



Fergusson College (Autonomous) Pune

Learning Outcomes-Based Curriculum for

**Industrial Mathematics with
Computer Applications (IMCA)**

With effect from June 2026-2027

**Deccan Education Society's
Fergusson College (Autonomous), Pune
Program Outcomes (POs)**

PO1	Disciplinary Knowledge: Demonstrate comprehensive knowledge of the discipline that forms a part of a postgraduate programme. Execute strong theoretical and practical understanding generated from the specific programme in the area of work.
PO2	Critical Thinking and Problem solving: Exhibit the skill of critical thinking and understand scientific texts and place scientific statements and themes in contexts and also evaluate them in terms of generic conventions. Identify the problem by observing the situation closely, take actions and apply lateral thinking and analytical skills to design the solutions.
PO3	Social competence: Exhibit thoughts and ideas effectively in writing and orally; communicate with others using appropriate media, build effective interactive and presenting skills to meet global competencies. Elicit views of others, present complex information in a clear and concise way and help reach conclusions in group settings.
PO4	Research-related skills and Scientific temper: Infer scientific literature, build a sense of enquiry and able to formulate, test, analyse, interpret, and establish hypothesis and research questions; and to identify and consult relevant sources to find answers. Plan and write a research paper/project while emphasizing on academics and research ethics, scientific conduct and creating awareness about intellectual property rights and issues of plagiarism.
PO5	Trans-disciplinary knowledge: Create new conceptual, theoretical and methodological understanding that integrates and transcends beyond discipline-specific approaches to address a common problem.
PO6	Personal and professional competence: Perform independently and also collaboratively as a part of a team to meet defined objectives and carry out work across interdisciplinary fields. Execute interpersonal relationships, self-motivation and adaptability skills and commit to professional ethics.
PO7	Effective Citizenship and Ethics: Demonstrate empathetic social concern and equity centred national development, and ability to act with an informed awareness of moral and ethical issues and commit to professional ethics and responsibility.
PO8	Environment and Sustainability: Understand the impact of the scientific solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development.
PO9	Self-directed and Life-long learning: Acquire the ability to engage in independent and life-long learning in the broadest context of socio-technological changes.

Department of Mathematics
Programme: M.Sc. Industrial Mathematics with Computer Applications

PSO No.	Program Specific Outcomes (PSOs) Upon completion of this Programme the student will be able to
PSO1	Academic competence Understand basic facts about Mathematics -annotations, terminology, geometrical figures, graphical displays, and its major subfields (Analysis, Algebra, Applied Mathematics and Statistics). Design and develop computer programs/computer-based systems in the areas related to algorithms, networking, web design, cloud computing, IoT and data analytics. (i) Demonstrate unifying structures of Mathematics and the basic constructs of several programming languages and their application areas.
PSO2	Personal and Professional Competence ● Apply mathematical solutions in a variety of contexts related to science, technology, business and industry, and carry out hands-on activities on several programming technologies as well as academic projects. ● Analyze the data by selecting and using appropriate mathematical formulae or techniques/programming technology stack in order to draw the relevant conclusion. ● Execute the problem-solving skills, equip with Mathematical modelling abilities and develop competent professionals who will be able to address challenges in the field of IT at global level.
PSO3	Research Competence (i) Apply advanced knowledge on topics in pure Mathematics and impart analytical skills to develop initiatives and come up with innovative ideas for R&D in various fields. (ii) Integrate the knowledge of Computer Science and Mathematics to solve and authenticate real-time data from various fields.
PSO4	Entrepreneurial and Social competence (i) Develop analytical skills required to get distinguishing employment opportunities in several fields including IT, Research and Development, teaching field and gain understanding about the ethical issues related to protection of intellectual property - copyrights, trademarks, and patents. (ii) Execute social competence including communication and effective interaction with others, listening, speaking, observational skills and presenting skills.

Programme Structure

PG II:

Semester	Paper Code	Paper Title	Credits
III	MTS -601	Design and Analysis of Algorithms	4
	MTS -607	Cryptography	4
	MTS -604 OR MTS -606 OR	Dynamical Systems	4
		Computational Intelligence	4
	MTS -608 & MTS - 609	Operating Systems	2
		Computer Networks	2
	MTS -610	Research Project	4
	MTS -622	Practical Lab VI: Introduction to Web Technologies	2
	MTS -623	Practical Lab VII: Machine Learning Techniques using Python	2
	MNC - 02	Mandatory Non-credit Course (Cyber Security -2)	2
	MNC - 03	Mandatory Non-credit Course (Skill Development - 2)	2
	Total Semester Credits		20
	IV	MTS -653	Financial Mathematics
OR MTS -654		Optimization Techniques	
OR MTS -654		High-Performance Computing	
MTS -660		Research Project	6
MTS -671 OR MTS -672 OR MTS -673		Emerging Technologies: Introduction to Blockchain Technology	2
		Emerging Technologies: Cloud Computing	
		Emerging Technologies: Edge Computing	
MTS -674		Industrial Training	4
MTS-675		OJT (On Job Training)	4
Total Semester Credits		20	
Total PG-II Credits			40

Teaching and Evaluation (Only for FORMAL education courses)

Course Credits	No. of Hours per Semester	No. of Hours per Week	Maximum Marks	CE 40 %	ESE 60%
	Theory/Practical	Theory/Practical			
1	15 / 30	1 / 2	25	10	15
2	30 / 60	2 / 4	50	20	30
3	45 / 90	3 / 6	75	30	45
4	60 / 120	4 / 8	100	40	60

`Eligibility: As per the rules and regulations of Savitribai Phule Pune University (SPPU)

S.Y. M.Sc. Semester III		
MTS-601	Design and Analysis of Algorithms	Number of Credits : 04
Course Outcomes (COs)		
On completion of the course, the students will be able to:		
CO1	Describe basic concepts in algorithms	
CO2	Illustrate the concepts of recurrences, hash tables to run complex functions	
CO3	Solve problems based on Polynomial and Non polynomial time approaches	
CO4	Analyze concepts of dynamic programming, greedy and graph theoretic algorithms	
CO5	Evaluate and compare performance of various algorithms	
CO6	Develop algorithm based on design techniques to compute asymptotic complexities	

Unit No	Title of Unit and Contents
I	Growth Functions Asymptotic Notations
II	Recurrences The Substitution Method, Recursion Tree Method, Master theorem
III	Sorting of Ordered Statistics Heap Sort, Quick Sort, Sorting in Linear Time, Median and Ordered Statistics
IV	Dynamic Programming Matrix Chain Multiplication, Longest Common subsequence, Optimal Binary Trees
V	Greedy Algorithm Activity Selection problem, Elements of Greedy Strategy, Huffman Codes
VI	String Matching The Naïve String Matching Algorithm, The Rabin Karp Algorithm, String Matching with finite automata, The Knuth-Morris-Pratt algorithm
VII	Hash Tables Direct Address Tables, Hash Tables, Hash Functions, Open Addressing, Perfect Hashing
VIII	NP completeness Informal concepts of Deterministic and Non deterministic, P , NP and NP Completeness, Statement of Cooks Theorem, Discussion on Vertex cover problem, Discussion on Travelling Salesman Problem

Learning Resources:

1. Introduction to Algorithms, H Coremen, Rivest, Stein and Leiserson
2. Fundamentals of Algorithms, Horowiitz, Sahani and Rajasekeran
3. Mathematics for Analysis of Algorithms, Donald Knuth and H.Green
4. Algorithm Design, Eva Tardos , Jon Klienberg
5. An Introduction to Analysis of Algorithms, Micheal Soltys
6. The Algorithm Design Manual, Steven Skiena

S.Y. M.Sc. Semester III		
MTS-607	Cryptography	Number of Credits : 04
Course Outcomes (COs)		
On completion of the course, the students will be able to:		
CO1	Describe key notions and principles related to basic Cryptography.	
CO2	Discuss the notion of classical cryptography, basic principles, theory of public key.	
CO3	Implement basic symmetric key algorithms and public key algorithms using programming language.	
CO4	Explain different cryptographic algorithms.	
CO5	Justify different cryptographic algorithms to obtain security and to encrypt, decrypt data, different types of attacks on data.	
CO6	Integrate basic algorithms on elliptic curves and its use in cryptography.	

Unit No	Title of Unit and Contents
I	Basic number theoretic concepts Congruence, Chinese Remainder Theorem, Primitive Roots, Quadratic reciprocity Finite fields, Arithmetic functions, Primality Testing and factorization algorithms, Pseudo-primes, Fermat's pseudo-primes, Prime Numbers, Testing for Primality, Continued fractions, Hash Functions
II	Classical Cryptosystems Cryptography in Modern world, Substitution cipher, Ceaser cipher as a special case of substitution cipher, Mono alphabetic ciphers Transposition Cipher, Polyalphabetic substitution ciphers, Vigenère Cipher, Introduction to poly-graphic substitution ciphers cryptanalysis of substitution cipher
III	Symmetric Key Cryptography Introduction and overview, Stream Cipher, Block ciphers, one-time Pad, Modes of operation electronic codebook, Digital signature, Data Encryption Standard, Advanced Encryption Standard, IDEA (International Data Encryption Algorithm), Various attacks on data encryption algorithms
IV	Public Key Cryptography Introduction and Overview, The RSA algorithm, Generation of keys, Diffie Hellman Key Agreement protocol, El Gamal Encryption Algorithms, Discrete Logarithm, Attacks against RSA, Discrete Logarithm Problem
V	Elliptic Curve Cryptography Introduction and Overview, Elliptic Curves over Real Numbers, Elliptic curves over finite fields, Elliptic curve cryptography, Diffie Hellman key exchange over Elliptic curves.
VI	Hashing Motivation and applications, Cryptographically secure hashing, Message authentication codes (MAC), Hash Algorithm - MD5, Hash Algorithms – SHA, Examples of Hashing using programming language.
VII	Introduction to Quantum Cryptography Over view of quantum cryptography and its applications.

Learning Resources:

- 1) D. R. Stinson: CRYPTOGRAPHY, Theory and practice, CRC Press, 1995
- 2) Neil Koblitz: A course in Number theory and Cryptography, 2nd Edition, Springer
- 3) Robert Edward Lewand: Cryptological Mathematics (Mathematical Association of America).
- 4) Jeffrey Hoffstein, Jill Pipher, Joseph H. Silverman: An introduction to Mathematical Cryptography, Springer
- 5) Adam J. Elbirt: (CRC press): Understanding and Applying cryptography and Data security.
- 6) Bruce Schneier: Applied Cryptography (Wiley India Edition)
- 7) Atul Kahate: Cryptography and Network security (Tata McGraw Hill)
- 8) Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016. (Free download available)
- 9) Joseph Bonneau et al, SoK: Research perspectives and challenges for Bitcoin and cryptocurrency, IEEE Symposium on security and Privacy, 2015 (article available for free download) { curtain raiser kind of generic article, written by seasoned experts and pioneers}.

S.Y. M.Sc. Semester III		
MTS-604	Dynamical Systems	Number of Credits : 04
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Retrieve basic concepts in Differential Equations and Systems	
CO2	Interpret properties of solutions to planar systems	
CO3	Apply the concept of limit cycles and bifurcations to have a detailed study of planar autonomous systems	
CO4	Analyze the interacting Species model within the survival ecosystem	
CO5	Formulate the theory of Hamiltonian systems, Lyapunov Functions and determine its stability	
CO6	Review the theory of three-dimensional autonomous systems and Chaos	

Unit No	Title of Unit and Contents
I	Differential Equations Simple Differential Equations, Applications to Chemical Kinematics, Applications to Electrical Circuits, Existence and Uniqueness Theorem
II	Planar Systems Canonical Forms, Eigen values defining Stable and Unstable Manifolds, Phase Plane Portraits, Linearization and Hartman's Theorem, Constructing Phase Plane Diagram
III	Interacting Species Competing Species, Predator Prey Models, Other Characteristics Affecting Interacting Species
IV	Limit Cycles Existence and Uniqueness of Limit Cycles in a Plane, Non-Existence of Limit Cycles,
V	Hamiltonian Cycles, Lyapunov Functions and Stability Hamiltonian Systems in the plane, Lyapunov Systems and Stability
VI	Bifurcation Theory Bifurcations of Non-Linear Systems, Normal Forms, Multistability and Bistability
VII	Three Dimensional Autonomous Systems and Chaos Linear System and Canonical Forms, Non-Linear Systems and Stability, The Rossler System and Chaos, The Lorentz Equations

Learning Resources

1. Stephen Lynch, Dynamical Systems with Applications using Python, Birkhauser Publications
2. Morris Hirsch, Stephen Smale, Differential Equations, Dynamical Systems and introduction to Chaos, Academic Press
3. James .D. Miess, Differential Dynamical Systems, Society for Industrial and Applied Mathematics (SIAM)
4. Lawrence Perko, Differential Equations and Dynamical Systems, Springer Texts in Applied Mathematics
5. Kathleen Alligood, Tim Sauer, James Yorke, CHAOS An introduction to Dynamical Systems, Springer

S.Y. M.Sc. Semester III		
MTS-606	Computational Intelligence	Number of Credits :4
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Describe key concepts associated with computational intelligence and hard computing.	
CO2	Illustrate various concepts associated with Genetic algorithms and compare Generic Algorithm vs. Traditional Algorithms	
CO3	Apply the concept of Artificial neural network to find linear separability and solve various problems.	
CO4	Explain and Compute Mathematical properties of Fuzzy Logic, Classical Sets and Fuzzy Sets and solve examples to justify the properties.	
CO5	Review the concept of Swarm Computing and Ant colony optimization, formulate the concept of Associative Memory Network and Neuro Fuzzy Modeling to discuss real life problems.	
CO6	Design and analyze the concept of Supervised learning and unsupervised learning to solve real life problems.	

Unit No.	Title of Unit and Contents
I	Introduction to computational intelligence and Soft computing What is soft computing? Differences between soft computing and hard computing, Soft Computing constituents, Methods in soft computing, Applications of Soft Computing
II	Introduction to Fuzzy Logic, Classical Sets and Fuzzy Sets Operations on classical and fuzzy sets Properties of Classical and fuzzy Sets Classical Relations and Fuzzy Relations Tolerance and Equivalence Relations Membership Functions Features of the Membership Functions Fuzzification Methods of Membership Value Assignments Defuzzification Lambda-Cuts for Fuzzy Sets (Alpha-Cuts) Lambda-Cuts for Fuzzy Relations Defuzzification Methods Fuzzy Arithmetic and Fuzzy Measures Extension Principle Fuzzy Measures Fuzzy Integrals Introduction to Fuzzy Rule Base and Approximate Reasoning Introduction to Fuzzy Decision Making Introduction to Fuzzy Logic Control Systems
III	Genetic Algorithms (GA) What are Genetic Algorithms? Why Generic Algorithms? Biological Background Traditional Optimization and Search Techniques Genetic Algorithm and Search Space Generic Algorithm vs. Traditional Algorithms Simple GA, General Genetic Algorithm, Operators in Generic Algorithm

IV	Artificial Neural Network: Fundamental Concept Evolution of Neural Networks Basic Models of Artificial Neural Network Important Terminologies of ANN McCulloch-Pitts Neuron Linear Separability Hebb Network
V	Supervised Learning Network Introduction Perceptron Nonworks Adaptive Linear Neuron (Adaline) Multiple Adaptive Linear Neurons Back-Propagation Network
VI	Unsupervised Learning Networks Introduction Fixed Weight Competitive Nets Kohonen Self-Organizing Feature Maps Learning Vector Quantization Counter propagation Networks Adaptive Resonance Theory Network
VII	Introduction to Neuro Fuzzy Modeling, Swarm Computing and Ant colony optimization

Learning Resources:

- Timothy J. Ross, Fuzzy Logic: With Engineering Applications Wiley India, Third Edition ISBN: 978-81-265-3126-4
- Kumar Satish, Neural Networks: A Classroom Approach, 1/e TMH, ISBN: 9780070482920
- David E. Goldberg, Genetic Algorithms in search, Optimization & Machine Learning by, Pearson Education, ISBN: 81-7808-130-X
- James A. Freeman and David M. Skapura, Neural Networks Algorithms, Applications, and Programming Techniques, Edition: Pearson Edn., 2003.V. Kecman, Learning and Soft Computing, MIT Press, 2001.
- J. S. R. Jang, C. T. Sun, and E. Mizutani, "Neuro-Fuzzy and Soft Computing", PHI, 2004, Pearson Education.
- S. N. Sivanandam, S. N. Deepa, Principles of Soft Computing (With CD) Wiley India, ISBN: 9788126527410
- S. Rajasekaran and G. A. Vijayalakshmi Pai, Neural Network, Fuzzy Logic and Genetic Algorithms - Synthesis and Applications", (2005), Prentice Hall
- Konar A., "Computational Intelligence: Principles, Techniques and Applications", Springer Verlag, 2005
- Engelbrecht, A.P, "Fundamentals of Computational Swarm Intelligence", John Wiley & Sons, 2006.

S.Y. M.Sc. Semester III		
MTS-608	Operating Systems	Number of Credits :2
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Explain the structure and core functions of an Operating System, including kernel architectures, system calls, virtualization, and process/resource abstraction.	
CO2	Apply process management concepts to analyze scheduling algorithms and evaluate their performance based on throughput, turnaround time, response time, and fairness.	
CO3	Analyze concurrency problems and implement synchronization mechanisms (semaphores, monitors) to prevent race conditions and resolve classical synchronization problems.	
CO4	Examine memory management techniques and compare virtual memory and page replacement algorithms in terms of efficiency and system performance.	
CO5	Evaluate I/O scheduling methods, storage organization, RAID configurations, and file system structures to assess reliability and performance trade-offs.	
CO6	Assess operating system security mechanisms and propose appropriate protection strategies using access control models and security principles for secure system design.	

Unit No.	Title of Unit and Contents
I	OS Structure & Foundations (4 Lectures) • Role & functions of an OS in industrial computing • System calls, kernel architecture (monolithic, microkernel, modular, hybrid) • Process and resource abstraction • Virtualization, containers (Docker), hypervisors (concepts only)
II	Process Management & Scheduling (6 Lectures) • Process model: PCB, context switching, threads, multicore & parallelism • Scheduling goals: fairness, throughput, turnaround, response time • Algorithms: FCFS, SJF, SRTF, Round Robin, Priority, Multilevel queues • Real-time scheduling basics (Rate-monotonic, EDF (Earliest Deadline First) – concepts)
III	Concurrency, Synchronization & Deadlocks (6 Lectures) • Race conditions, mutual exclusion • Semaphores, monitors, reader-writer problems • Classical synchronization problems (Producer-Consumer, Dining Philosophers) • Deadlock modelling: resource allocation graphs, deadlock detection & recovery • Deadlock avoidance

IV	Memory Management & Virtual Memory (6 Lectures) ● Contiguous vs non-contiguous allocation; fragmentation ● Paging, segmentation, TLB, address translation ● Concepts of locality, working set model ● Page replacement algorithms: FIFO, LRU, Optimal, Clock ● Thrashing: causes & remedies
V	I/O Systems, Storage & File Systems (5 Lectures) ● Disk organization & performance parameters ● Disk scheduling: FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK ● RAID levels: reliability & performance characteristics ● File system structure: inodes, directory organization, allocation methods ● Journaling & crash recovery (conceptual)
VI	OS Security & Protection (3 Lectures) ● Security principles: least privilege, isolation, sandboxing ● Access control models: DAC, MAC, RBAC ● Authentication methods; capabilities ● Basic vulnerability concepts (buffer overflow, privilege escalation – conceptual)

Learning Resources:

- Abraham Silberschatz, Peter B. Galvin, Greg Gagne, *Operating System Concepts*, Wiley India, Tenth Edition, ISBN: 978-1-119-32091-3
- William Stallings, *Operating Systems: Internals and Design Principles*, Pearson India, Fifth Edition, ISBN: 978-0-13-467095-9
- Andrew S. Tanenbaum, Herbert Bos, *Modern Operating Systems*, Pearson India, Fourth Edition, ISBN: 978-0-13-359162-0
- Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau, *Operating Systems: Three Easy Pieces*, Arpaci-Dusseau Books, Third Edition, ISBN: 978-1-984-50867-3
- Robert Love, *Linux Kernel Development*, Addison-Wesley Professional, Second Edition, ISBN: 978-0-672-32946-3
- Nigel Poulton, *Docker Deep Dive*, Nigel Poulton, Second Edition, ISBN: 978-1-909-67183-8

S.Y. M.Sc. Semester III		
MTS-609	Computer Networks	Number of Credits :2
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Explain basic networking concepts and types of networks used in real life.	
CO2	Describe OSI and TCP/IP models and how data travels on the internet.	
CO3	Identify IP addresses, ports, and basic routing concepts.	
CO4	Understand how web, email, and file transfer applications work.	
CO5	Apply basic subnetting and network configuration concepts.	
CO6	Recognize modern networking trends like cloud, Wi-Fi, and network security.	

Unit No.	Title of Unit and Contents
I	Networking Basics - What is Computer Networking? - Importance of networks in modern industry (cloud, internet, mobile apps, IoT) - Types of networks: LAN, WAN, Internet - Network topologies: star, bus, ring, mesh (basic idea only) - Data communication basics: sender, receiver, medium, protocol - Introduction to protocols and standards (HTTP, TCP/IP, Wi-Fi) - How internet works, home/office networks, cloud connectivity
II	Network Models & Internet Architecture - Why layered models are used - OSI model (basic functions of each layer) - TCP/IP model (practical internet model) - IP address, MAC address, Port number (basic understanding) - Introduction to DNS and how websites work - How data travels when we open a website or send email
III	Data Transmission Basics - Analog vs digital data (simple idea) - Transmission media: wired, wireless, fiber - Bandwidth and speed basics - Switching concepts: packet switching (used in internet) - Basic idea of multiplexing - Internet speed, Wi-Fi, mobile networks, fiber optics
IV	Data Link Layer & Error Control - Framing concept - Error detection basics: parity, CRC (concept only) - Flow control (stop and wait idea) - Introduction to MAC address - Ethernet and Wi-Fi basics - How data reaches correct device in local network
V	Network Layer (Internet Layer) - IPv4 addressing basics - Subnet mask (simple problems) - IPv6 introduction (why needed) - Routing basics (how packets travel across networks) - NAT and routers - IP addressing, routers, home network configuration

VI	Transport Layer ● TCP vs UDP (simple comparison) ● Ports and sockets ● Reliable vs fast communication ● Basic idea of congestion control ● Video streaming vs email vs gaming traffic
VII	Application Layer & Modern Internet ● DNS (how domain names work) ● HTTP/HTTPS basics ● Email working (SMTP, POP, IMAP – concept only) ● FTP and file transfer ● Introduction to cloud networking and APIs ● Web applications, cloud services, online tools
VIII	Modern Networking Concepts ● Wi-Fi, mobile networks (4G/5G basics) ● Network security basics (firewall, encryption, HTTPS) ● Virtual networks & cloud basics ● Introduction to IoT networking ● Cybersecurity awareness, cloud computing, real-world networking

Learning Resources:

1. Data Communications and Networking, Behrouz A. Forouzan, Fifth Edition -McGraw-Hill Forouzan Networking Series
2. Computer Networks, Andrew S. Tanenbaum, David Wetherall, Fourth Edition, 5th Edition Pearson

S.Y. M.Sc. Semester III		
MTS-610	Research Project	Number of Credits :4
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Critical Thinking: Encourage critical thinking skills by requiring students to evaluate and synthesize existing literature, identify research gaps, and propose innovative solutions or perspectives principles of research, including its purpose, significance, and different methodologies.	
CO2	Hypothesis Formulation: Help students formulate clear and testable hypotheses or research questions, guiding them through the process of defining the scope and objectives of their research	
CO3	Research Design: Familiarize students with various research designs and methodologies, enabling them to choose the most appropriate approach for their specific research question	
CO4	Data Collection and Analysis: Train students in the methods of collecting and analysing data, whether through qualitative or quantitative approaches, and help them develop skills in using relevant tools and techniques	
CO5	Ethical Considerations: Instil a strong understanding of research ethics, ensuring that students conduct them research in a responsible and ethical manner, respecting the rights and well-being of participants.	
CO6	Communication, Time management and problem-solving Skills: Improve students' ability to communicate their research effectively, both in written and oral forms. This includes preparing research proposals, reports, and presenting findings to diverse audiences Develop effective time management skills, helping students create realistic timelines, set milestones, and manage their project efficiently. Problem-solving: Cultivate problem-solving skills by challenging students to overcome obstacles and adapt their research methods when faced with unexpected issues.	

S. Y. M. Sc. Semester IV		
MTS-622	Practical Lab VI: Introduction to Web Technologies	Number of Credits:02
Course Outcome (COs) On completion of the course, the students will be able to:		
CO1	State design process for single-page applications using ReactJS.	
CO2	Illustrate the use of JavaScript in webpages.	
CO3	Demonstrate React JS and Node JS features	
CO4	Analyze the importance of user experience in designing of the websites through various web-based technologies.	
CO5	Test and validate web applications using JavaScript.	
CO6	Write web pages using various web technologies.	

Unit No	Title of Unit and Contents
1	Overview of HTML5 and CSS3 What is HTML? Tags, Heading, paragraph, anchor, image, table, Lists (ordered, unordered, description), form, label, input, button, br, hr, script (simple JavaScript code) Audio, Video, Progress, Data list Tag, Header Tag, Footer Tag, Article Tag, Aside Tag, Canvas, SVG, Google Maps, Geolocation, Web Storage What is CSS? Syntax, Selector, Types, Comments Background, colors, Border, Margin, Padding, Height/Width, Box Model, Outline, Text, Fonts, Icons, Links, Lists, Tables, Display, Max-width, Position, Overflow, Inline-block etc. (Cover basic properties)
2	JavaScript Introduction, JavaScript BOM, Comments, Variables, Operators, Data Types, Functions, Objects, Scope, Events (onclick, onchange, onmouseover, onmouseout, onkeydown, onload), Strings, String Methods (indexOf(), lastIndexOf(), search(), slice(), substring(), substr()), Numbers, Number Methods (toString(), toExponential(), toFixed(), toPrecision(), valueOf()), Math (min (), max (), pow (), random (), sqrt(), ceil(), floor() methods), Array, Array methods (toString(), join(), pop(), push(), shift(), unshift(), splice(),concat()), Date & its methods, Booleans, Comparisons, Conditions, Switch, for loop, while loop, break, Type Conversion, Debugging, Class, Objects, Object Properties, Object Methods Asynchronous Programming Callbacks, Promises. Async-Await

2	Introduction to ReactJS What is ReactJS? Benefits, Who is using it?, SPA(Single Page Application), What is JSX, What are state and props? , Stateful and Stateless components , CSS and ReactJS : CSS Modules , render() method , Lifecycle methods, Adding Styles to React Elements, Import and Export of Modules, Creating Class-based Components , setState() Method, How to pass Props to Class-based Components, How to pass Function as Props, Case study
3	Node Js : What is Node.js? Installing Node.js and Visual Studio Code, what is Node.js? What it's use? Your first Node.js script, Node.js Module System, how to import Node.js core modules? How to import your own files? Importing npm modules Global npm modules and nodemon, getting input from user, Storing data with JSON, ES6 Arrow functions, Debugging Node.js, Introduction to Asynchronous Node.js, Asynchronous basics, Http requests, Handling errors, Callback function, Node.js as a Web Server, Serving up HTML and JSON, Case Study

Learning Resources:

- Beginning HTML5 and CSS3 by Christopher Murphy, Divya Manian, Oliver Stud Holme and Richard W. Clark (APress)
- Beginning JavaScript by Jeremy McPeak and Paul Wilton (Wrox)
- Beginning JQuery by Jack Franklin (APress)
- Bootstrap by Jake Spurlock (O'Reilly)
- Head First HTML5 Programming, Building Web Apps with JavaScript By Eric Freeman, Elisabeth Robson (O'Reilly)
- Head First JavaScript Programming By Eric T. Freeman, Elisabeth Robson (O'Reilly)
- Head First Ajax By Rebecca M. Riordan (O'Reilly)
- Head First jQuery, A Brain-Friendly Guide By Ryan Benedetti, Ronan Cranley (O'Reilly)
- Roger S. Pressman. *Software Engineering: A Practitioner's Approach* (Sixth Edition, International Edition). McGraw-Hill, 2005.
- Ian Sommerville. *Software Engineering* (Seventh Edition). Addison-Wesley, 2004.

Important URLs: <https://www.w3schools.com/>, <https://getbootstrap.com/>

S. Y. M. Sc. Semester IV		
MTS-623	Practical Lab VII: Machine Learning Techniques using Python	Number of Credits: 02
Course Outcome (COs) On completion of the course, the students will be able to:		
CO1	Explain fundamental concepts and mathematical foundations of machine learning.	
CO2	Apply regression and classification algorithms to modeling problems.	
CO3	Understand graphical models and ensemble techniques.	
CO4	Analyze performance of ML models using appropriate evaluation methods.	
CO5	Explain theoretical aspects like bias-variance tradeoff and learning theory	
CO6	Apply clustering and pattern mining techniques to discover patterns in data.	

Unit No	Title of Unit and Contents
1.	Overview of Python : Variables and Data Structures in Python, Control Flow and functions and Modules, Use of libraries : Numpy, Matplotlib, Pandas
2.	Foundations of Machine Learning : Types of learning: supervised, unsupervised, semi-supervised, reinforcement, Applications of ML in industry, Basics required for ML: Probability theory (random variables, expectation, variance), Linear algebra (vectors, matrices, dot product), Optimization basics (loss function, gradient descent)
3	Regression Analysis : Linear regression (simple & multiple), Loss functions and least squares method, Overfitting and regularization, Ridge regression, Lasso regression, Bayesian regression (basic idea), Regression using basis functions, Model evaluation (MSE, R^2)
4	Classification Methods I: Introduction to classification, Linear classifiers, Logistic regression, Linear Discriminant Analysis (LDA), Perceptron algorithm, Concept of decision boundaries, Model evaluation: confusion matrix, accuracy, precision, recall, K-Fold Cross-Validation: Splitting data into k subsets to ensure the model generalizes well. Stratified Shuffling: Ensuring class balance in each fold (critical for classification). Bias - Variance Trade-off: Using cross-validation to detect if the model is overfitting.
5	Classification Methods II : Support Vector Machines and Kernels, Large margin classification, Support Vector Machines (SVM), Kernel methods (linear, polynomial, RBF), Decision Trees, Multilayer Perceptron (MLP), Backpropagation (basic idea)
6	Clustering Techniques : Introduction to clustering, Distance measures, k-means clustering, k-medoids, Hierarchical clustering, Agglomerative Divisive
7	Introduction to Deep Learning : Single Layered Neural Networks , Multiplayered Neural Networks, Convolutional Neural Networks- Convolution Layers, Pooling Layers, Architecture of convolutional Neural Network, Data Augmentation, Results using Pretrained Classifier
8	Sequence Models and Modern Architectures: Recurrent Neural Networks : Document Classification Sequential Models for Document, Time Series Forecasting, Concept of Attention Mechanism , Transformer Basics: Encoder-Decoder Architecture , self attention, positional encoder and Applications

Learning Resources :

1. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Jonathan Taylor, An Introduction to Statistical Learning with Applications in Python, 2023, Springer
2. Data Mining: Tools and Techniques, 3rd Edition. Jiawei Han and Michelline Kamber
3. Elements of Statistical Learning. Hastie, Tibshirani, and Friedman. Springer
4. Pattern Recognition and Machine Learning. Christopher Bishop.
5. Kevin R Murphy, Machine Learning - A Probabilistic Perspective, The MIT Press, 2012.

S. Y. M. Sc. Semester IV		
MTS-653	Financial Mathematics	Number of Credits: 04
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Retrieve basic concepts in Mathematical Finance	
CO2	Interpret properties of Brownian and Geometric Brownian motion	
CO3	Implement Arbitrage theorem to value investments	
CO4	Analyze investments using Expected utility	
CO5	Formulate the theory of Black Scholes to value options in continuous time	
CO6	Choose the appropriate hedging strategy to value against options	

Unit No	Title of Unit and Contents
I	Brownian Motion and Geometric Brownian Motion Brownian Motion, Brownian Motion as Limit of Simpler Models, Geometric Brownian Motion, The Maximum Variable, The Cameron Martin Theorem
II	Interest Rates and Present Value Analysis Interest Rates, Present Value Analysis, Rate of Return, continuously varying Interest Rates
III	Pricing Contracts vs Arbitrage An example of Options Pricing, Other Examples of Pricing via Arbitrage, The Arbitrage Theorem, The Multi-period Binomial Model, Proof of Arbitrage theorem
IV	Black Scholes Formula Black Scholes Formula, Properties of Black Scholes Option Cost, The Delta Hedging Arbitrage Strategy, Some Derivations, European Put Options
V	Additional Results and Options Call Options on Dividend Paying Securities, Pricing American Put Options, Adding Jumps to Geometric Brownian Motion, Estimating Volatility Parameter
VI	Valuing by Expected Utility Limitations of Arbitrage Pricing, Valuing Investments by Expected Utility, The Portfolio Selection problem, Value at Risk and Conditional Value at risk, The Capital Assets Pricing Model, Rates of Return: Single Period and Geometric Brownian Motion

Learning Resources:

1. Sheldon M Ross, An Elementary Introduction to Mathematical Finance, Third Edition, Cambridge University Press.
2. D.G. Luenberger, Investment Science, Oxford University Press
3. Marek Capinski, An introduction to Financial Engineering, Springer Publications. Amber Habib, Calculus of Finance, Universities Press
4. John Hull, Sanskaran Basu, Options, Futures & Derivatives, Pearson Editi

S. Y. M. Sc. Semester IV		
MTS-654	Optimization Techniques	Number of Credits :4
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Identify and state basic concepts in Linear, Non-linear programming and Game theory.	
CO2	Interpret the Game as a Linear Programming problem and discuss methods to solve them.	
CO3	Apply methods to solve Integer programming problems and examine the solutions	
CO4	Analyze the primal-dual relationship of a Linear programming problem and compute the dual.	
CO5	Determine local solutions to develop techniques and solve non-linear programming problems.	
CO6	Formulate and solve a Linear Programming problem using Simplex method.	

Unit No	Title of Unit and Contents
I	Introduction to Linear Programming Prototype Example, The Linear Programming Model, Assumptions of Linear Programming, Additional Examples, Case Studies
II	Solving Linear Programming Problem: Simplex Method The Essence of Simplex Method, Setting up the Simplex Method, Algebra of Simplex method, Simplex Method in Tabular Form, Tie Breaking in Simplex Method, Adapting to Other forms, Post Optimality Analysis, Conclusions, Case Studies
III	Duality and Sensitivity Analysis The Essence of Duality Theory, Economic Interpretation of Duality, Primal Dual Relationships, Adapting to Other Primal Forms, The Role of Duality in Sensitivity Analysis, The Essence of Sensitivity Analysis, Applying Sensitivity Analysis, Conclusions, Case Studies
IV	Integer Programming Prototype Example, Some BIP Applications, Innovative use of Binary Variables in Model Formulation, Some Formulation Examples, Some Perspectives of solving Integer Programming Problems, The Branch and Bound Technique and its applications to Integer Programming, A Branch and Bound Technique for Mixed Integer Programming, Other Developments in solving BIP Problems, Conclusions, Case Studies
V	Non-Linear Programming Sample Applications, Graphical Illustration of Non Linear Programming Problems, Types of Non Linear Programming Problems, One Variable unconstrained Optimization, Multivariable unconstrained Optimization, The Karush Kuhn Tucker conditions for constrained Optimization, Quadratic Programming, Separable Programming, Convex Programming, Non Convex Programming, Conclusions, Case Studies

VI	Game Theory The Formulation of Two Person Zero Sum Games, Solving Simple Games- Prototype Example, Games with Mixed Strategies, Graphical Solution Procedure, Solving by Linear Programming, Extensions , Conclusion
-----------	--

References:

1. Frederick Hiller & Gerald Lieberman, Introduction to Operational Research, McGraw-Hill
2. Mykel J Kochenderfer and Tim Wheeler, Algorithms for Optimization, MIT Press
3. S.D. Sharma, Operations Research, Kedarnath Ram Nath
4. Prem Kumar Gupta, D.S. Hira, Operations Research

S. Y. M. Sc. Semester IV		
MTS-655	High-Performance Computing	Number of Credits :4
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	To understand the fundamentals of High Performance Computing, parallel architectures, and supercomputing systems	
CO2	To analyze performance models and scalability using Amdahl's and Gustafson's Laws.	
CO3	To design and implement shared-memory parallel programs using OpenMP.	
CO4	To develop distributed-memory applications using Message Passing Interface.	
CO5	To implement GPU-based parallel solutions using CUDA and understand GPGPU concepts.	
CO6	To apply HPC techniques to solve real-world scientific, engineering, and industrial mathematical problems efficiently.	

Unit No	Title of Unit and Contents
I	Introduction to High Performance Computing: Evolution of computing: Serial to parallel systems, Need for HPC in scientific, engineering, and industrial applications, Flynn's Classification – SISD, SIMD, MISD, MIMD, Shared vs Distributed memory systems, Multicore and many-core architectures, Basics of cluster computing, Introduction to Supercomputers
II	Parallel Architectures and Performance Models: Memory hierarchy and cache coherence, Interconnection networks, Parallel computer architecture, Distributed computing systems, Parallel programming paradigms, Data parallelism, Task parallelism, Performance metrics, Speedup, Efficiency, Scalability: Amdahl's Law, Gustafson's Law
III	Shared Memory Programming – OpenMP: Introduction to OpenMP, • Fork-join model, Parallel regions and work-sharing constructs, Synchronization (critical, atomic, barrier), Scheduling techniques, Reduction operations, Case studies: Matrix multiplication, Numerical integration
IV	Distributed Memory Programming – MPI: Introduction to MPI, • Point-to-point communication, Collective communication, Communicators and groups, Process topology, Parallel matrix operations using MPI, Implementation examples: Parallel sum, Parallel matrix multiplication, Distributed numerical computations

V	GPU Computing & GPGPU: Introduction to GPU architecture, Comparison: CPU vs GPU, CUDA programming model (conceptual + practical orientation), Thread hierarchy (Grid, Block, Thread), Memory types in GPU (Global, Shared, Constant, Registers), Kernel functions, Synchronization in GPU, Performance bottlenecks in GPU computing, Optimization techniques, GPGPU concepts, Case studies: Scientific computation using GPU, Parallel matrix operations on GPU, Numerical methods acceleration
VI	Applications of HPC in Industrial Mathematics: HPC in Numerical Linear Algebra, Parallel solution of large systems of equations, Simulation and Modeling, Finite Difference Method (parallel implementation overview), Big Data analytics (conceptual introduction), HPC in Machine Learning, Real-world industrial case studies

Learning Resources:

Textbooks:

- 1) Quinn, M. J. "Parallel Programming in C with MPI and OpenMP", 1st Edition, Tata McGraw Hill Education, 2003.
- 2) Pacheco, P. "An Introduction to Parallel Programming", 1st Edition, Morgan Kaufmann, 2011.
- 3) Grama, A., Gupta, A., Karypis, G., & Kumar, V. "Introduction to Parallel Computing", 2nd Edition, Pearson Education, 2003.

References :

- 1) Gropp, W., Lusk, E., & Skjellum, A. "Using MPI: Portable Parallel Programming with the Message-Passing Interface", 2nd Edition, MIT Press, 1999.
- 2) Snu, V. S. "Parallel Programming: Concepts and Practice", PHI Learning, 2020.
- 3) Chandra, R., Menon, R., Dagum, L., Kohr, D., Maydan, D., & McDonald, J. "Parallel Programming in OpenMP", Morgan Kaufmann, 2001.

S. Y. M. Sc. Semester IV		
MTS-660	Research Project	Number of Credits :6
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Critical Thinking: Encourage critical thinking skills by requiring students to evaluate and synthesize existing literature, identify research gaps, and propose innovative solutions or perspectives principles of research, including its purpose, significance, and different methodologies.	
CO2	Hypothesis Formulation: Help students formulate clear and testable hypotheses or research questions, guiding them through the process of defining the scope and objectives of their research	
CO3	Research Design: Familiarize students with various research designs and methodologies, enabling them to choose the most appropriate approach for their specific research question	
CO4	Data Collection and Analysis: Train students in the methods of collecting and analysing data, whether through qualitative or quantitative approaches, and help them develop skills in using relevant tools and techniques	
CO5	Ethical Considerations: Instil a strong understanding of research ethics, ensuring that students conduct their research in a responsible and ethical manner, respecting the rights and well-being of participants.	
CO6	Communication, Time management and problem-solving Skills: Improve students' ability to communicate their research effectively, both in written and oral forms. This includes preparing research proposals, reports, and presenting findings to diverse audiences Develop effective time management skills, helping students create realistic timelines, set milestones, and manage their project efficiently. Problem-solving: Cultivate problem-solving skills by challenging students to overcome obstacles and adapt their research methods when faced with unexpected issues.	

S. Y. M. Sc. Semester IV		
MTS-671	Emerging Technologies: Introduction to Blockchain Technology	Number of Credits : 02
Course Outcomes (COs)		
On completion of the course, the students will be able to:		
CO1	Understanding of basic concepts of Blockchain.	
CO2	Learn the technology used for blockchain.	
CO3	Familiarize the primitives of the distributed computing and cryptography related to blockchain.	
CO4	Illustrate the concepts of Bitcoin and their usage.	
CO5	To understand the Ethereum and smart contracts in blockchain technology	
CO6	Understand use of smart contracts in real world applications.	

Unit No	Title of Unit and Contents
I	Introduction: Need for Distributed Record Keeping, Modeling faults and adversaries, Byzantine Generals problem, Consensus algorithms and their scalability problems, Nakamoto's concept with Blockchain based cryptocurrency, Technologies Borrowed in Blockchain – hash pointers, consensus, byzantine fault-tolerant distributed computing, digital cash etc.
II	Basic Distributed Computing & Crypto primitives: Atomic Broadcast, Consensus, Byzantine Models of fault tolerance, Hash functions, Puzzle friendly Hash, Collision resistant hash, digital signatures, public key crypto, verifiable random functions, Zero-knowledge systems
III	Bitcoin basics: Bitcoin blockchain, Challenges and solutions, proof of work, Proof of stake, alternatives to Bitcoin consensus, Bitcoin scripting language and their use
IV	Ethereum basics: Ethereum and Smart Contracts, The Turing Completeness of Smart Contract Languages and verification challenges, using smart contracts to enforce legal contracts, comparing Bitcoin scripting vs. Ethereum Smart Contracts, writing smart contracts using Solidity & JavaScript
V	Privacy, Security issues in Blockchain: Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Blockchains: Sybil attacks, selfish mining, 51% attacks advent of algorand; Sharding based consensus algorithms to prevent these attacks
VI	Case Studies: Block chain in Financial Service, Supply Chain Management and Government Services

List of References:

1. Narayanan, Bonneau, Felten, Miller and Goldfeder, “Bitcoin and Cryptocurrency Technologies – A Comprehensive Introduction”, Princeton University Press.
2. Josh Thompson, ‘Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming’, Create Space Independent Publishing Platform, 2017.
3. Imran Bashir, “Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained”, Packt Publishing.
4. Merunas Grincalaitis, “Mastering Ethereum: Implement Advanced Blockchain Applications Using Ethereum-supported Tools, Services, and Protocols”, Packt Publishing.
5. Prof. Sandip Chakraborty, Dr. Praveen Jayachandran, “Blockchain Architecture Design and Use Cases” [MOOC], NPTEL: <https://nptel.ac.in/courses/106/105/106105184/>
6. BLOCKCHAIN BASICS, Daniel Drescher, Apress Publication, 2017.
7. Blockchain, Melanie Swan, O'reilly Publication, 2015.

S. Y. M. Sc. Semester IV		
MTS-672	Emerging Technologies: Cloud Computing	Number of Credits: 02
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Understanding the Overview of Cloud Computing	
CO2	Learn the technology used for Cloud Computing Models	
CO3	Familiarize the primitives of service models	
CO4	Illustrate the concepts of Virtualization and Cloud Service Providers	
CO5	To understand concept of Data centre in cloud computing	
CO6	Understand use of services from the aspect of cloud service models and data centres.	

Unit No	Title of the Unit and Contents
I	Introduction to Cloud Computing: Overview, Characteristics, Architecture, how it works? Advantages/Disadvantages, Risks of Cloud Computing, Applications
II	Cloud Computing Deployment Models: Cloud Computing Deployment Models Types of Cloud (Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud, Multi-Cloud)
III	Cloud Computing Service Models: Software as a service (SaaS), Platform as a service (PaaS), Infrastructure as a service (IaaS), Difference between IaaS, PaaS, and SaaS
IV	Virtualization: What is Virtualization, what is Virtual Machine, what is Container, Data Virtualization, Hardware Virtualization, Software Virtualization, Server Virtualization, Storage Virtualization, OS Virtualization, Windows Virtualization, Demo (any one)
V	Cloud Service Providers: Cloud Service Provider Companies, Amazon Web Services (AWS), Microsoft Azure, Google Cloud (In brief)
VI	Data Centre in Cloud Computing: What is a Data Centre? How do Data Centres work? Why are data centres important? Main components of data centre, Data Centre vs. Cloud

Learning Resources:

1. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011
2. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
3. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010
4. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley- India, 2010

S. Y. M. Sc. Semester IV		
MTS-673	Emerging Technologies: Edge Computing	Number of Credits: 02
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Understanding the basics of cloud computing and Edge Computing	
CO2	Learn the Architecture in Edge Computing	
CO3	Familiarize the primitives of User applications in Edge Computing	
CO4	Illustrate the concepts of Distributed Systems in Edge Computing	
CO5	To understand edge data centres, services, and edge Containers.	
CO6	Understand use of Data Integration Platform and Use cases	

Unit No	Title of the Unit and Contents
I	Background of Edge Computing - Introduction to Cloud and its limitations to support low latency and RTT. From Cloud to Edge computing: Waves of innovation
II	Overview of Edge Computing, Architectures in Edge Computing
III	Supporting User Applications - Edge Computing to support User Applications (5G-Slicing, self-driving cars and more)
IV	Distributed Systems in Edge Computing: Concepts of distributed systems in edge computing such as time ordering and clock synchronization, distributed snapshot, etc.
V	Edge Data Centers and Services: Introduction to Edge Data Center, Lightweight Edge Clouds and its services provided by different service providers.
VI	Edge Containers: Introduction to docker container and Kubernetes in edge computing. Design of edge storage systems like key-value stores
VII	Data Integration Platforms: Introduction to MQTT and Kafka for end-to-end edge pipeline. Edge analytics topologies for M2M and WSN network (MQTT)
VIII	Use cases based on Machine Learning for edge sensor data in predictive maintenance, image classifier and self-driving cars.

Learning Resources:

1. "Fog and Edge Computing: Principles and Paradigms", Rajkumar Buyya (Editor), Satish Narayana Srirama (Editor), Wiley, 2019
2. Cloud Computing: Principles and Paradigms", Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011
3. "Cloud and Distributed Computing: Algorithms and Systems", Rajiv Misra, Yashwant Patel, Wiley 2020.

S. Y. M. Sc. Semester IV		
MTS-674	Industrial Training	Number of Credits :4
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Describe and develop the various skills, attitude, and knowledge to understand the professionalism in the IT industry.	
CO2	Discuss and explain the working culture of the Industry in view to maintain quality standards.	
CO3	Implement the confidence, presentation skills and logical thinking, communication skills in developing the system.	
CO4	Differentiate between the academics and professional work culture in timely delivery of projects.	
CO5	Compare and contrast the professional development of the programs and project.	
CO6	Combine the techniques to enhance oneself as a thorough software professional.	

The structure for the Industrial Training Project (ITP) will be as follows

A student can complete an Industrial Training Project (ITP) in any I.T. industry / academic institute / with a research project of a teacher / an expert funded by any funding agency for a minimum period of three months.

There will be a teacher coordinator mentoring a group of 10 students throughout the Semester. The teacher coordinator is expected to perform the following tasks

- Maintain a weekly status/progress report of the student. The student will report to the assigned student coordinator once a week either offline/ online mode regarding the progress of his/her work at the Industry/Academic Institute.
- Maintain contact with the Industry/Academic regarding the internship offered to the student.
- Help the student in solving difficulties
- Organize presentations and discussions as required
- Guide the student in preparing the Final Project Documentation
- Maintain a track record for each student through the semester
- Conduct an Internal Assessment for each student consisting of 50

Marks The work load for the teacher coordinator is proposed as four hours per week.

The workload for a teacher coordinator who is guiding 3 students doing their ITP in Fergusson College (Autonomous) Pune (no mentor from industry) is proposed as four hours per week.

S. Y. M. Sc. Semester IV		
MTS-675	On Job Training (OJT)	Number of Credits :4
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Describe and demonstrate the professional skills, attitudes, ethics, and knowledge required to function effectively in an IT industry environment.	
CO2	Explain and follow industry work culture, organizational practices, quality standards, workplace policies, and professional responsibilities.	
CO3	Apply technical knowledge, logical thinking, communication skills, teamwork, and presentation skills while working on assigned tasks and projects.	
CO4	Differentiate between academic learning and professional work practices with respect to project planning, task management, deadlines, and timely delivery.	
CO5	Analyze and evaluate professional approaches used in software development, project management, documentation, testing, and problem-solving within the organization.	
CO6	Develop professional competencies by integrating technical skills, workplace experience, communication, teamwork, adaptability, and continuous learning to enhance career readiness.	

OJT Guidelines [Click here](#)