introduction to Data Analytics for IoT - Apache Hadoop- Map Reduce job execution workflow

UNIT – V
IoT Product Manufacturing - From prototype to reality, Business model for IoT product manufacturing, Business models canvas, Funding an IoT Startup, Masss manufacturing - designing kits, designing PCB, 3D Printing, certification, Scaling up software, Ethical issues in IoT - Privacy, Control, Environment, solutions to ethical issues.

Suggested Reading:

1	*Internet of Things - Converging Technologies for smart environments and Integrated ecosystems*, Ovidiu Vermesan, Peter Friess, River Publishers, 2022
2	*Designing the Internet of Things* , Adrian McEwen, Hakim Cassimally. Wiley India Publishers, 2013.
3	*Fundamentals of Embedded Software: where C meets assembly*, Daniel W Lewis, Prentice Hall , 2002.

(KMEC, NGIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PE 632 AI	QUANTUM COMPUTING					
Prerequisites	**Operating Systems**		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To Understand basic linear algebra, complex numbers, and set theory relevant to quantum computing.
2	To Learn essential quantum physics concepts and quantum theory needed for computation.
3	To Explore quantum computing architecture, hardware, and the role of qubits.
4	To Study key quantum algorithms and understand their working principles.
5	To Understand current cryptographic methods and how quantum computing affects them.

Course Outcomes On completion of this course, the student will be able to	
CO1	Apply linear algebra and complex number concepts in the context of quantum computing.
CO2	Explain basic quantum physics principles including quantum states and entanglement.
CO3	Describe the structure of quantum computers, qubits, gates, and architectures.
CO4	Analyze the working of major quantum algorithms like Shor's and Grover's.
CO5	Evaluate the impact of quantum computing on existing cryptographic systems.

UNIT – I:

Introduction to Essential Linear Algebra: Some Basic Algebra, Matrix Math, Vectors and Vector Spaces, Set Theory. **Complex Numbers:** Definition of Complex Numbers, Algebra of Complex Numbers, Complex Numbers Graphically, Vector Representations of Complex Numbers, Pauli Matrice, Transcendental Numbers

UNIT – II:

Basic Physics for Quantum Computing: The Journey to Quantum, Quantum Physics Essentials, Basic Atomic Structure, Hilbert Spaces, Uncertainty, Quantum States, Entanglement.
Basic Quantum Theory: Further with Quantum Mechanics, Quantum Decoherence, Quantum Electrodynamics, Quantum Chromodynamics, Feynman Diagram
Quantum Entanglement and QKD, Quantum Entanglement, Interpretation, QKE

UNIT – III

Quantum Architecture: Further with Qubits, Quantum Gates, More with Gates, Quantum Circuits, The D-Wave Quantum Architecture. **Quantum Hardware:** Qubits, How Many Qubits Are Needed? Addressing Decoherence, Topological Quantum Computing, Quantum Essentials.

UNIT – IV

Quantum Algorithms: What Is an Algorithm? Deutsch's Algorithm, Deutsch-Jozsa Algorithm, Bernstein-Vazirani Algorithm, Simon's Algorithm, Shor's Algorithm, Grover's Algorithm.

UNIT – V
Current Asymmetric Algorithms: RSA, Diffie-Hellman, Elliptic Curve. **The Impact of Quantum Computing on Cryptography:** Asymmetric Cryptography, Specific Algorithms, Specific Applications.

Suggested Reading:

1	Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press
2	Dr. Chuck Easttom, Quantum Computing Fundamentals, Pearson
3	Quantum Computing for Computer Scientists by Noson S. Yanofsky and Mirco A. Mannucci

(KMEC, NOAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PE 633 AI	**NATURAL LANGUAGE PROCESSING**					
Prerequisites	**Data Mining and Machine Learning**		**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To Understand basic language processing using Python and access lexical resources.
2	To Learn to process raw text using regular expressions and word tagging techniques.
3	To Gain knowledge on text classification, information extraction, and NER.
4	To Study syntax and sentence structure using context-free grammars and parsing.
5	To Explore NLP applications such as sentiment analysis, translation, and speech processing.

Course Outcomes On completion of this course, the student will be able to	
CO1	Apply Python for text analysis and use resources like WordNet and corpora.
CO2	Process and clean raw text using regular expressions and tagging methods.
CO3	Build classifiers and perform information extraction from textual data.
CO4	Analyze grammatical structure using parsing and context-free grammar.
CO5	Implement real-world NLP applications like sentiment analysis and translation.

UNIT – I

Language Processing and Python: Computing with Language: Texts and Words, A Closer Look at Python: Texts as Lists of Words, Computing with Language: Simple Statistics, Back to Python: Making Decisions and Taking Control, Automatic Natural Language Understanding Accessing Text Corpora and Lexical Resources: Accessing Text Corpora, Conditional Frequency Distributions, Lexical Resources, WordNet.

UNIT – II

Processing Raw Text: Accessing Text from the Web and from Disk, Strings: Text Processing at the Lowest Level, Text Processing with Unicode, Regular Expressions for Detecting Word Patterns, Useful Applications of Regular Expressions, Normalizing Text, Regular Expressions for Tokenizing Text, Segmentation, Formatting: From Lists to Strings. Categorizing and Tagging Words: Using a Tagger, Tagged Corpora, Mapping Words to Properties Using Python Dictionaries, Automatic Tagging, N-Gram Tagging, Transformation Based Tagging, How to Determine the Category of a Word**.**

UNIT – III

Learning to Classify Text: Supervised Classification, Evaluation, Naive Bayes Classifiers Extracting Information from Text: Information Extraction, Chunking, Developing and Evaluating Chunkers, Recursion in Linguistic Structure, Named Entity Recognition, Relation Extraction.

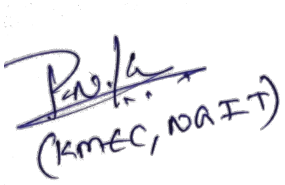
UNIT – IV

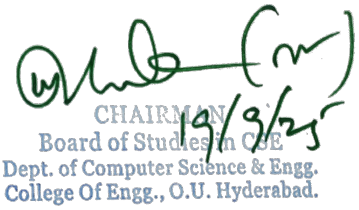
Analyzing Sentence Structure: Some Grammatical Dilemmas, Usage of Syntax. Context-Free Grammar, Parsing with Context-Free Grammar, Dependencies and Dependency Grammar, Grammar Development, Building Feature-Based Grammars.

UNIT – V
NLP Applications: Topic modeling, Text classification, Sentiment analysis, Word sense disambiguation, Speech recognition and speech to text, Text to speech, Language detection and translation.

Suggested Reading:

1	Steven Bird, Ewan Klein, and Edward Lope, " *Natural Language Processing with Python*", O"Reily, 2009.
2	Akshay Kulkarni, Adarsha Shivananda, " *Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning using Python*", Apress, 2019.





PE 634 AI	**DISTRIBUTED DATABASES**					
Prerequisites			**L**	**T**	**P**	**C**
			3	**0**	**0**	**3**
Evaluation	**CIE**	**30 Marks**	**SEE**		**70 Marks**	

Course Objectives	
1	To Understand database concepts, architectures, relational models, and algebra operations.
2	To Learn query processing, optimization techniques, and cost estimation methods.
3	To Study parallel database systems, architectures, and levels of parallelism.
4	To Understand distributed database design, fragmentation, and access mechanisms.
5	To Explore global query translation, access strategy optimization, and database security.

Course Outcomes On completion of this course, the student will be able to	
CO1	Explain the fundamentals of database systems, relational models, and data operations.
CO2	Analyze query processing and apply optimization techniques to improve performance.
CO3	Describe architectures and techniques used in parallel database systems.
CO4	Design distributed databases using fragmentation and allocation strategies.
CO5	Optimize global queries and understand access control and security in distributed systems.

UNIT – I

Introduction: Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Object-Based and Semistructured Databases, Data Storage and Querying, Transaction Management, Data Mining and Analysis, Database Architecture, Database Users and Administrators, History of Database Systems.

Relational Model: Structure of Relational Databases, Fundamental Relational-Algebra Operations, Additional Relational-Algebra Operations, Extended Relational-Algebra Operations, Null Values, Modification of the Database.

UNIT – II

Query Processing: Overview, Measures of query cost, Selection operation, sorting, Join operation, other operations, Evaluation of Expressions.

Query Optimization: Overview, Transformation of Relational expressions, Estimating statistics of expression results, Choice of evaluation plans, Materialized views.

UNIT – III

Parallel Systems: Speedup and Scaleup, Interconnection Networks, Parallel Database Architectures.

Parallel Databases: Introduction, I/O Parallelism, Interquery Parallelism, Intraquery Parallelism, Interoperation Parallelism, Intraoperation Parallelism, Design of Parallel Systems.

UNIT – IV
Distributed Databases: Reference architecture for DDB, Types of Data Fragmentation, Distribution Transparency for Read-only applications, Distribution Transparency for Update applications, Distributed Database Access Primitives, Integrity Constraints in DDB. Distributed Database Design: A framework for Distributed Database Design, The design of Database fragmentation, The allocation of fragmentation.

UNIT – V
Translation of Global Queries to Fragment Queries: Equivalence transformations for queries, Transforming global queries into fragment queries, Distributed grouping and aggregate function evaluation, Parametric queries.Optimization of Access Strategies: Access Control Models, Database Security, A framework for query optimization, Join queries, General queries.

Suggested Reading:

1	Silberschatz A, Korth HF, Sudarshan S, *Database System Concepts*, McGraw-Hill International Edition, 5th Edition, 2006
2	Ceri S, Pelagatti G, *Distributed Databases: Principles and Systems*, McGraw-Hill International Edition, 1984

(KMEC, NGIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

PC 651 AI	**DEEP LEARNING LAB**					
Prerequisites			**L**	**T**	**P**	**C**
			0	**0**	**2**	**1**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	Understand the concepts of Artificial Neural Networks and Deep Learning concepts.
2	Implement ANN and DL algorithms with Tensorflow and Keras.
3	Gain knowledge on Sequence learning with RNN.
4	Gain knowledge on Image processing and analysis with CNN.
5	Get information on advanced concepts of computer vision.

Course Outcomes On completion of this course, the student will be able to	
CO1	Develop ANN without using Machine Learning/Deep learning libraries.
CO2	Understand the Training ANN model with backpropagation.
CO3	Develop a model for sequence learning using RNN.
CO4	Develop image classification models using ANN and CNN.
CO5	Generate a new image with auto-encoder and GAN.

List of Experiments

1. Create Tensors and perform basic operations with tensors

2. Create Tensors and apply split & merge operations and statistics operations

3. Design a single unit perceptron for classification of Iris dataset without using predefined models

4. Design, train, and test the MLP for tabular data and verify various activation functions and optimizers using Tensorflow

5. Design and implement to classify 32x32 images using MLP using Tensorflow/Keras and check the accuracy

6. Design and implement a simple RNN model with Tensorflow/Keras and check accuracy

7. Design and implement LSTM model with Tensorflow/Keras and check accuracy

8. Design and implement GRU model with Tensorflow/Keras and check accuracy

9. Design and implement a CNN model to classify multi-category JPG images with Tensorflow/Keras and check accuracy. Predict labels for new images

10. Design and implement a CNN model to classify multi-category tiff images with Tensorflow/Keras and check accuracy. Check whether your model is overfit/underfit/perfect fit and apply the techniques to avoid overfit and underfit
11. Implement CNN architectures (LeNet, AlexNet, VGG, etc.) model to classify multi-

category Satellite images with Tensorflow/Keras and check the accuracy. Check whether your model is overfit/underfit/perfect fit and apply the techniques to avoid overfit and underfit 12. Implement an Autoencoder to de-noise image 13. Implement a GAN application to convert images

(KMEC, NDAIT)

19/9/25
MJ CC-1

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.
19/9/25

PC 652 AI	**COMPUTER NETWORKS LAB**					
Prerequisites			**L**	**T**	**P**	**C**
			0	**0**	**4**	**2**
Evaluation	**CIE**	**25 Marks**	**SEE**		**50 Marks**	

Course Objectives	
1	To Learn to use basic network commands and tools for checking and analyzing networks.
2	To Understand how to set up and configure routers and switches using simulators or real devices.
3	To Build simple client-server programs using TCP and UDP socket programming.
4	To Analyze network traffic and simulate network behavior using tools like Wireshark and NS2/NS3.
5	To Practice advanced network programming using raw sockets and RPC.

Course Outcomes On completion of this course, the student will be able to	
CO1	Use basic network tools to check network activity.
CO2	Set up and configure network devices like routers and switches correctly.
CO3	Write and test TCP/UDP programs like time server, echo server, and DNS using sockets.
CO4	Capture and study network packets and simulate network behavior using tools like NS2/NS3.
CO5	Create network programs using raw sockets and remote procedure calls (RPC).

List of Programs	
1	Running and using services/commands like tcpdump, netstat, ifconfig, nslookup, FTP, TELNET and traceroute. Capture ping and trace route PDUs using a network protocol analyzer and examine.
2	Configuration of router, switch. (using real devices or simulators)
3	Socket programming using UDP and TCP (e.g. simple DNS, data & time client/server, echo client/server, iterative & concurrent servers)
4	Network packet analysis using tools like Wireshark, tcpdump, etc.
5	Network simulation using tools like Cisco Packet Tracer, NetSim, OMNeT++, NS2, NS3, etc.
6	Study of Network simulator (NS) and Simulation of Congestion Control Algorithms using NS. Performance evaluation of Routing protocols using Simulation tools.
7	Programming using raw sockets
8	Programming using RPC

Prof. (KMEC, NOAIT)

19/9/25 MJ CC-1

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.
19/9/25

PW651 AI	MINI PROJECT					
Prerequisites			L	T	P	C
			0	0	2	1
Evaluation	CIE	25	SEE 50		50 Marks	

Course Objectives

1	To help students identify and understand real-world problems in their field.
2.	To encourage application of technical knowledge and skills learned in previous courses.
3	To improve students' ability to work in teams and manage projects effectively.
4	To enhance communication skills through technical writing and presentations.
5	To promote awareness of current technologies, industry practices, and research trends.

Course Outcomes

After completion of the course , Student will be able to	
CO1	Identify a relevant problem and define a clear project objective.
CO2	Apply technical and analytical skills to design and develop a solution.
CO3	Work effectively as a team to plan, implement, and manage the project.
CO4	Communicate ideas clearly through oral presentations and written reports.
CO5	Demonstrate awareness of industry standards and modern tools used in their domain

The department can initiate the project allotment procedure at the end of V semester and finalize it in the first two weeks of VI semester.
The department will appoint a project coordinator who will coordinate the following: Collection of project topics/ descriptions from faculty members (Problems can also be invited from the industries) Grouping of students (max 3 in a group)

Allotment of project guides

The aim of the mini project is to develop solutions to realistic problems applying the knowledge and skills obtained in different courses, new technologies and current industry practices. This requires students to understand current problems in their domain and methodologies to solve these problems.
To get awareness on current problems and solution techniques, the first Two (2) weeks of VI semester will be spent on special lectures by faculty members, research scholars, post graduate students of the department and invited lectures by engineers from industries and R&D institutions.
After completion of these seminars each group has to formalize the project proposal based on their own ideas or as suggested by the project guide.
Seminar schedule will be prepared by the coordinator for all the students from the 5th week to the last week of the semester which should be strictly adhered to.
Each group will be required to:

1. Submit a one page synopsis before the seminar for display on the notice board.
2. Give a 30 minutes presentation followed by a 10 minutes discussion.

3. Submit a technical write-up on the talk.

At least two teachers will be associated with the Mini Project to evaluate students for the award of sessional marks which will be on the basis of performance in all the 3 items stated above.

The seminar presentation should include the following components of the project:

*Problem definition and specification

*Literature survey

*Broad knowledge of available techniques to solve a particular problem.

*Planning of the work, preparation of bar (activity) charts

*Presentation- oral and write

(KMEC, NDAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.

SI 651 AI	SUMMER INTERNSHIP					
Prerequisites			**L**	**T**	**P**	**C**
			-	-	-	-
Evaluation	**SEE**	-	**CIE**		-	

Course Objectives:

1.	To expose students to real-time industrial environments and practices.
2.	To help students understand how classroom knowledge is applied in the industry.
3.	To encourage professional interactions and collaboration with industry experts.
4	To develop problem-solving and project execution skills in real-world scenarios.
5	To improve technical reporting and presentation abilities.

Course Outcomes:	
After completion of this course student will be able to do:	
CO1	Gain hands-on experience by working on real industry problems.
CO2	Apply academic knowledge to practical and professional situations.
CO3	Interact and work effectively with professionals in a team environment.
CO4	Prepare a clear and concise technical report based on the work done.
CO5	Deliver a structured seminar presentation summarizing the internship experience.

Summer Internship is introduced as part of the curricula of encouraging students to work on problems of interest to industries. A batch of three students will be attached to a person from the Computer Industry/Software Companies/R&D Organization for a period of 8 weeks.

This will be during the summer vacation following the completion of the III year Course. One faculty coordinator will also be attached to the group of Three (3) students to monitor the progress and to interact with the industry co-ordinate (person from industry). After the completion of the project, student will submit a brief technical report on the project executed and present the work through a seminar talk to be organized by the Department.

Award of sessionals are to be based on the performance of the students, to be judged by a committee constituted by the department. One faculty member will co-ordinate the overall activity of Industry Attachment Program.

Students have to undergo summer internship of Six Weeks duration at the end of semester VI and the credits will be awarded after evaluation in VII semester

(KMEC, NDAIT)

19/9/25

CHAIRMAN
Board of Studies in CSE
Dept. of Computer Science & Engg.
College Of Engg., O.U. Hyderabad.