Savitribai Phule Pune University, Pune

Maharashtra, India

Faculty of Science and Technology

Curriculum Structure and Syllabus

MCA (Under Engineering) (2025 Pattern)

Master of Computer Applications (Under Engineering)

(With effect from Academic Year 2025-26)

www.unipune.ac.in

Contents

Nomenclature

CCE	Comprehensive Continuous Evaluation
OJT	On-the-Job Training
PCC	Programme Core Course
PEC	Programme Elective Course
PEO	Programme Educational Objectives
PSO	Program Specific Outcomes
WK	Knowledge and Attitude Profile

Preface by Board of Studies

Dear Students and Teachers,

We, the members of Board of Studies Computer Engineering, are very happy to present First Year Master of Computer Applications (MCA) syllabus effective from the AY Year 2025-26. Subsequently this will be carried forward for Second Year in the AY 2026-27.

Master of Computer Applications (MCA) have emerged as transformative forces reshaping industries, driving innovation, and impacting our daily lives. Recognizing the growing importance and pervasive nature of these fields, we have designed this comprehensive syllabus to equip students with the foundational knowledge, practical skills. This curriculum is meticulously crafted to provide a holistic learning experience, blending theoretical concepts with hands-on applications. It aims to foster critical thinking, problem-solving abilities, enabling graduates to contribute meaningfully to the advancement and responsible deployment of Computer Applications. The revised syllabus falls in line with the objectives of NEP-2020, Savitribai Phule Pune University, AICTE New Delhi, UGC, and various accreditation agencies by keeping an eye on the technological developments, innovations, and industry requirements.

Learners are now getting sufficient time for self-learning either through online courses or additional projects for enhancing their knowledge and skill sets. Learners can be advised to take up online courses, on successful completion they are required to submit certification for the same. This will definitely help learners to facilitate their enhanced learning based on their interest. We believe that this well-structured and comprehensive syllabus will serve as a robust foundation for aspiring Computer Applications professionals, enabling them to contribute significantly to the technological progress and address the challenges of the 21st century.

We would like to place on record our gratefulness to the faculty, students, industry experts and stakeholders for having helped us in the formulation of this syllabus.

Dr. Nilesh Uke
Chairman - Board of Studies (Computer Engineering)
Savitribai Phule Pune University

Members of Board of Studies - Computer Engineering	
Dr. Pramod Patil	Dr. Dipti Patil
Dr. Dhananjay Kshirsagar	Dr. Amol Potgantwar
Dr. Sachin Babar	Dr. Balwant Sonkamble
Dr. Suhasini Itkar	Dr. Sachin Sakhare
Dr. Dipak Patil	Dr. Vandana Dhingra
Dr. Deepali Ujalambkar	Dr. Vaishali Vikhe
Dr. Pradip Jawandhiya	Dr. Sandeep Deshmukh

Program Specific Outcomes (PSO)

- **PSO1:** Demonstrate proficiency in essential concepts of computer science and programming solutions.
- **PSO2:** Formulate robust software design, execution, and testing strategies employing a software paradigms and Computer Application knowledge to solve real word problems.
- **PSO3:** Adapt and exhibit expertise in evolving areas of computer science and technology.

Programme Educational Objectives (PEO)

Program Educational Objectives (PEOs): Program Educational Objectives are broad statements that describe the career and professional accomplishments that the program is preparing graduates to achieve.

PEO	PEO Focus	PEO Statements
PEO1	Core competence	To prepare globally competent post graduates with enhanced domain knowledge and skills attaining professional excellence and updated with modern technology to provide effective solutions for engineering and research problems.
PEO2	Problem solving skills and Ethics	To prepare the post graduates to work as a committed professionals with strong professional ethics and values, sense of responsibilities, understanding of legal, safety, health, societal, cultural and environmental issues.
PEO3	Professionalism and Lifelong Learning	To prepare motivated post graduates with research attitude, lifelong learning, investigative approach, and multidisciplinary thinking to succeed in the career in industry/academia/research.
PEO4	Team Building	To prepare post graduates with strong managerial and communication skills to work effectively as an individual as well as in teams.

Knowledge and Attitude Profile (WK)

A Knowledge and Attitude Profile (KAP), often represented as WK (Knowledge and Attitude Profile) in some contexts, is a framework or assessment tool used to evaluate an individual's knowledge and attitudes related to a specific area, topic, or domain.

WK1	A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.
WK2	Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.
WK3	A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.
WK4	Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.
WK5	Knowledge, including efficient resource use, environmental impacts, whole-life cost, re-use of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.
WK6	Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.
WK7	Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.
WK8	Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.
WK9	Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

Programme Outcomes (PO)

POs are statements that describe what students are expected to know and be able to do upon graduating from the program. These relate to the skills, knowledge, analytical ability attitude and behaviour that students acquire through the program. The POs essentially indicates what the students can do from course-wise knowledge acquired by them during the program. As such, POs define the professional profile of a graduate of MCA Program.

NBA has defined the following eight POs for a graduate of MCA Program:

PO1	Foundation Knowledge	Apply knowledge of mathematics, programming logic and coding fundamentals for solution architecture and problem solving
PO2	Problem analysis	Identify, review, formulate and analyze problems for primarily focusing on customer requirements using critical thinking frameworks.
PO3	Development of Solutions	Design, develop and investigate problems with as an innovative approach for solution incorporating ESG/DSG goals
PO4	Modern Tool Usage	Select, adapt and apply modern computational tools such as development of algorithms with an understanding of the limitations including human biases.
PO5	Individual and Teamwork	Function and communicate effectively as an individual or a team leader in diverse and multidisciplinary groups. Use methodologies such as agile.
PO6	Project Management and Finance	Use the principles of project management such as scheduling, work breakdown structure and be conversant with the principles of Finance for profitable project management.
PO7	Ethics	Commit to professional ethics in managing software projects with financial aspects. Learn to use new technologies for cyber security and insulate customers from malware
PO8	Life-long learning	Change management skills and the ability to learn, keep up with contemporary technologies and ways of working.

General Rules and Guidelines

- **Course Outcomes (CO):** Course Outcomes are narrower statements that describe what students are expected to know, and are able to do at the end of each course. These relate to the skills, knowledge and behaviour that students acquire in their progress through the course.
- **Assessment:** Assessment is one or more processes, carried out by the institution, that identify, collect, and prepare data to evaluate the achievement of Program Educational Objectives and Program Outcomes.
- **Evaluation:** Evaluation is one or more processes, done by the Evaluation Team, for interpreting the data and evidence accumulated through assessment practices. Evaluation determines the extent to which Program Educational Objectives or Program Outcomes are being achieved, and results in decisions and actions to improve the program.

Guidelines for Examination Scheme

Theory Examination: The theory examination shall be conducted in two different parts Comprehensive Continuous Evaluation (CCE) and End-Semester Examination (ESE).

Comprehensive Continuous Evaluation (CCE) of 50 marks based on all the Units of course syllabus to be scheduled and conducted at institute level. To design a Comprehensive Continuous Evaluation (CCE) scheme for a theory subject of 50 marks with the specified parameters, the allocation of marks and the structure can be detailed as follows:

Sr.	Parameters	Marks	Coverage of Units
1	Unit Test	20 Marks	Units 1 & Unit 2 (10 Marks/Unit)
2	Assignments / Case Study	20 Marks	Units 3 & Unit 4 (10 Marks/Unit)
3	Seminar Presentation / Open Book Test/ Quiz	10 Marks	Unit 5

Format and Implementation of Comprehensive Continuous Evaluation (CCE)

- **Unit Test:**
  - **Format :** Questions designed as per Bloom's Taxonomy guidelines to assess various cognitive levels (Remember, Understand, Apply, Analyze, Evaluate, Create).
  - **Implementation**: Schedule the test after completing Units 1 and 2. Ensure the question paper is balanced and covers key concepts and applications.
- **Sample Question Distribution:**
  - **Remembering (4 Marks):** Define key terms related to [Topic from Units 1 and 2].
  - **Understanding (4 Marks):** Explain the principle of [Concept] in [Context].
  - **Applying (4 Marks):** Demonstrate how [Concept] can be used in [Scenario].
  - **Analyzing (4 Marks):** Compare & contrast [Two related concepts] from Units 1 and 2.

- **Evaluating (4 Marks):** Evaluate the effectiveness of [Theory/Model] in [Situation].

- **Assignments / Case Study:** Students should submit one assignment or one Case Study Report based on Unit 3 and one assignment or one Case Study Report based on Unit 4.
  - **Format:** Problem-solving tasks, theoretical questions, practical exercises, or case studies that require in-depth analysis and application of concepts.
  - **Implementation:** Distribute the assignments or case study after covering Units 3 and 4. Provide clear guidelines and a rubric for evaluation.

- **Seminar Presentation:**
  - **Format:** Oral presentation on a topic from Unit 5, followed by a Q&A session.
  - **Deliverables:** Presentation slides, a summary report in 2 to 3 pages, and performance during the presentation.
  - **Implementation:** Schedule the seminar presentations towards the end of the course. Provide students with ample time to prepare and offer guidance on presentation skills.

- **Open Book Test:**
  - **Format:** Analytical and application-based questions to assess depth of understanding.
  - **Implementation:** Schedule the open book test towards the end of the course, ensuring it covers critical aspects of Unit 5.

- **Quiz:**
  - **Format:** Quizzes can help your students practice existing knowledge while stimulating interest in learning about new topic in that course. You can set your quizzes to be completed individually or in small groups.
  - **Implementation:** Online tools and software can be used create quiz. Each quiz is made up of a variety of question types including multiple choice, missing words, true or false etc.

- **Term Paper/Conference Paper Publication**
  - **Format:** Prepare a research paper on a course-related topic in IEEE format (Abstract, Introduction, Methodology, Results, Conclusion, References). Aims to build research and writing skills.
  - **Implementation:** Students will choose a topic (with faculty approval) and conduct literature review, data collection/analysis, and structured writing as per IEEE guidelines. Papers can be prepared individually or in pairs. Final submissions is evaluated internally and submitted to reputed conferences or journals for publication consideration.

- **Example Timeline for conducting CCE:**
  - Weeks 1-4 : Cover Units 1 and 2

- Week 5 : Conduct Unit Test (20 marks)
- Weeks 6-8 : Cover Units 3 and 4
- Week 9 : Distribute and collect Assignments / Case Study (20 marks)
- Weeks 10-12 : Cover Unit 5
- Week 13 : Conduct Seminar Presentations or Open Book Test or Quiz (10 marks)

- **Evaluation and Feedback:**
  - **Unit Test:** Evaluate promptly and provide constructive feedback on strengths and areas for improvement.
  - **Assignments / Case Study:** Assess the quality of submissions based on the provided rubric. Offer feedback to help students understand their performance.
  - **Seminar Presentation:** Evaluate based on content, delivery, and engagement during the Q&A session. Provide feedback on presentation skills and comprehension of the topic.
  - **Open Book Test**: Evaluate based on the depth of analysis and application of concepts. Provide feedback on critical thinking and problem-solving skills.

By following this scheme, you can ensure a structured and comprehensive evaluation of students' understanding and application of the course material, adhering to Bloom's Taxonomy guidelines for cognitive skills evaluation.

End-Semester Examination (ESE)

End-Semester Examination (ESE) of 50 marks written theory examination based on all the unit of course syllabus scheduled by university. Question papers will be sent by the University through QPD (Question Paper Delivery). University will schedule and conduct ESE at the end of the semester.

Format and Implementation:

- **Question Paper Design** : Below structure is to be followed to design an End-Semester Examination (ESE) for a theory subject of 50 marks on all 5 units of the syllabus with questions set as per Bloom's Taxonomy guidelines and 10 marks allocated per unit.
- **Balanced Coverage**: Ensure balanced coverage of all units with questions that assess different cognitive levels of Bloom's Taxonomy: Remember, Understand, Apply, Analyze, Evaluate, and Create. The questions should be structured to cover:
  - Remembering: Basic recall of facts and concepts.
  - Understanding: Explanation of ideas or concepts.
  - Applying: Use of information in new situations.
  - Analyzing: Drawing connections among ideas.
  - Evaluating: Justifying a decision or course of action.
  - Creating: Producing new or original work (if applicable).

- **Detailed Scheme:** Unit-Wise Allocation (10 Marks per Unit): Each unit will have a combination of questions designed to assess different cognitive levels. By following this scheme, you can ensure a comprehensive and fair assessment of students' understanding and application of the course material, adhering to Bloom's Taxonomy guidelines for cognitive skills evaluation.

End Semester Examination Guidelines:

Question Paper :

One section with five questions (10 marks each). Each has an alternate from the same unit.

Framing of questions should be according to Anderson/Revised Bloom's Taxonomy and disseminated through the question papers with a mention of course outcomes as well.

Assessment

Assessment will be done at the centralized assessment programme (CAP) Centre of the College by the Expert who is appointed as an examiner for the courses as per 48(3) panel of Maharashtra public university act 2016.

Moderation will be done at the CAP Centre designated by the University by the Expert who is appointed as an examiner for the subject as per 48(3) panel.

Monitoring of CCE

Periodic monitoring of CCE will be done by respecting board of studies members, whenever required.

Colleges should maintain all the records related to CCE with CEO for verification.

Staff members should keep all the records of Unit Tests, Assignments / Case Study, Seminar Presentation / Open Book Test/ Quiz duly signed by head of the institute with College Examination officer.

General Guidelines for Laboratory Subject

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a hands-on resource and reference. The instructor's manual needs to include prologue (about University/program/ institute/ department/foreword/ preface etc), copy of curriculum, conduction & Assessment guidelines, topics under consideration-concept,objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Lab Journal

The laboratory assignments are to be submitted by students in the form of a journal. Journal consists of prologue, Certificate, table of contents, and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software & Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory-Concept in brief, features of tool/framework/language used, Design, test cases, conclusion. Program codes with sample output of all performed assignments are to be submitted as softcopy.

As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journals may be avoided. Use of Drive containing students programs maintained by lab In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory.

Guidelines for Lab /TW Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of students. Each lab assignment assessment will assign grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Laboratory Conduction

List of laboratory assignments is provided below for reference. The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy should address the average students and inclusive of an element to attract and promote the intelligent students. The instructor may set multiple sets of assignments and distribute them among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of coding style, proper indentation and comments.

Use of open source software and recent versions is to be encouraged.

In addition to these, instructors may assign one real life application in the form of a mini-project.

Based on the concepts learned. Instructors may also set one assignment or mini-project that is suitable to each branch beyond the scope of the syllabus.

First Year MCA (Under Engineering) (2025 Pattern) – Master of Computer Applications

Curriculum Structure - Semester I

Course Code	Course Type	Course Name	Teaching Scheme			Examination Scheme and Marks						Credits			
			Theory	Tutorial	Practical	CCE*	End-Sem	Term Work	Practical	Oral/ Present.*	Total	Theory	Tutorial	Practical	Total
PCC-501-MCA	Programme Core Course	Mathematical foundation for Computer Applications	3	-	-	50	50	-	-	-	100	3	-	-	3
PCC-502-MCA	Programme Core Course	Data Structures and Algorithms	3	-	-	50	50	-	-	-	100	3	-	-	3
PCC-503-MCA	Programme Core Course	Object Oriented Programming	3	-	-	50	50	-	-	-	100	3	-	-	3
RM-530-MCA	RM	Research Methodology	3	-	-	50	50	-	-	-	100	3	-	-	3
PEC-520-MCA	Programme Elective Course (Elective –I)	Elective I	3	-	-	50	50	-	-	-	100	3	-	-	3
PCC-504-MCA	Programme Core Course	Data Structures Laboratory	-	-	4	-	-	50	50	-	100	-	-	2	2
PCC-505-MCA	Programme Core Course	Object Oriented Programming Laboratory	-	-	4	-	-	25	25	-	50	-	-	2	2
PEC-521-MCA	Programme Elective Course	Elective-I Laboratory	-	-	2	-	-	25	-	25	50	-	-	1	1
PCC-506-MCA	Programme Core Course	Python Programming Laboratory	-	1	2	-	-	50	-	-	50	-	1	1	2
Total			15	1	12	250	250	150	75	25	750	15	1	6	22

List of Elective I Courses:

PEC-520A-MCA	Artificial Intelligence
PEC-520B-MCA	Mobile Computing
PEC-520C-MCA	Data Analytics

CCE*: Comprehensive Continuous Evaluation

Present.*: Presentation

First Year MCA (Under Engineering) (2025 Pattern) – Master of Computer Applications

Curriculum Structure - Semester II

Course Code	Course Type	Course Name	Teaching Scheme			Examination Scheme and Marks						Credits			
			Theory	Tutorial	Practical	CCE*	End-Sem	Term Work	Practical	Oral/ Present.*	Total	Theory	Tutorial	Practical	Total
PCC-551-MCA	Programme Core Course	Operating System and Network Fundamentals	3	-	-	50	50	-	-	-	100	3	-	-	3
PCC-552-MCA	Programme Core Course	Database Management System	3	-	-	50	50	-	-	-	100	3	-	-	3
PCC-553-MCA	Programme Core Course	Software Engineering and Project Management	3	-	-	50	50	-	-	-	100	3	-	-	3
PCC-554-MCA	Programme Core Course	Java and Advance Java Programming	3	-	-	50	50	-	-	-	100	3	-	-	3
PEC-570-MCA	Programme Elective Course (Elective –II)	Elective II	3		-	50	50	-	-	-	100	3	-	-	3
PCC-555-MCA	Programme Core Course	Java Programming Laboratory	-	-	4	-	-	25	50	-	75	-	-	2	2
PCC-556-MCA	Programme Core Course	Database Laboratory	-	-	2	-	-	-	-	50	50	-	-	1	1
PCC-557-MCA	Programme Core Course	Project Based Learning	-	-	2	-	-	25	-	-	25	-	-	1	1
OJT-581-MCA	Internship/OJT (IN/OJT)	Internship	-	-	6	-	-	50	-	50	100	-	-	3	3
Total			15	-	14	250	250	100	50	100	750	15	-	7	22

List of Elective II Courses:

PEC-570A-MCA	Machine Learning
PEC-570B-MCA	Internet of Things
PEC-570C-MCA	Data Mining and Data Warehousing

CCE*: Comprehensive Continuous Evaluation

Present.*: Presentation

Savitribai Phule Pune University, Pune

Maharashtra, India

Master of Computer Applications (Under Engineering) (2025 Pattern)

Semester I

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-501-MCA : Mathematical Foundation for Computer Applications		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses: Students must possess a clear knowledge of all the Fundamentals of Mathematics.

Companion Course: NA

Course Objectives: The course aims to:

1. To study discrete objects and relationships among them.
2. To use appropriate set, function and relation models to understand practical examples, and interpret the associated operations and terminologies in context.
3. To determine number of logical possibilities of events.
4. To learn basic ideas of statistical inference and random variables.
5. To study basics of graph and tree concepts and its operations

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Solve** mathematical problems using set and Induction methods.
- **CO2: Find** relations and functions between sets using various techniques.
- **CO3: Examine** outcomes for computational processes using permutation and combinations.
- **CO4: Predict** the occurrence of events using statistical concepts
- **CO5: Apply** graph theory and tree concept for real world applications.

Course Contents

Unit I - Set Theory and Logic - (09 Hours)

Set Theory: Basic concepts of set theory, Types of sets –Countable and Uncountable Sets, Finite and Infinite Sets, Countably Infinite and Uncountably Infinite Sets, Power set. Multi set, Operations on set, Principle of inclusion and exclusion, Venn diagram.

Propositional Logic: Logic, Propositional equivalences, Application of Propositional Logic-Translating English Sentences, Proof by Mathematical Induction

 # Exemplar /Case Studies: Familiarity with some basic mathematical objects, notions & logic.

Unit II - Relations and Functions - (09 Hours)

Relations: Relation and their properties, Representing Relations, Closures of Relations, Equivalence Relations, POSET, HASSE Diagram, Lattices-Chains and Anti-chains, Transitive Closure and Warshalls Algorithm.

Functions: Subjective, Injective and Bijective functions, Inverse Functions and composition of functions, The Pigeonhole Principle.

#Exemplar/Case Studies: Understanding relations, function and their types with best examples.

Unit III - Probability and Combinatorics - (09 Hours)

Probability : Introduction to probability, sample space and events, Axioms of probability, conditional probability, Bayes Theorem.

Permutations and Combinations : The Basics of Counting, rule of Sum and Product, Generalized Permutations and Combinations. Binomial Coefficients Identities and Pascal Triangle.

#Exemplar/Case Studies: Expected to have a strong foundation for probabilities and permutation, combination.

Unit IV - Descriptive Statistics and Discrete Distribution - (09 Hours)

Sampling : Types of Sampling, Random Sampling Frequency distributions: Mean, Median, Mode, Variance and Standard Deviation for Grouped and Ungrouped Data. Co-relation, Regression and two regression equations (Regression Line of X on Y, Regression Line of Y on X).

Discrete Distribution : Geometric Distribution, Binomial Distribution, Poisson Distribution, Introduction to Hypothesis Testing.

#Exemplar/Case Studies : Ability to calculate basic statistical measures like mean, median, mode, standard deviation, and range.

Unit V - Tree and Graph Theory - (09 Hours)

Tree : Introduction and properties of trees, rooted and binary trees, Spanning Tree.

Graph Theory : Applications of graph, finite and infinite graphs, Matrix Representation, Isolated vertex, pendant vertex, and null graph, Isomorphism, Subgraphs, Euler graphs, Hamiltonian paths and circuits, Minimum spanning tree algorithms, Prim and Kruskal's algorithm.

#Exemplar/Case Studies : Familiarity with Tree and Graph application

Learning Resources

Text Books:

1. Kenneth H Rosen, "Discrete Mathematics & its Applications" 7th Edition, McGraw-Hill, 2010.
2. Jayant Ganguly: A Treatise on Discrete Mathematical Structures", Pearson Education, 2010.
3. Narasingh Deo , "Graph Theory with applications to engineering and computer Science, Prentice Hall of India, 1990, 0 – 87692 –145 –4.

Reference Books:

1. Bernard Kolman, Robert C. Busby and Sharon Ross, “Discrete Mathematical Structures”, Prentice Hall of India Pearson, ISBN: 0132078457, 9780132078450.

2. N. Biggs, “Discrete Mathematics”, 3rd Edition, Oxford University Press, ISBN 0 –19 850717 –8.

3. Dr. K. D. Joshi, “Foundations of Discrete Mathematics”, New Age International Limited, Publishers, January 1996, ISBN: 8122408265, 9788122408263

4. Papoulis, Pillai, “Probability, Random Variables and Stochastic Processes”, 4th Edition ISBN:0-07- 048658-1.

5. Veerarajanl, “Probability, Statistics And Random”, Tata McGraw-Hill, ISBN:0-07-049482-7

6. S.C. Gupta, V.K Kapoor, S. Chand, “Fundamentals of Mathematical Statistics”, Sultan Chand & Sons, ISBN 10: 8180545288.

7. D .P. Apte , “Probability and Combinatorics”, Excel Books, ISBN-13: 978-8174465207

e-Books:

1. https://www.cis.upenn.edu/∼jean/discmath-root-b.pdf

2. https://www.cs.yale.edu/homes/aspnes/classes/202/notes.pdf

3. http://discrete.openmathbooks.org/pdfs/dmoi-tablet.pdf

MOOC Courses:

1. https://www.coursera.org/specializations/discrete-mathematics

2. https://www.coursera.org/learn/discrete-mathematics

3. https://www.my-mooc.com/en/categorie/statistics-and-probability

4. https://www.edx.org/learn/statistics

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-502-MCA : Data Structures and Algorithms		
Course Name: Data Structures and Algorithms	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses: Basics of Computers .

Companion Course: Data Structures Laboratory

Course Objectives: The course aims to:

1. To introduce the fundamental concepts of data structures, algorithms, and their applications in problem-solving.
2. To enable students to implement various data structures such as arrays, linked lists, stacks, queues, trees, and graphs.
3. To develop the ability to synthesize and analyze algorithms.
4. To equip students with the knowledge to use appropriate data structures and algorithms in software development and optimization.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Understand** and **Apply** basic concepts of data structures and algorithms to solve computational problems.
- **CO2: Analyze** the behavior and use of linear data structures such as arrays and linked lists in problem-solving.
- **CO3: Apply** stacks and queues using arrays and linked lists to solve computational problems like expression evaluation and task scheduling.
- **CO4: Analyze** the structure and operations of trees and graphs, including binary trees, binary search trees, and graph traversal techniques.
- **CO5: Compare** and **analyze** different searching and sorting algorithms to evaluate their efficiency and effectiveness.

Course Contents

Unit I - Introduction to Algorithms and Data Structures - (09 Hours)

Analysis of Algorithms: Algorithm, Complexity of an Algorithm, Asymptotic Notations, Big O Notation, Big Omega Notation, Big Theta Notation, Rate of Growth and Big O Notation.

Introduction to data Structures: Data Structures, Classification of Data Structures, Primitive Data Types, Abstract Data Types.

Arrays: Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Traversing, Insertion, Deletion, Multidimensional Arrays, Memory Representation of Two Dimensional Arrays, General Multi-Dimensional Arrays, Sparse Arrays, Sparse Matrix, Memory Representation of Special kind of Matrices, Advantages and Limitations of Arrays.

#Exemplar/Case Studies: Student Records Management System

Unit II - Linked list - (09 Hours)

Linked list: introduction to Linked List, Dynamic memory allocation, Singly Linked List operations: Create, Traversing, Searching, Insertion node, Deletion node, Merging Two Linked Lists, Reversing linked List.

Doubly Linked List: Definition and structure of DLL, DLL Operations: Insertion: Insert at the beginning/end/ after a given node/before a given node Deletion: Delete from beginning/ end/ a specific node by value. Traversal:Forward traversal (head to last node), Backward traversal (last node to head), Searching: Search for a specific element. Updating:Modify data of a node.

Circular Linked List : Definition and types. Applications of SLL- Polynomial representation.

#Exemplar/Case Studies: Contact Management System : linked list-based

Unit III - Stacks and Queues - (09 Hours)

Stack: Introduction, Stack as ADT- operations on stack, Array Representation of Stack. Applications- Matching Parenthesis, Recursion.

Queue: Introduction, Queue as ADT- operations on queue, Array and Linked List Representation of Queue, Basics of special types of queues: Circular Queue, Deque , Priority Queue. Applications of Queue - Job Scheduling, CPU Scheduling (Round Robin scheduling), Print Spooling (managing multiple print jobs), Call Center Systems (customer call management).

#Exemplar/Case Studies: Web Browser Navigation System: Study back and forward navigation feature for a web browser using Stacks and Queues.

Unit IV - Trees and Graphs - (09 Hours)

Trees : Concept of nonlinear data structure,Types of Trees: Binary Tree, Full Binary Tree, Complete Binary Tree, Skewed Binary Tree, Balanced Trees (like AVL Trees - brief introduction).

Trees and binary trees-concept and terminology, Sequential & Linked representation of binary trees, Binary Tree Traversals: Recursive Traversal Algorithms: Inorder, Preorder, Postorder. Non-Recursive Traversal Algorithms:Using Stack for Inorder, Preorder, and Postorder Traversals and Level Order Traversal using Queue (Breadth-First Traversal), , Binary search trees (BST) , BST operations.

Graphs: Definition and Basic Terminologies: Representation of graph - Adjacency matrix and Adjacency list and Edge List Representation , Types of Graphs, Graph traversals- Depth First Search (DFS) : Recursive and Non-Recursive approaches, Breadth First Search (BSF): Using Queue.

#Exemplar/Case Studies: Social Network Friend Recommendation System

Unit V - Searching, Sorting and Hashing - (09 Hours)

Searching: Search Techniques, Sequential search, Binary search.

Sorting: Types of sorting-Internal and external sorting, General sort concepts-sort order, stability, efficiency, number of passes, Sorting methods- Bubble sort, Insertion sort, Selection sort, Quick sort, Comparison of All Sorting Methods

Hashing: Hash Tables, Hash function: Division, Folding, Mid-Square function. Collision Resolution: Open Addressing, Chaining.

#Exemplar/Case Studies: Airline Reservation System : search and sort flight reservations

Learning Resources

Text Books:

1. Aho A., Hopcroft J., Ulman J., "Data Structures and Algorithms", Pearson Education, ISBN-0201- 43578-02.
2. Brassard & Bratley, "Fundamentals of Algorithmics", Prentice Hall India/Pearson Education, ISBN 13-9788120311312.
3. Weiss, M. A., "Data Structures and Algorithm Analysis in C++ ", (4th ed.). Pearson. ISBN-13: 978- 0-13-284737-7. (2013)

Reference Books:

1. Goodrich, Tamassia, and Goldwasser, "Data Structures and Algorithms in Java," Wiley, ISBN-10: 1118771338, ISBN-13: 9781118771334.
2. Lafore, "Data Structures and Algorithms in Java," Sams Publishing, ISBN-10: 0672324539, ISBN- 13: 9780672324536.
3. Horowitz and Sahani, "Fundamentals of Data Structures in C++", University Press, ISBN 10:0716782928 ISBN 13: 9780716782926.2.
4. Drozdek, "Data Structures and Algorithms in Java", 2nd ed. Boston, MA: Course Technology, 2004.
5. Yedidyah Langsam, Moshe J Augenstein, Aron M Tenenbaum, "Data Structures using C and C++",Pearson Education, ISBN 81-317-0328-2.3.
6. Trembley, J.P. and Sorenson P.G., "An Introduction to Data Structures with Applications",McGrawHill

e-Books:

1. https://eprints.triatmamulya.ac.id/1694/1/Data%20Structures%20and%20Algorithms%20Made%20Easy_%20Data%20Structures%20and%20Algorithmic%20Puzzles.pdf
2. https://mimoza.marmara.edu.tr/~msakalli/cse706_12/SkienaTheAlgorithmDesignManual.pdf
3. https://dahlan.unimal.ac.id/files/ebooks/2013%20Algorithms_Unlocked.pdf
4. https://dl.ebooksworld.ir/books/Introduction.to.Algorithms.4th.Leiserson.Stein.Rivest.Cormen.MIT.Press.9780262046305.EBooksWorld.ir.pdf

MOOC Courses:

1. https://nptel.ac.in/courses/106106133
2. https://nptel.ac.in/courses/106106127
3. https://nptel.ac.in/courses/106105085
4. https://nptel.ac.in/courses/106102064

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-503-MCA : Object Oriented Programming		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses: Basics of programming languages

Companion Course: Object-Oriented Programming Laboratory

Course Objectives: The course aims to:

1. To introduce the fundamental concepts and principles of Object-Oriented Programming (OOP).
2. To provide a clear understanding of key Object-Oriented Programming features such as data abstraction, encapsulation, inheritance, polymorphism, and dynamic binding.
3. To enable students to apply the object-oriented approach in software design and development.
4. To enhance problem-solving skills through the development of programs using Object-Oriented Programming techniques in C++.
5. To build a strong foundation for advanced programming and software engineering topics by fostering structured and modular coding practices.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Explain** the fundamental principles of Object-Oriented Programming (OOP) such as abstraction, encapsulation, inheritance, and polymorphism.
- **CO2: Apply** object-oriented programming concepts to solve basic computational problems using C++.
- **CO3: Design** simple software systems using the object-oriented paradigm with appropriate use of classes and objects.
- **CO4: Analyze** the effectiveness of object-oriented constructs in enhancing program modularity, readability, and maintainability.
- **CO5: Develop** C++ programs that demonstrate the use of inheritance, polymorphism, and dynamic binding in solving real-world problems.

Course Contents

Unit I - Introduction to C++ and OOP Concepts - (09 Hours)

Introduction to OOP: Encapsulation, Abstraction, Inheritance, Polymorphism, Procedural vs Object-Oriented Programming ,Basic Syntax, Data Types, Operators in C++, Manipulators in C++,Control Structures :Conditional Statements, Loops, Functions in C++: Call by value and Call by reference, Inline, Default Arguments

 # Exemplar /Case Studies: Banking System Overview.

Unit II - Classes and Objects - (09 Hours)

Defining Classes and Objects, Access Specifiers: Public, Private, Protected, Scope resolution operator. Constructors: Types of constructor. and Destructors, Member Functions and Static Members, Friend Functions and Friend Classes, Dynamic Memory Allocation :new, delete & this Pointer.

#Exemplar/Case Studies: Student Management System.

Unit III - Inheritance and Polymorphism- (09 Hours)

Types of Inheritance: Single, Multiple, Multilevel, Hierarchical, Hybrid, Virtual Base Classes and Abstract Classes, Function Overriding, Compile-time Polymorphism: Function & Operator Overloading, Run-time Polymorphism: Virtual Functions & Pure Virtual Functions

#Exemplar/Case Studies: Employee Payroll System.

Unit IV - Templates and Exception Handling - (09 Hours)

Function Templates and Class Templates, Standard Template Library (STL): Vectors, Lists, Maps, Exception Handling: try, catch, throw, Handling Multiple Exceptions & User-defined Exceptions.

#Exemplar/Case Studies : Generic Sorting System.

Unit V - File Handling and Advanced Topics - (09 Hours)

File Streams : ifstream, ofstream, fstream, Reading and Writing to Files, File Modes and Error Handling, Command Line Arguments, Object-Oriented Design Principles & C++ Standard Library

#Exemplar/Case Studies : Library Management System

Learning Resources

Text Books:

1. Bjarne Stroustrup, “The C++ Programming language”, Third edition, Pearson Education. ISBN 9780201889543.
2. E.Balagurusamy,Programming with C++,Tata McGraw Hill, 3rd Edition ISBN-10: 0070669074
3. Deitel, “C++ How to Program”, 4thEdition, Pearson Education, ISBN:81-297-0276-2

Reference Books:

1. Robert Lafore, “Object-Oriented Programming in C++”, fourth edition, Sams Publishing, ISBN:067232 (ISBN 13: 9780672323089)
2. E. Balgurusamy, “Object oriented programming in C++ “, Tata McGraw Hill, ISBN: 9780071072830,

3. Herbert Schildt, “C++ The complete reference”, Eighth Edition, McGraw Hill Professional, ISBN:978-00-72226805

4. Matt Weisfeld, “The Object-Oriented Thought Process”, Third Edition, Pearson ISBN-13:075-2063330166

5. Cox Brad, Andrew J. Novobilski, “Object Oriented Programming: An Evolutionary Approach”, Second Edition, Addison–Wesley, ISBN: 13:978-020-1548341.

e-Books:

1. http://www.freebookcentre.net/Language/Free-C++-Books-Download.html
2. https://fac.ksu.edu.sa/sites/default/files/ObjectOrientedProgramminginC4thEdition.pdf
3. https://www.pdfdrive.com/c-object-oriented-programming-e39562275.html

MOOC Courses:

1. https://nptel.ac.in/courses/106/101/106101208/
2. https://nptel.ac.in/courses/106/105/106105151/
3. https://swayam.gov.in/nd1_noc20_cs53/preview
4. https://swayam.gov.in/nd1_noc20_cs07/preview

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
RM-530-MCA : Research Methodology		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Knowledge of concepts in computer science
- Knowledge of Introductory statistics

Companion Course: NA

Course Objectives: The course aims to:

1. Explain fundamental research concepts and principles.
2. Develop skills to design and conduct research studies.
3. Learn to conduct thorough literature reviews and evaluate existing research.
4. Gain knowledge of various research designs and methodologies.
5. Acquire data collection and analysis skills using appropriate tools.
6. Understand and apply ethical considerations in research.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Understanding** the basic concepts, purposes, and significance of research methodology in academic and professional contexts.
- **CO2: Apply** various research designs and their appropriateness for different types of research Problems.
- **CO3: Apply** suitable data collection and sampling methods to gather reliable and valid data for research studies.
- **CO4: Use** appropriate statistical tools and techniques to demonstrate research data and interpret the results effectively.
- **CO5: Apply** skills in writing clear, coherent, and well-structured research reports that effectively communicate research findings

Course Contents

Unit I - Introduction to Research Methodology - (09 Hours)

Meaning of Research, Definition and Importance of Research in academic and industrial contexts, Objectives of Research, Motivation in Research, Types of Research (Basic vs. applied research, Qualitative vs. quantitative research, Cross-sectional vs. longitudinal research), Research Methods versus Methodology, Criteria of Good Research, Research Process and Steps, Identifying the problem, Literature review, Research questions, research methods in computer application.

 # Exemplar /Case Studies: A survey-based study that evaluates teaching strategies in relation to student achievement.

Unit II - Research Design - (09 Hours)

Meaning of Research Design, Need for Research Design, importance of a Good Design, Basic Principles of Experimental Designs, Types of Research Designs: Exploratory design, Descriptive design, Experimental design, Quasi-experimental design, Components of Research Design: Objectives, Hypotheses, Variables, Methods of data collection, Sampling design, Validity and Reliability: Internal validity, External validity.

#Exemplar/Case Studies: Study on Impact of Climate Change on Agriculture.

Unit III - Data Collection and sampling methods - (09 Hours)

Data Collection: Primary and Secondary Data; Primary and Secondary Data Sources; Data Collection Methods Data collection techniques: surveys, interviews, observations, etc.

Sampling: Concept of sampling and sampling methods – sampling frame, sample, characteristics of good sample, simple random sampling, purposive sampling, convenience sampling, snowball sampling, Sample Size, Sampling Techniques or Methods, Choice of Sampling Techniques.

#Exemplar/Case Studies: Analyzing Mental Health and Screen Time Correlation of students.

Unit IV - Data Analysis and Research Ethics - (09 Hours)

Data Analysis: Introduction to Data Analysis, Descriptive Statistics, Inferential Statistics, Qualitative vs Quantitative data analysis, Regression Analysis and Correlation, Data Visualization Techniques.

Research Ethics: Importance of Ethics in research, Principals of Ethical Research, Plagiarism and Copyright Infringement, intellectual property rights and patent law, scholarly publishing- IMRAD concept

and Design of research paper, citation and acknowledgement.

#Exemplar/Case Studies : Sentiment Analysis of Online Product Reviews Evaluating customer feedback using AI.

Unit V - Research Report Writing - (09 Hours)

Research Report: Definition of Research Report, Importance of Research Report in Academia and Industry, Types of Research Report: Technical Report, Thesis/Dissertation, Journal/Conference Research Paper, Case Study, Components of Research Report, Ethical Consideration in Research reporting. Structure and organization of Research Report: Title, Abstract, keyword, Introduction, Literature review, Methodology, Result and Discussion, References, Formatting and Citation (APA, IEEE, MLA). Finalizing and Reviewing the Research Report: Proofreading (grammar, coherence, and consistency),

Plagiarism

check, Formatting according to journal or university guidelines, Peer review, self-assessment.

#Exemplar/Case Studies : Networks Researching next-generation wireless technology advancements - Future of 6G Communication.

Learning Resources

Text Books:

1. C. R. Kothari, “Research methodology: Methods and Techniques”, 3rd Edn.,New age International, ISBN 978-81-224-1522-3 2014.
2. R. Kumar, “Research methodology a step-by-step guide for beginners”, Sage Publications, London, ISBN 978-184920449-3, 2011.
3. C.G. Thomas, “Research methodology and scientific writing, Ane books”, Delhi, ISBN 978-9-388- 26448-8, 2015.
4. P. Laake, H. B. Benestad and B. R. Olsen, “Research methodology in the medical and biological sciences”, Academic Press, ISBN 978-0-12-373874-5, 2007.
5. H. J. Ader and G. J. Mellenbergh, “Research Methodology in the Social, Behavioural and Life Sciences Designs, Models and Methods”,3rd Edn., Sage Publications, London, ISBN-13: 978-0761958840, 2000.

Reference Books:

1. Ranjit Kumar,"Research Methodology: A Step-by-Step Guide for Beginners" , 4th edition, ISBN-13: 9781446269978, 2014
2. John W. Creswell and Vicki L. Plano Clark ,"Designing and Conducting Mixed Methods Research", 3rd Edition, ISBN-13: 978-1483344379, 2017.
3. Christopher M. Bishop,"Pattern Recognition and Machine Learning", ISBN-10: 0387310738, Springer 2006
4. Foster Provost and Tom Fawcett "Data Science for Business", ISBN-10: 1449361323, O’Reilly Media, 2013
5. Trevor Hastie, Robert Tibshirani, and Jerome Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction" , 2nd edition, ISBN-13: 978-0387848570, Springer, 2009.
6. Tamara Munzner, "Visualization Analysis and Design" , ISBN-10: 1466508914, A K Peters/CRC Press, 2014.

e-Books:

1. http://dl.saintgits.org/jspui/bitstream/123456789/1133/1/Research%20Methodology%20C%20R%2ri%20(Eng)%201.81%20MB.pdf

MOOC Courses:

1. http://kcl.digimat.in/nptel/courses/video/127106227/L13.html
2. http://kcl.digimat.in/nptel/courses/video/127106227/L28.html

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PEC-520A-MCA : Elective I-Artificial Intelligence		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Basic programming knowledge (Python/C++)
- Fundamental concepts of logic and probability

Companion Course: Elective laboratory

Course Objectives: The course aims to:

1. To introduce fundamental AI principles and problem-solving approaches.
2. To understand search techniques, knowledge representation, and reasoning.
3. To explore automated planning, robotics, and expert systems.
4. To introduce artificial neural networks and basic learning algorithms.
5. To discuss ethical considerations and future AI trends.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Identify** the need of Intelligent agents in problem solving.
- **CO2: Compare** and **analyze** different search techniques applied for problem solving.
- **CO3: Demonstrate** knowledge of reasoning and knowledge representation for solving real world problems.
- **CO4: Apply** AI techniques in planning and analyze neural networks
- **CO5: Explain** the concept of expert systems and future AI trends.

Course Contents

Unit I - Introduction to Artificial Intelligence and Intelligent Agent - (09 Hours)

Introduction: What Is Artificial Intelligence (AI), History of AI, and Applications of AI, AI Ethics, Advantage and Disadvantage of AI.

Intelligent Agents: Types, Environments, Rationality, Agent Functions, The Nature of Environments, Features of Environments, Problem-Solving in Artificial Intelligence.

 # Exemplar /Case Studies:AI in Healthcare – Diagnosis & Decision Support Systems.

Unit II - Problem-Solving & Search Strategies - (09 Hours)

State Space Representation, Solving Problems by Searching: Study and analysis of Various searching algorithms, Searching for Solutions.

Uninformed Search Strategies: Breadth-first search, Uniform-cost search, Depth-first search, Depth-limited search, Iterative deepening depth-first search, Bidirectional search.

Informed (Heuristic) Search Strategies: Greedy best first search A* search: Minimizing the total estimated solution cost, Conditions for optimality: Admissibility and consistency, Optimality of A*, Heuristic Functions.

Adversarial Search: Minimax Algorithm, Alpha-Beta Pruning, Constraint Satisfaction Problems(CSPs).

#Exemplar/Case Studies: AI in Games – Chess, Alpha Go, Reinforcement-based Agents.

Unit III - Knowledge Representation & Reasoning - (09 Hours)

Definition of knowledge, types of knowledge, properties for knowledge representation system, AI Knowledge Cycle,

Logical Agents: Propositional Logic, First-Order Predicate Logic (FOPL), Backward Chaining, Forward Chaining, Semantic Network, Truth Maintenance System (TMS), Fuzzy Logic, Statistical and probabilistic reasoning.

#Exemplar/Case Studies: AI for Legal Decision-Making – Knowledge-Based Systems.

Unit IV - Planning and Learning - (09 Hours)

Planning: Search in planning, search vs. planning, components of a planning, Forward planning, Hierarchical planning, STRIPS (Stanford Research Institute Problem Solver)

Learning: Supervised, Unsupervised and Reinforcement Learning, Introduction to Neural Networks, Working of a Neuron, The basic components of ANN, Issues related to Neural computation, Feedforward Networks, Back propagation Algorithm, Applications of Neural Networks.

#Exemplar/Case Studies : AI in Autonomous Systems – Self-driving Cars & Drones.

Unit V - Expert Systems and Emerging trends in AI - (09 Hours)

Expert Systems in AI: Knowledge Engineering, Architecture and Components of expert system Case-Based Reasoning, Steps for building expert system, Rule-Based vs. Data-Driven Approaches.

Emerging Trends in AI: Cognitive Computing, Neuromorphic Computing, AI &Block chain Integration, Symbolic AI vs. Connectionist AI.

#Exemplar/Case Studies : AI for Pattern Recognition – Handwritten Character Recognition.

Learning Resources

Text Books:

1. Stuart Russell and Peter Norvig. "Artificial Intelligence: A Modern Approach". 4th Edition, Pearson, 2020. ISBN: 9780134610993.
2. Saroj Kaushik. "Artificial Intelligence". Cengage Learning India, 2011. ISBN: 9788131518315.
3. Elaine Rich, Kevin Knight, and Shivashankar B. Nair. "Artificial Intelligence". 3rd Edition, McGraw Hill Education, 2017. ISBN: 9781259080972.

Reference Books:

1. Nilsson Nils J. "Artificial Intelligence: A new Synthesis". 2nd Edition Morgan Kaufmann Publishers Inc. San Francisco, CA, ISBN: 9780080948348, 0080948340.
2. Patrick Henry Winston. "Artificial Intelligence". 3rd Edition, Pearson, 1992. ISBN: 9780201533774
3. George F. Luger. "Artificial Intelligence: Structures and Strategies for Complex Problem Solving". 6th Edition, Pearson, 2008. ISBN: 9780321545893
4. Padhy N. P. "Artificial Intelligence and Intelligent Systems". Oxford University Press, 2005. ISBN: 9780195671723

e-Books:

1. https://www.uoitc.edu.iq/images/documents/informatic institute/Competitive_exam/Artificial_Intelli
2. https://www.amazon.in/Artificial-Intelligence-As-AICTE-Intelligent/dp/8126579943.

MOOC Courses:

1. https://onlinecourses.nptel.ac.in/noc25_cs07/preview.
2. https://nptel.ac.in/courses/106/102/106102220/.

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PEC- 520B-MCA : Elective I-Mobile Computing		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Basic understanding of computer networks, operating systems, and programming languages
- Familiarity with basic concepts in wireless communication and database systems.

Companion Course: Mobile Computing Lab

Course Objectives: The course aims to:

1. To explain the architecture and protocols involved in mobile computing and wireless technologies.
2. To explore key mobile platforms and operating systems, particularly Android.
3. To understand data management and application development challenges in mobile environments.
4. To learn Mobile security with cloud integration.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Explain** the concepts, challenges, and technologies in mobile computing and wireless communication.
- **CO2: Analyze** mobile networks and wireless LANs, bluetooth and different protocols like DSR, AODV, ZRP.
- **CO3:Implement** mobile data storage, retrieval, and management in a secure and efficient manner with cloud Integration.
- **CO4:Understand** basic and advanced mobile applications using the Android platform.
- **CO5: Design** advanced mobile application with security.

Course Contents

Unit I - Introduction to Mobile Computing and Wireless Communication - (09 Hours)

Mobile Computing Overview: Introduction, Issues in Mobile Computing, Trends in Mobile Computing. Cellular Concepts: Wireless Telephony, Cellular Networks, Mobile Phone Technologies – 1G to 5G and Beyond (6G Introduction).

GSM Architecture: Air Interface, Channel Structure, HLR-VLR, Handoffs, Channel Allocation.CDMA, GPRS, LTE, and VoLTE Concepts. UMTS: Architecture, Handover, Security.

Value-Added Services (VAS): SMS, MMS, Cell Broadcast Service, and Location-Based Services (LBS). Protocols for Localization, Calling, and Handover.

 # Exemplar /Case Studies: Study GSM Architecture

Unit II - Wireless Networking and Communication Protocols - (09 Hours)

Wireless LANs (WLANs): Overview, MAC Issues, IEEE 802.11 Standards (802.11 a/b/g/n/ac/ax).

Bluetooth: Architecture, Protocols, and Applications.

Wireless Multiple Access Protocols: FDMA, TDMA, CDMA, OFDMA.

Mobile IP: IP Addressing, Handoff, and Mobility Management.

AdHoc networks: Routing, Proactive protocol-DSDV, Reactive Routing Protocols: DSR, AODV

Hybrid routing –ZRP, Multicast Routing: ODMRP, Vehicular Ad Hoc networks (VANET) MANET Vs VANET Security.

Wireless Application Protocol (WAP): WAP Architecture, Protocol Stack, Application Environment. Security Issues in Mobile Networks and Applications

#Exemplar/Case Studies: Study AODV protocol in MANET

Unit III - Mobile Data Management and Cloud Integration- (09 Hours)

Mobile Data Management: Data Synchronization, Replication Issues, and Adaptive Clustering.

Mobile File Systems: Caching, Disconnected Operations, and File Access in Mobile Environments.

Cloud and Mobile Integration: Cloud Storage Services, Mobile Backend as a Service (MBaaS), and IoT Integration.

Managing Big Data in Mobile Environments.Mobile Database and Security Challenges.

#Exemplar/Case Studies: Study Mobile with Cloud like Azure.

Unit IV - Mobile Operating Systems and Platforms - (09 Hours)

Overview of Mobile Operating Systems: Android, iOS, Windows Mobile, and Embedded Linux.

Android Development: Overview of Android Devices, Why Develop for Android, Android Features and Architecture.

Designing User Interfaces in Android: Views, View Groups, Layouts, Menus, Dialogs, and Fragments. Introduction to Cross-Platform Mobile Development (React Native, Flutter).

#Exemplar/Case Studies : Study All Mobile Operating Systems.

Unit V - Advanced Android Application Development and Mobile Security - (09 Hours)

Intents, Adapters, and Notifications.

Accessing Device Services: Camera, GPS, Sensors, and Location-Based Services.

Data Storage and Sharing: Internal and External Storage, SQ Lite, File System in Android.

Networking in Android: HTTP Requests, Web Services, and REST APIs.

Advanced Topics: Push Notifications, Firebase Integration, and Mobile Security Best Practices.

#Exemplar/Case Studies : Study various storage used in android application development.

Learning Resources

Text Books:

1. Jochen Schiller, Mobile Communications, 2nd Edition, Pearson Education.
2. Reto Meier, Professional Android Development, Wiley India.
3. Raj Kamal, Mobile Computing, Oxford University Press.

Reference Books:

1. Mobile Communications J. Schiller, Addition Wesley Publication ISBN-10: 0-321-12381-6
2. GSM System Engineering A.Mehrotra, Addition Wesley Publication ISBN: 0890068607 978089006860
3. Understanding WAP M. Heijden, M. Taylor, Artech House Publication ISBN: 1580530931
4. Professional Android Application Development Wrox Publications, Reto Meier ISBN 1118237226.
5. Upadhyaya, Mobile Computing, Springer ISBN 978-0-306-47602-0
6. Sams teach yourself Android application development, Lauren Dercy and ShandeConder, Sams publishing ISBN: 9780134178318
7. Mobile Computing: Asoke K Talukdar, Roopa R. Yavagal, ISBN-13: 978-0-07-014457-6, Tata McGraw-Hill
8. Principles of Mobile Computing , Hansmann, Merk, Nicklous, Stober, Springer, second edition

e-Books:

1. http://freecomputerbooks.com/mobileDeviceProgrammingBooks.html
2. http://www.freebookcentre.net/mobile-technology/mobile-technology-books.html

MOOC Courses:

1. https://www.coursera.org/courses?query=mobile%20cloud%20computing
2. https://www.ed.youth4work.com/course/479-mobile-computing-online-course

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PEC-520C-MCA : Elective I-Data Analytics		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Basics of Computers
- Basic Data Base Management System and mathematics

Companion Course: Elective Lab

Course Objectives: The course aims to:

1. To Gain knowledge of the basic concepts of data analytics.
2. To Explain the concept of Data Collection and pre-processing
3. To Gain the ability to grasp the importance of EDA in the data analytics pipeline to uncover patterns.
4. To gain the knowledge of DDDM
5. To Explain Key Concepts in Text Analytics.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Define** the fundamental concepts of data analytics.
- **CO2: Apply** the fundamental concepts of data collection and pre-processing.
- **CO3: Explain** the concept of EDA.
- **CO4: Explain** the concept and importance of Data-Driven Decision Making.
- **CO5: Define** the fundamental concepts of Text analytics.

Course Contents

Unit I - Introduction to Data Analytics - (09 Hours)

Introduction to Data Analytics: Overview of Data Analytics: Importance, scope, and applications of data analytics.

Types of Data Analytics: Descriptive, Diagnostic, Predictive, Prescriptive , Data Science vs Data Analytics vs Business Intelligence

Data Types: Structured, unstructured, and semi-structured data, Overview of the Data Analytics Process

Data Collection Methods: Surveys, Web, Databases, APIs, IoT

 # Exemplar /Case Studies: Analyzing Sales Performance of a Retail Chain

Unit II - Data Collection and Preprocessing - (09 Hours)

Importance of Pre-processing in Data Analytics

Data Quality: Common issues (e.g., missing data, duplicates) and their solutions.

Data Cleaning Techniques: Handling missing data, duplicates, and outliers

Data Transformation: Normalization, Scaling, Encoding , Data Integration and Reduction

Encoding Categorical Data: Label Encoding, One-Hot Encoding, Ordinal Encoding

Introduction to Tools: Excel, Python (Pandas), or R

#Exemplar/Case Studies: Customer Feedback Analysis for a Food Delivery App.

Unit III - Exploratory Data Analysis (EDA) - (09 Hours)

Definition and Purpose of EDA , Understanding the Structure and Type of Data , Calculating basic statistical measures , Central Tendency: Mean, Median, Mode , Dispersion: Variance, Standard Deviation, Range, IQR , Understanding the Structure and Type of Data, Identifying types of variables: Numerical (continuous, discrete), Categorical (ordinal, nominal), Date/Time. Checking dataset structure using .shape, .info(), dtypes (in Python) or summary tools in Excel/R , Univariate Analysis , Bivariate and Multivariate Analysis .

#Exemplar/Case Studies: EDA on Online Retail Dataset

Unit IV -Data-Driven Decision Making - (09 Hours)

Introduction to Data-Driven Decision Making (DDDM) , Steps in the data-driven decision lifecycle , Tools and Techniques for DDDM , Role of tools like Excel, Python (Pandas, Matplotlib), R in decision support Dashboards and Visual Analytics (e.g., Power BI, Tableau) , Data Ethics and Responsible Decision Making, Reporting and communication of insights to stakeholders .

#Exemplar/Case Studies : Marketing Campaign Analysis

Unit V - Introduction to Text Analytics - (09 Hours)

Understanding Text Analytics, Importance and challenges of analysing unstructured data, Applications of text analytics: sentiment analysis, customer feedback, document classification. Basics of Natural Language Processing (NLP), Key NLP tasks: Tokenization, Lemmatization, Stopword Removal. Text Representation Techniques : Bag of Words (BoW) , TF-IDF (Term Frequency–Inverse Document Frequency) , Basic comparison of BoW vs TF-IDF

#Exemplar/Case Studies : Analyzing Customer Reviews for a Product

Learning Resources

Text Books:

1. Provost, F., & Fawcett, T. (2013). Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media. ISBN: 978-1449361327
2. Bruce, P., Bruce, A., & Gedeck, P. (2020). Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python (2nd ed.). O'Reilly Media. ISBN: 978-1492072942

3. McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython (2nd ed.). O'Reilly Media. ISBN: 978-1491957660

4. Sarkar, D. (2016). Text Analytics with Python: A Practical Real-World Approach to Gaining Actionable Insights from Your Data. Apress. ISBN: 978-1484223871

Reference Books:

1. Atkinson-Abutridy, J. (2022). Text Analytics: An Introduction to the Science and Applications of Unstructured Information Analysis. CRC Press. ISBN: 978-1032245263

2. Öhman, E. (2024). Introduction to Text Analytics: A Guide for Digital Humanities & Social Sciences. SAGE Publications Ltd. ISBN: 978-1529620047

e-Books:

1. https://python-course.eu/books/bernd_klein_python_data_analysis_a4.pdf

MOOC Courses:

1. Data Analytics with Python - Course

2. NPTEL :: Computer Science and Engineering - NOC:Data Analytics with Python

3. nptel.ac.in/courses/106107220

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-504-MCA : Data Structures Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 04 Hours/Week	02	**Term Work** : 50 Marks **Practical** : 50 Marks

Prerequisite Courses: NA

Companion Course: Data Structures and Algorithms

Course Objectives: The course aims to:

1. To understand the representation, implementation of basic data structures.
2. To apply and relate various linear & nonlinear data structures
3. To classify and solve applications of Data Structure in solving real life problems
4. To apply various searching & sorting algorithms.
5. To understand and apply various hashing techniques

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Analyze** the structure and operations of elementary data structures such as arrays and linked lists.
- **CO2: Differentiate** between various representations and applications of linked lists based on problem requirements.
- **CO3: Analyze** and **evaluate** the applications of stacks and queues in solving computational problems.
- **CO4: Analyze** and **apply** tree and graph data structures to model and solve mathematical and real-world problems.
- **CO5: Apply** and **compare** different searching, sorting, and hashing algorithms for performance and efficiency.

Suggested List of Laboratory Assignments

- From group A, B, C and D any 2 assignments can be implemented whereas from Group E any 4 programs can be implemented.
- Total 12 assignments to be submitted for final evaluation.
- Programs can be implemented in C/ C++

Group A (Any Two Assignments)

1. Create an array to store student names in a class, allowing insertion, deletion, and traversal operations.

2. Represent a railway timetable as a sparse matrix where only active train schedules are stored efficiently.
3. Implement a program to analyze the performance of sorting student names alphabetically and measure its time complexity using Big O notation.
4. Store and display marks of students in multiple subjects using a 2D array.

Group B (Any Two Assignments)

1. Implement a student record system using a singly linked list that supports creating, inserting, deleting, and displaying student records.
2. Write a program to store a sequence of train stations in a singly linked list and reverse the order to simulate a return journey.
3. Merge two linked lists containing student names from two different classes into a single list.
4. Implement a music playlist where songs can be played forward and backward using a doubly linked list.
5. Implement a circular linked list to simulate a round-robin scheduling algorithm for CPU processes.

Group C (Any Two Assignments)

1. Implement a stack-based text editor feature that supports undo/redo operations.
2. Use the stack ADT to check whether parentheses in an arithmetic expression are balanced.
3. Implement a queue to simulate a customer service system where customers join and leave in FIFO order.
4. Design a circular queue to efficiently manage a printer queue where print jobs are cyclically processed.
5. Implement a priority queue for a hospital emergency room, where patients with higher severity are treated first.

Group D (Any Two Assignments)

1. Develop a BST to store and search for student records based on roll numbers.
2. Implement tree traversal techniques to display hierarchical employee structures in a company.
3. Write a program to compute the depth of a tournament bracket structure in a sports event.
4. Implement DFS to find paths in a city's metro rail system using an adjacency list.
5. Implement BFS to find the shortest route between two locations in a road network.

Group E (Any Four Assignments)

1. A school maintains a list of student names who have registered for an event. Given a student's name, implement sequential search to check if they have registered.
2. A university stores student roll numbers in a sorted list. Implement binary search to find whether a given roll number exists in the list.
3. A library maintains a large database of books, where each book has a unique ISBN number. Implement hashing to store and search for books efficiently using their ISBN numbers.
4. Sort a dataset of e-commerce product prices using quick sort.
5. Implement the Bubble Sort algorithm to sort an array of student marks.
6. Implement the Selection Sort algorithm to sort product prices in the shop.
7. Implement the Insertion Sort algorithm to sort student names in ascending order

Mini Project Topic (Any one project/ concept)

1. Design a Library Management System that manages books, users, and book transactions (issue, return) efficiently. Features to Implement:

- Book Issue and Return: Users can issue and return books. If the book is available, it is removed from the available list, and if the book is returned, it's added back to the list.
- Reservation: If a book is already issued, users can reserve the book by joining a queue. Once the book is returned, the next person in the queue receives it.
- Book Search: Allow searching for a book by title or author using the linked list traversal.
- User Management: Store user details (ID, name, issued books) in an array, with the ability to search and update.

2. Create a navigation and services management system for a Smart City, including finding routes, managing emergency services, and maintaining citizen service records.

- Implementation Details: Cities and roads are represented as a graph, with each node being an intersection and edges representing roads with weights (distance or time). Use Dijkstra's algorithm to calculate the shortest path for navigation. Optionally, implement BFS for unweighted graphs.

Learning Resources

Text Books:

1. Aho A., Hopcroft J., Ulman J., "Data Structures and Algorithms", Pearson Education, ISBN-0201-43578- 02.

2. Brassard & Bratley, “Fundamentals of Algorithmics”, Prentice Hall India/Pearson Education, ISBN 13- 9788120311312.

3. Weiss, M. A., “Data Structures and Algorithm Analysis in C++ “, (4th ed.). Pearson. ISBN-13: 978-0-13- 284737-7. (2013)

Reference Books:

1. Goodrich, Tamassia, and Goldwasser, "Data Structures and Algorithms in Java," Wiley, ISBN-10: 1118771338, ISBN-13: 9781118771334.

2. Lafore, "Data Structures and Algorithms in Java," Sams Publishing, ISBN-10: 0672324539, ISBN- 13: 9780672324536.

3. Horowitz and Sahani, “Fundamentals of Data Structures in C++”, University Press, ISBN 10:0716782928 ISBN 13: 9780716782926.2.

4. Drozdek, “Data Structures and Algorithms in Java”, 2nd ed. Boston, MA: Course Technology, 2004.

5. Yedidyah Langsam, Moshe J Augenstein, Aron M Tenenbaum, “Data Structures using C and C++”,Pearson Education, ISBN 81-317-0328-2.3.

6. Trembley, J.P. and Sorenson P.G., “An Introduction to Data Structures with Applications”,McGrawHill

E-Books Links

1. https://eprints.triatmamulya.ac.id/1694/1/Data%20Structures%20and%20Algorithms%20Made%20Easy_%20Data%20Structures%20and%20Algorithmic%20Puzzles.pdf

2. https://mimoza.marmara.edu.tr/~msakalli/cse706_12/SkienaTheAlgorithmDesignManual.pdf

3. https://dahlan.unimal.ac.id/files/ebooks/2013%20Algorithms_Unlocked.pdf

4. https://dl.ebooksworld.ir/books/Introduction.to.Algorithms.4th.Leiserson.Stein.Rivest.Cormen.MIT.Press.9780262046305.EBooksWorld.ir.pdf

MOOC Courses:

1. https://nptel.ac.in/courses/106106133

2. https://nptel.ac.in/courses/106106127

3. https://nptel.ac.in/courses/106105085

4. https://nptel.ac.in/courses/106102064

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
PCC-505-MCA : Object Oriented Programming Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 04 Hours/Week	02	**Term Work** : 25 Marks **Practical** : 25 Marks

Prerequisite Courses: Fundamental knowledge of programming concepts

Companion Course: Object Oriented Programming

Course Objectives: The course aims to:

1. To provide hands-on experience in applying object-oriented programming concepts such as classes, objects, inheritance, and polymorphism using C++.
2. To enable students to write efficient and reusable code by utilizing constructors, destructors, operator overloading, and function overloading.
3. To develop skills in managing runtime errors and generic programming through the use of exception handling and templates.
4. To familiarize students with file handling techniques in C++ for building real-time, data-persistent applications.
5. To promote the development of modular and well-documented software solutions by integrating multiple object-oriented features in solving real-world problems.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Implement** C++ programs that demonstrate basic OOP principles such as classes, objects, and constructors.
- **CO2: Use** inheritance and polymorphism to design modular and reusable code components.
- **CO3: Develop** C++ programs using templates and exception handling to create generic and robust solutions.
- **CO4: Apply** file handling techniques to build persistent data storage in object-oriented applications.
- **CO5: Design** and implement mini projects demonstrating integrated use of OOP features in solving real-world problems.

List of Laboratory Experiments/Assignments (Any 10 laboratory assignments) based on Programming

Sr. No.	Practical Assignments

1	Design a mini ATM interface that simulates balance inquiry, deposit, and withdrawal using encapsulation and basic C++ constructs.
2	Create a C++ program to simulate a basic calculator supporting expression parsing using function overloading and inline functions.
3	Write a menu-driven program using control structures to simulate a student grading system with support for conditional and looping constructs.
4	Implement a class for a Complex Number, overload the +, -, and * operators, and manage memory using dynamic allocation.
5	Design a class for a Student Database that allows the addition, deletion, and search of student records using static members and friend functions.
6	Create a pointer-based implementation of a class where objects are dynamically created and manipulated using the this pointer and constructors/destructors.
7	Implement a multi-level inheritance model representing an organization hierarchy and demonstrate method overriding and virtual functions.
8	Create a program using hybrid inheritance to model academic and extracurricular performance and resolve ambiguity using virtual base classes.
9	Design a payroll system using run-time polymorphism (abstract classes + pure virtual functions) for different employee types (hourly, salaried, contractual).
10	Create a template-based sorting class that works with multiple data types and incorporates exception handling for invalid inputs.
11	Develop a custom vector class using C++ STL and templates, with member functions to perform insert, delete, and search operations.
12	Write a C++ program to simulate a bank transaction system that throws and handles multiple exceptions (e.g., insufficient funds, invalid account).
13	Create a Library Management System using file I/O to store and retrieve book records. Include command-line arguments to filter search results.
14	Design a C++ program that logs errors into a file and uses exception handling to manage file not found or access denied issues.
15	Build a program using object-oriented design principles to simulate user authentication (login/logout), with user data stored in a file.

Mini Project Topic (Any one project/ concept)

1. Develop a simple banking system that allows account creation and basic transactions (case study).
2. Develop a student management system to store and display student details (case study).
3. Create an employee payroll system with different types of employees using inheritance (case study).
4. Develop a generic sorting system that sorts different data types using templates (case study).

5. Develop a Library Management System to maintain records of issued and available books (case study).

Learning Resources

Text Books:

1. Bjarne Stroustrup, The C++ Programming Language, 3rd Edition, Pearson Education. ISBN: 9780201889543
2. Paul Deitel, Harvey Deitel, C++ How to Program, 4th Edition, Pearson Education. ISBN: 81-297-0276-2

Reference Books:

1. Robert Lafore, Object-Oriented Programming in C++, 4th Edition, Sams Publishing. ISBN: 0672323087
2. E. Balagurusamy, Object-Oriented Programming with C++, Tata McGraw Hill. ISBN: 9780071072830
3. Herbert Schildt, C++: The Complete Reference, 8th Edition, McGraw Hill Professional. ISBN: 9780072226805
4. Matt Weisfeld, The Object-Oriented Thought Process, 3rd Edition, Pearson. ISBN-13: 0752063330166
5. Cox Brad, Andrew J. Novobilski, Object-Oriented Programming: An Evolutionary Approach, 2nd Edition, Addison-Wesley. ISBN-13: 9780201548341

E-Books Links

https://www.freebookcentre.net/programming-books-download/Lecture-Notes-On-Object-Oriented-Programming-Using-C++.html

MOOC Courses:

1. https://nptel.ac.in/courses/106101208
2. https://nptel.ac.in/courses/106105151
3. https://onlinecourses.nptel.ac.in/noc20_cs53/preview
4. https://onlinecourses.nptel.ac.in/noc20_cs07/preview

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
PEC-521-MCA : Elective-I Laboratory (Artificial Intelligence)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week	01	**Term Work** : 25 Marks **Oral** : 25 Marks

Prerequisite Courses:

- Basic programming knowledge (Python/C++)
- Fundamental concepts of logic and probability

Companion Course: Artificial Intelligence

Course Objectives: The course aims to:

1. To implement AI search techniques for problem-solving.
2. To apply knowledge representation and reasoning using Python.
3. To explore AI planning, robotics, and expert systems practically.
4. To introduce basic artificial neural networks and learning algorithms.
5. To develop AI applications relevant to industry and placements.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Implement** AI search strategies and problem-solving techniques.
- **CO2: Develop** knowledge-based AI systems using logic programming.
- **CO3: Apply** AI planning and robotics concepts to real-world problems.
- **CO4: Build** artificial neural networks and apply basic learning algorithms.

List of Laboratory Experiments/Assignments (Any 6 to 8 laboratory assignments) based on Programming.

Sr. No.	Practical Assignments
1	Implement a program of Tic-Tac-Toe game using Python.
2	Create a program of 8-Puzzle problem using Python.
3	Implement a program of Water-Jug problem using Python.
4	Write a program of Travelling Salesman Problem using Python.
5	Implement a program of Tower of Hanoi using Python.
6	Implement a program of Monkey Banana Problem using Python.
7	Write a program of Alpha-Beta Pruning using Python.

8	Implement a program of 8-Queens Problem using Python.
9	Design an expert system of Employee performance evaluation system using python.
10	Design an expert system of Information management system using python.

Mini Project Topic (Any one project/ concept)

1. AI Chatbot for Customer Support
   - Use Natural Language Processing (NLP) to develop a chatbot for handling basic customer queries.
   - Tools: Python, NLTK, Dialogflow.
2. AI-Based Resume Screening System
   - Develop an AI system that analyzes and ranks resumes based on job descriptions.
   - Tools: Python, TF-IDF, NLP.
3. AI-Based Handwritten Character Recognition
   - Implement Artificial Neural Networks (ANN) to recognize handwritten characters.
   - Tools: TensorFlow, Keras, OpenCV.
4. AI-Powered Recommendation System
   - Build a movie or product recommendation system using AI techniques.
   - Tools: Python, Pandas, Scikit-learn.

Learning Resources

Text Books:

1. Stuart Russell and Peter Norvig. "Artificial Intelligence: A Modern Approach". 4th Edition, Pearson, 2020. ISBN: 9780134610993.
2. Saroj Kaushik. "Artificial Intelligence". Cengage Learning India, 2011. ISBN: 9788131518315.
3. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372

Reference Books:

1. Elaine Rich, Kevin Knight, and Shivashankar B. Nair. "Artificial Intelligence". 3rd Edition, McGraw Hill Education, 2017. ISBN: 9781259080972.
2. Padhy N. P. "Artificial Intelligence and Intelligent Systems". Oxford University Press, 2005. ISBN: 9780195671723.

3. David Beazley, Brian K. Jones “Python Cookbook”, 3rd Edition. O’Reilly Publications

4. Martin C. Brown, “Python: The Complete Reference”, McGraw Hill Education, ISBN-10:978938757294 ISBN-13:978-9387572942, ASIN: 9387572943.

E-Books Links

https://www.uoitc.edu.iq/images/documents/informaticsinstitute/Competitive_exam/Artificial_Intelligence.pdf

MOOC Courses:

https://onlinecourses.nptel.ac.in/noc25_cs07/preview

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
PEC-521-MCA : Elective-I Laboratory (Mobile Computing)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week	01	**Term Work** : 25 Marks **Oral** : 25 Marks

Prerequisite Courses: Android Studio, J2ME

Companion Course: Mobile Computing

Course Objectives: The course aims to:

1. To study the emerging technologies in mobile computing and wireless communication.
2. To understand the mobile computing environment.
3. To learn about recent and future technologies use to develop mobile applications.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Design** successful mobile computing applications and services.
- **CO2: Use** contemporary development environment and languages (e.g., C#, Java,J2SE, WML) to develop mobile applications.
- **CO3: Program** the typical functionalities of modern smartphones (e.g., light sensor, gyro, accelerometer, cameras, microphones, GPS, barometer).
- **CO4: Differentiate** and **Design** Mobile App using Intent and activity.
- **CO5: Work** effectively as a member of a team to complete a large programming project.

List of Laboratory Experiments/Assignments (Any 6 to 8 laboratory assignments) based on Programming

Sr. No.	Practical Assignments
1	Study of GSM architecture.
2	Installation of Android studio.
3	Development Of Hello World Application.
4	Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.
5	Create a screen that has input boxes for User Name, Password, Address, Gender(radio buttons for male and female), Age (numeric), Date of Birth (Datepicker), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button (use any layout)
6	Design an android application to create page using Intent and one Button and pass the Values from one Activity to second Activity

7	Design an android application Send SMS using Intent.
8	Create an android application using Fragments.
9	Design an android application Using Radio Buttons.
10	Design an android application for Menu.

Mini Project Topic (Any one project/ concept)

1. Android Smart City Traveler: This application is to create a schedule for the traveler travelling to the city and wants to explore the city by specifying the time in hours.
2. Fake Caller Application: This app will be used to make a fake call to the device for emergency situations.
3. Color guessing game: This app will be a fun game where users have to guess the color of the word shown on the screen.
4. Parental Control Application: This app will allow parents to monitor and control their children's online activities.
5. Timetable Manager: This app will help students to manage their daily schedule and reminders.

Learning Resources

Text Books:

1. Jochen Schiller, Mobile Communications, 2nd Edition, Pearson Education.
2. Reto Meier, Professional Android Development, Wiley India.
3. Raj Kamal, Mobile Computing, Oxford University Press.

Reference Books:

1. Mobile Communications J. Schiller, Addition Wesley Publication
2. GSM System Engineering A.Mehrotra, Addition Wesley Publication ISBN:0890068607, 978089006860
3. Understanding WAP M. Heijden, M. Taylor, Artech House Publication

E-Books Links

1. http://freecomputerbooks.com/mobileDeviceProgrammingBooks.html
2. http://www.freebookcentre.net/mobile-technology/mobile-technology-books.html

MOOC Courses:

1. https://www.coursera.org/courses?query=mobile%20cloud%20computing
2. https://www.ed.youth4work.com/course/479-mobile-computing-online-course

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
PEC-521-MCA : Elective-I Laboratory (Data Analytics)		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week	01	**Term Work :** 25 Marks **Oral :** 25 Marks

Prerequisite Courses:

- Basics of Computer programming
- Basics of Python /R Programming

Companion Course: Elective I-Data Analytics

Course Objectives: The course aims to:

1. To Gain knowledge of the basic concepts of data analytics.
2. To Explain the concept of Data Collection and pre-processing
3. To Gain the ability to grasp the importance of EDA in the data analytics pipeline to uncover patterns.
4. To gain the knowledge of DDDM
5. To Explain Key Concepts in Text Analytics

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Apply** data pre-processing techniques on dataset
- **CO2: Apply** EDA Techniques
- **CO3: Build** dashboard using power BI /Tableau
- **CO4: Apply** text analytics techniques

List of Laboratory Experiments/Assignments (Any 6 to 8 laboratory assignments) based on Programming

Sr. No.	Practical Assignments
1	Getting Started with Python & Jupyter Notebook a) Install Anaconda and open Jupyter Notebook. b) Print the message: “Welcome to Data Analytics”. c) Perform simple arithmetic operations: addition, subtraction, multiplication, and division. On a= 10 and b=5

2	Create a small table using Python to represent structured data (like name and marks) a) Display unstructured data (a paragraph of text). b) Create a JSON example for semi-structured data.
3	Write a program to perform descriptive statistics on sample data a) Create a list of numbers. b) Calculate mean, minimum, and maximum. c) Create a simple bar chart using the values.
4	Write a program for Reading and Exploring a CSV File a) Download a CSV file (e.g., students.csv) with names and marks. b) Read it using pandas. c) Display the first 5 rows.
5	Write a program to explore data pre-processing techniques : a) Create a sample dataset with missing values and duplicates. b) Identify and display the missing values and duplicates. c) Remove duplicate rows from the dataset , Drop rows with missing values d) Apply Min-Max normalization and Standard scaling to the values.
6	Write a program to explore Encoding Categorical Variables Given a dataset with a "City" column containing [’Mumbai’, ’Delhi’, ’Pune’] a) Perform Label Encoding b) Perform One-Hot Encoding c) Perform Ordinal Encoding (assuming some order)
7	Write a program to explore EDA technique and dataset Structure a) Load a sample dataset (e.g., Titanic, Iris, or your own CSV). b) Check its shape, data types, and summary using .shape, .info(), .dtypes, .head(). c) From the given dataset (titanic.csv or student.csv), select one numerical column (e.g., Age or Marks) and one categorical column (Gender). d) Perform univariate analysis on the numerical column by plotting a histogram and a boxplot. e) Using the student.csv dataset, analyze the relationship between two numerical variables for example, StudyHours and MarksObtained by creating a scatter plot. Comment on any visible trend between study time and performance.
8	Create Dashboards and Visual Analytics Introduction using Power BI/Tableau Import student.csv into Power BI or Tableau. a) Create a simple dashboard with visuals like: • StudyHours vs Marks • Pie chart of attendance • Average marks by category (if available, e.g., gender or class) b) Interpret the dashboard insights.

9	Write a program to perform basic NLP tasks on sample text data Convert the text to lowercase b) Tokenize the text c) Remove stopwords d) Lemmatize the remaining words
10	Write a program for Text Representation using Bag of Words a) Create a list of 3 sample sentences b) Convert the text data into Bag of Words (BoW) representation using CountVectorizer c) Show the BoW matrix and feature names
11	Write a program for Text Representation using TF-IDF a) Use the same text corpus from the previous assignment b) Convert the text into TF-IDF representation c) Compare it with BoW in terms of values

Mini Project Topic (Any one project/ concept)

1. Student Performance Analysis
2. Survey on Mobile Phone Usage Among Students
3. Retail Store Sales Data Analysis
4. YouTube or OTT Platform Viewing Habits

Learning Resources

Text Books:

1. Maheshwari, A. (2014) Data Analytics Made Accessible (1st ed.). Amazon Digital Services. ISBN: 978-0692331325
2. Provost, F., & Fawcett, T. (2013). Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media. ISBN: 978-1449361327
3. McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython (2nd ed.). O'Reilly Media.ISBN: 978-1491957660

Reference Books:

1. Albright, S. C., & Winston, W. L. (2016) Business Analytics: Data Analysis and Decision Making (6th ed.). Cengage Learning.ISBN: 978-1337406420
2. Wilke, C. O. (2019) Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures. O'Reilly Media. ISBN: 978-1492031086

E-Books Links

1. https://jakevdp.github.io/PythonDataScienceHandbook/

MOOC Courses:

1. https://onlinecourses.swayam2.ac.in/cec22_cs20/preview
2. https://onlinecourses.nptel.ac.in/noc24_cs65/preview
3. https://www.udemy.com/course/the-business-intelligence-analyst-course-2018

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-506-MCA : Python Programming Laboratory		
Teaching /scheme	**Credits**	**Examination Scheme**
Tutorial: 1 Hours/Week **Practical:** 2 Hours/Week	01 01	**Term work:** 50 Marks

Prerequisite Courses: Fundamental knowledge of programming concepts

Companion Course: NA

Course Objectives: The course aims to:

1. Introduce students to Python programming fundamentals and problem-solving techniques.
2. Develop the ability to write modular programs with functions, and handle files and exceptions effectively.
3. Familiarize learners with Object-Oriented Programming concepts in Python for structured software design.
4. Equip students with Python libraries for data analytics and visualization to support real-world applications in data science.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Demonstrate** understanding of Python fundamentals including data types, control structures, and basic programming constructs.
- **CO2: Apply** functions, file handling, and exception handling techniques to build robust Python applications.
- **CO3: Design** and **implement** object-oriented solutions using classes, inheritance, polymorphism, and abstraction in Python.
- **CO4: Utilize** Python libraries such as NumPy, Pandas, Matplotlib, and Seaborn for data analytics and visualization.

Course Contents

Unit I - Python Fundamental - (03 Hours)

Introduction to Python, Features of Python, Python interpreter, interactive and non-interactive mode of Python Literal constants, variables and identifiers, Data Types, Input operation, Comments, Reserved words, Indentation, Operators and expressions, Expressions in Python. Decision control statement, branching Statements: if, if-else, nested if, if-else statements, Structures/Iterative statements: while loop, for loop, selecting appropriate loop. Nested loops, The break, continue, pass, else statement used with loops. Other data types- Tuples, Lists, Sets and Dictionary.

Strings: Creating and Storing Strings, String-only Operators, String Built-in Methods, Special Features of Strings, Accessing Characters in String by Index Number, String Slicing and Joining, Formatting Strings.

Unit II - Functions, Modules, File & Exception Handling - (04 Hours)

Functions: Calling Functions, Creating Functions, Formal Arguments, Positional Arguments, Default Arguments, Default Function Object Argument Example, Variable-length Arguments, Non-keyword Variable Arguments (Tuple), Keyword Variable Arguments (Dictionary), user defined functions and library functions, The return Statement and void Function, Scope and Lifetime of Variables, *args and **kwargs, Command Line Arguments.

Modules and Packages: Standard Library modules, Commonly Used Modules, Categorizing the Standard Types, Understanding Packages Powerful Lambda or anonymous function in python,

File Handling: Opening/closing files, Reading/writing text & binary files, File operations with context manager (with)

Exception Handling: Try, except, finally, else, User-defined exceptions, Debugging techniques

Unit III - Object Oriented Programming Concept - (03 Hours)

Object Oriented programming(OOP), Python Objects, Standard Types, Other Built-in Types, Internal Types, scope, Classes and Objects, Creating Classes in Python, Creating Objects in Python, The Constructor Method, Classes with Multiple Objects, Class Attributes versus Data Attributes, Encapsulation, Inheritance, and Polymorphism, Composition, containership, reusability, delegation, data abstraction.

Unit IV - Basic Python Libraries - (04 Hours)

NumPy: arrays, operations, slicing, broadcasting.

Pandas: Series, DataFrame, importing/exporting data (CSV, Excel), data cleaning.

Matplotlib: line, bar, scatter, histogram plots, pie chart.

Seaborn: heatmap, pairplot, boxplot.

Basic exploratory data analysis (EDA) workflow.

List of Laboratory Experiments/Assignments (Any 6 to 8 laboratory assignments) based on Programming

Sr. No.	Practical Assignments
1	Write a Python program to calculate the sum of three given numbers, if the values are equal then return three times of their sum.
2	Write a Python program to find the factorial of a non-negative integer using decision control statements.
3	To accept student's five courses marks and compute his/her result. Student is passing if he/she scores marks equal to and above 40 in each course. If student scores aggregate greater than 75%, then the grade is distinction. If aggregate is 60>= and <75 then the grade if first division. If aggregate is 50>= and <60, then the grade is second division. If aggregate is 40>= and <50, then the grade is third division.

4	To check whether input number is Armstrong number or not. An Armstrong number is an integer with three digits such that the sum of the cubes of its digits is equal to the number itself.
5	Write a Python program which accepts a sequence of comma-separated numbers from user and generate a list and a tuple with those numbers.
6	Write a Python program to get a directory listing, sorted by creation date.
7	Write a Python program using Lambda functions to manipulate strings. • Convert a string to uppercase. • Check if a string starts with a specific character.
8	Using concepts of OOP, write a Python program for banking application which performs following operations: • Create an account (Account_no., Name, Account_type, Balance). • Perform transactions – Deposit money, Withdraw money. • Check the balance of the account. • Display account information.
9	Create class EMPLOYEE for storing details (Name, Designation, gender, Date of Joining and Salary). Define function members to compute a) total number of employees in an organization b) count of male and female employee c) Employee with salary more than 10,000 d) Employee with designation “Asst Manager”.
10	Write a Python program to calculate the Mean, Standard Deviation, Mode, Median, Minimum, and Maximum values of an array using appropriate NumPy functions.
11	Download “Vehicle Sales” dataset from Kaggle website and write a Python program to perform following operations: • Read the dataset file. • Show top ten sales records from the dataset. • Accept vehicle model and plot the Line graph based on sale date and selling price.
12	Write a Python program to create a simple plot using Matplotlib and utilize NumPy arrays to store and manipulate data points.

Mini Project Topic (Any one project/ concept)

1. **Student Grade Calculator:** A program to input marks, calculate averages, and assign grades.
2. **Library Management System:** Manage books using file handling with search, add, and delete

options.

3. **ATM Simulation:** Object-oriented program for account creation, balance inquiry, and transactions.

4. **Sales Data Analysis Dashboard:** Use Pandas and Matplotlib/Seaborn to analyze and visualize sales trends.

Learning Resources

Text Books:

1. Chun, J Wesley, Core Python Programming, Second Edition, Pearson, 2007 Reprint 2010

2. Kenneth A. Lambert, The Fundamentals of Python: First Programs, Cengage Learning, ISBN:978-1111822705

3. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372

Reference Books:

1. Barry, Paul, Head First Python, 2nd Edition, O Rielly, 2010

2. Lutz, Mark, Learning Python, 4th Edition, O Rielly, 2009

3. David Beazley, Brian K. Jones "Python Cookbook", 3rd Edition. O Reilly Publications

4. Jake VanderPlas "Python Data Science Handbook" O Reilly Publications

5. Martin C. Brown, "Python: The Complete Reference", McGraw Hill Education, ISBN-10: 97893875729 ISBN-13: 978-9387572942, ASIN: 9387572943

MOOC / NPTEL/YouTube Links:

1. https://onlinecourses.swayam2.ac.in/cec22_cs20/preview

2. https://nptel.ac.in/courses/106106145

3. https://www.udemy.com/course/python-for-beginners-zero-to-hero-e

Savitribai Phule Pune University, Pune

Maharashtra, India

Master of Computer Applications (Under Engineering) (2025 Pattern)

Semester II

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-551-MCA : Operating System and Network Fundamentals		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Basics of Operating Systems
- Basics of Networking

Companion Course: NA

Course Objectives: The course aims to:

1. To understand the fundamental concepts of operating systems and memory management.
2. To understand concepts of storage and process management.
3. To understand types of networks, topologies, transmission media, and networking devices.
4. To understand roles of network protocols and various routing algorithms.
5. To develop knowledge of network security threats and management tools.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Identify** and differentiate different types of operating systems and memory management strategies.
- **CO2: Explain** and compare file systems, CPU scheduling, and process synchronization techniques.
- **CO3: Apply** knowledge to classify various types of networks, topologies, models, transmission media, and networking devices.
- **CO4: Analyze** the working of key network protocols and compare different routing algorithms.
- **CO5: Assess** network security threats and design appropriate management tools to enhance network security and reliability.

Course Contents

Unit I - Operating System Fundamentals - (09 Hours)

Introduction to Operating Systems: Definition and Purpose: Understanding the role and objectives of an OS.

Types of Operating Systems: Batch, Time-sharing, Distributed, Real-time, and Embedded Systems. Memory Management: Allocation Techniques: Contiguous vs. Non-contiguous memory allocation. Paging and Segmentation: Mechanisms and benefits. Virtual Memory: Concepts, demand paging, and page replacement algorithms.

 # Exemplar /Case Studies: Windows vs. Linux – A Comparative Analysis of Memory Management

Unit II - Storage Management and Process Management - (09 Hours)

Storage Management: File Systems: Directory structures, file allocation methods, and free-space management. Disk Scheduling: Algorithms like FCFS, SSTF, SCAN, and C-SCAN.

Process Management: Process Concepts: Definition, lifecycle, and control blocks. Thread Management: Multithreading models and benefits.

CPU Scheduling: Algorithms like FCFS, SJF, Round Robin, and Priority Scheduling. Process Synchronization: Critical sections, semaphores, and monitors.

Deadlocks: Conditions, prevention, avoidance, detection, and recovery.

#Exemplar/Case Studies: Process Scheduling in Modern Operating Systems – A Case of Android

Unit III - Networking Fundamentals - (09 Hours)

Introduction to Networking: Definition and Importance: Understanding the need for networks. Network Types: LAN, WAN, MAN, PAN, and their characteristics. Network Topologies: Bus, Star, Ring, Mesh, and Hybrid.

Network Models: OSI Model, TCP/IP Model, comparison between models.

Transmission Media: Guided Media: Twisted Pair, Coaxial Cable, Fiber Optics. Unguided Media: Radio waves, Microwaves, Infrared.

Network Devices : Hubs, Switches, Bridges, Routers, Gateways.

#Exemplar/Case Studies: Enterprise Network Design – A Case of a Corporate Office

Unit IV - Network Protocols and Application Layer Protocol - (09 Hours)

Network Protocols: IP, ICMP, ARP, TCP, UDP – features and comparisons

IP Addressing: IPv4 vs. IPv6, Subnetting concept. Routing Concepts, Routing Algorithms Routing Protocols: RIP, OSPF, BGP.

Application Layer Protocols: HTTP, FTP, SMTP, DNS

#Exemplar/Case Studies : IPv6 Migration – A Case of an Internet Service Provider

Unit V - Network Security and Management - (09 Hours)

Network Security: Threats and Vulnerabilities: Common network security issues. Cryptography: Symmetric and asymmetric encryption techniques. Firewalls and VPNs: Mechanisms to secure networks.

Network Management: SNMP: Simple Network Management Protocol and its operations. Network Monitoring Tools: Tools like Wireshark and NetFlow.

#Exemplar/Case Studies : Cybersecurity Threats and Defense Mechanisms – A Case of a Banking Network

Learning Resources

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, WILEY, ISBN 978-1-118-06333-0 , 9th Edition
2. Andrew S. Tenenbaum, “Computer Networks”, PHI, ISBN 81-203-2175-8.
3. Fourauzan B., "Data Communications and Networking", 5 th Edition, Tata McGraw- Hill, Publications, ISBN: 0 – 07 – 058408 – 7

Reference Books:

1. Andrew S. Tanenbaum & Herbert Bos, Modern Operating System, Pearson, ISBN-13: 9780133592221, 4th Edition
2. Kurose, Ross “Computer Networking a Top Down Approach Featuring the Internet”, Pearson, ISBN- 10: 0132856204
3. Efraim Turban, Linda Volonino, Gregory R. Wood “Computer Networking a Top Down Approach Featuring the Internet”, 10th Edition, Wiley; ISBN 13: 978-1-118-96126-1s
4. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, ISBN-10: 0-13- 380591-3, ISBN-13: 978-0-13-380591-8, 8th Edition

e-Books:

1. https://intronetworks.cs.luc.edu/current/ComputerNetworks.pdf
2. https://archive.org/details/operatingsystemconcepts10th

MOOC Courses:

1. https://onlinecourses.nptel.ac.in/noc21_cs18/preview
2. https://onlinecourses.nptel.ac.in/noc23_cs123/preview

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-552-MCA : Database Management System		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Basics of Computers
- Basic Mathematics

Companion Course: Database Laboratory

Course Objectives: The course aims to:

1. Understand the fundamental concepts of Database Management Systems (DBMS), including database design, models, and architecture.
2. Develop proficiency in SQL and PL/SQL for efficient data manipulation and querying.
3. Apply principles of normalization and relational database design to ensure data integrity and eliminate redundancy.
4. Understand transaction management, concurrency control, and recovery mechanisms for reliable database operations.
5. Explore modern database technologies, including NoSQL, Big Data, and Distributed Databases with real-world applications.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Design** relational databases using ER modeling and normalization techniques.
- **CO2: Make** use of SQL and PL/SQL queries for data retrieval, manipulation, and transaction processing.
- **CO3: Apply** normalization and indexing techniques to improve database performance and integrity.
- **CO4: Apply** transaction management and concurrency control to ensure ACID compliance in real-world applications.
- **CO5: Analyze** and utilize modern database technologies, including NoSQL, Big Data, and Distributed Databases, for scalable solutions.

Course Contents

Unit I - Introduction to Database Systems - (09 Hours)

Purpose and applications of DBMS, Data models: Hierarchical, Network, Relational, Database architecture and data independence, Data Modeling: Entity-Relationship (ER) modeling: Entities, attributes, relationships, constraints, keys, Design Process: Enhanced ER (EER) modeling: Generalization, specialization, aggregation.

 # Exemplar /Case Studies: Design an ER diagram for a university database capturing entities like students, courses, and instructors.

Unit II - Structured Query Language (SQL) and Procedural Extensions - (09 Hours)

SQL: Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL), Advanced SQL features: Joins, sub queries, set operations, aggregate functions, Views: Creation, modification, and deletion, Indexes: Purpose and implementation, Procedural extensions: PL/SQL basics, stored procedures, functions, triggers, cursors.

#Exemplar/Case Studies: Develop SQL queries and PL/SQL procedures for a library management system, including book inventory and member transactions.

Unit III - Relational Database Design and Normalization - (09 Hours)

Relational model concepts: Relations, tuples, attributes, CODD's rules, Integrity constraints: Domain, entity, referential, Functional dependencies and normalization: 1NF, 2NF, 3NF, BCNF, Designing efficient relational schemas.

#Exemplar/Case Studies: Normalize the schema of a sales database to eliminate redundancy and update anomalies

Unit IV - Transaction Management and Concurrency Control - (09 Hours)

Transactions: Properties (ACID), states, and lifecycle, Concurrency control mechanisms: Lock-based protocols, timestamp-based protocols, Deadlock handling: Detection, prevention, and avoidance, Recovery techniques: Log-based recovery, checkpoints, shadow paging, Isolation levels and their impact on transaction behavior.

#Exemplar/Case Studies : Analyze concurrency issues in an online banking system and propose solutions to maintain data consistency.

Unit V - Emerging Trends in Databases - (09 Hours)

Distributed databases: Architecture, data distribution strategies, and challenges, **NoSQL databases:** Types (document, key-value, column-family, graph), use cases, and CAP theorem, Big Data technologies: Introduction to Hadoop ecosystem, HDFS, MapReduce paradigm, Database security: Authentication, authorization, encryption techniques, Cloud databases: Benefits, challenges, and service models (DBaaS)

#Exemplar/Case Studies : Evaluate the suitability of a NoSQL database for a social media application handling large volumes of unstructured data.

Learning Resources

Text Books:

1. Database System Concepts Authors: Abraham Silberschatz, Henry F. Korth, S. Sudarshan Edition: 7th Edition Publisher: McGraw-Hill ISBN: 978-0078022159
2. Fundamentals of Database Systems Authors: Ramez Elmasri, Shamkant B. Navathe Edition: 7th Edition Publisher: Pearson ISBN: 978-0133970777
3. An Introduction to Database Systems Author: C.J. Date Edition: 8th Publisher: Pearson ISBN: 978 – 0321197849

Reference Books:

1. "Database Management Systems" – Raghu Ramakrishnan and Johannes Gehrke (McGraw Hill) ISBN: 978-9339213114
2. "SQL, PL/SQL: The Programming Language of Oracle" – Ivan Bayross (BPB Publications) ISBN 978-8176569644
3. NoSQL Distilled Authors: Pramod J. Sadalage, Martin Fowler eBook ISBN: 9780133036121 Available at: Safari Books Online (O'Reilly), Kindle

e-Books:

1. Coursera – Database Management Essentials – https://www.coursera.org/
2. W3Schools SQL Tutorial – https://www.w3schools.com/sql/

MOOC Courses:

1. NPTEL Online Courses – https://nptel.ac.in/

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-553-MCA : Software Engineering and Project Management		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses: Basics of Computers

Companion Course: NA

Course Objectives: The course aims to:

1. To understand software development and software lifecycle process models.
2. To know methods of capturing, specifying, visualizing and analyzing software requirements.
3. To introduce principles of agile software development, the SCRUM process and agile practices.
4. To learn about project planning, execution and tracking.
5. To understand project management, leadership its role and importance in successfully managing IT projects.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Choose** and **apply** appropriate lifecycle model of software development.
- **CO2: Analyze** software requirements by applying various modelling techniques.
- **CO3: Describe** principles of agile development, discuss the SCRUM process and distinguish Agile process model from other process models.
- **CO4: Describe** project schedule and cost estimation.
- **CO5: Understand** IT project management, define ethics and understand its importance in project leadership.

Course Contents

Unit I - Introduction to Software Engineering - (09 Hours)

Nature of Software, Software Process, Software Engineering Practices, Software Myths, Generic Process Model, Analysis and comparison of Process Models: Waterfall Model, Incremental Models, Evolutionary Models, Concurrent, Specialized Process Models, Personal and Team Process Models, Introduction to Clean Room Software Engineering. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, CMM Models.

Hands on Activity:

Using a visual tool of your choice (e.g., word , Excel, Draw.io, Lucidchart, or any diagramming software), draw and label the Software Development Life Cycle (SDLC) for one of the following software process models . (Waterfall Model , Iterative Model, Incremental Model, Spiral Model , Agile Model (e.g., Scrum).

Your diagram should clearly show:

- All phases of the SDLC relevant to your chosen model.
- Sequence or flow of phases using arrows or connectors.
- A short description or output artifact for each phase.
- Any feedback loops, iterations, or special characteristics of the model.

 # Exemplar /Case Studies: Online Library Management System – A Journey through Software Engineering Principles.

Unit II - Requirement Analysis - (09 Hours)

Requirements Capturing: requirements engineering (elicitation, specification, validation, negotiation, prioritizing requirements (Kano diagram) - real life application using case study. Requirements Analysis: basics, scenario based modelling, UML models: use case diagram and class diagram, data modelling, data and control flow model, behavioral modelling using state diagrams - real life application case study, Software Requirement specification(SRS).

Hands on Activity:

1. Choose a real-life or hypothetical software system from the list below.

- Online Library Management System
- Student Attendance System
- Online Food Ordering App

2. Draw the following UML diagrams using any of these tools (Draw.io, StarUML, Lucidchart) Choose at least 3 UML diagrams from:

- Use Case Diagram
- Class Diagram
- Sequence Diagram
- Activity Diagram
- State Diagram

#Exemplar/Case Studies: Online Voting System for Student Council.

Unit III - Agile Development Process - (09 Hours)

Agile Development: Agile manifesto, agility and cost of change, agility principles, myth of planned development, toolset for the agile process. Extreme Programming: XP values, process, industrial XP, SCRUM

- process flow, scrum roles, scrum cycle description, product backlog, sprint planning meeting, sprint backlog, sprint execution, daily scrum meeting, maintaining sprint backlog and burn-down chart, sprint review and retrospective.

Agile Practices: test driven development, refactoring, pair programming, continuous integration, exploratory testing versus scripted testing.

Hands on Activity:

Create a well-formatted table in Word comparing Agile and Waterfall across 5–6 dimensions. Columns should be Aspect, Agile Approach, Waterfall Approach, and Example.

#Exemplar/Case Studies: Online Course Feedback and Analysis System

Unit IV - Project Planning - (09 Hours)

Project initiation, Planning Scope Management, Creating the Work Breakdown Structure, Effort estimation and scheduling: Importance of Project Schedules, Estimating Activity Resources, Estimating Activity Durations, Developing the Schedule using Gantt Charts, Adding Milestones to Gantt Charts, Using Tracking Gantt Charts to Compare Planned and Actual Dates, Critical Path Method, Program Evaluation and Review Technique (PERT) with examples. Planning Cost Management, Estimating Costs, Types of Cost Estimates, Cost Estimation Tools and Techniques, Typical Problems with IT Cost Estimates.

Project Leadership: Approaches to Leadership, Leadership Styles, Emotional Intelligence, Ethics in Projects: Ethical Leadership, Common Ethical Dilemmas, Making Sound Ethical Decisions, Codes of Ethics and Professional Practices

Hands on Activity:

1. Create a Work Breakdown Structure (WBS) for a project of your choice (e.g., website development, event planning, or software development). You can use Excel or Word to create a clear hierarchical WBS structure with tasks and sub-tasks.
2. Using Excel, create a Gantt chart for a small project (e.g., organizing a college event). Add at least 5 milestones to the chart and track progress.

#Exemplar/Case Studies : Online Examination Management System.

Unit V - Project Management - (09 Hours)

Project monitoring and control: tools for project management, Software tools like Microsoft project management or any other open source tools. The Importance of Project Quality Management: Planning Quality Management, Performing Quality Assurance, Controlling Quality, Tools and Techniques for Quality Control (Statistical control, Six Sigma) Risk Analysis & Management: Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Risks Monitoring and Management, The RMMM plan for case study project. Software Configuration Management: The SCM repository, SCM process, Version Control and Change Control, SCM tools- GitHub or others, Configuration management for Web Apps. Maintenance & Reengineering: Software Maintenance, Reengineering, Business Process Reengineering.

Hands on Activity:

Create a GitHub account and set up a basic repository for a simple project.

#Exemplar/Case Studies : E-Learning Portal for Skill Development.

Learning Resources

Text Books:

1. Roger S Pressman, Software Engineering: A Practitioner's Approach, McGraw-Hill, Seventh or Eighth Edition. ISBN-13: 978-0078022128

2. Joseph Phillips, IT Project Management –On Track From Start to Finish, Tata McGraw-Hill, ISBN13: 978-0-07106727-0, ISBN-10: 0-07-106727-2

3. John m. Nicolas and Herman Steyn, Project Management for Engineering, Business and Technology,4th Edition, Elsevier, ISBN 978-0-08-096704-2

Reference Books:

1. Pankaj Jalote, Software Engineering: A Precise Approach, Wiley India, ISBN: 9788126523115.

2. Marchewka, Information Technology Project Management, Wiley India, ISBN: 9788126543946

e-Books:

1. https://www.e-booksdirectory.com/listing.php?category=25

2. https://www.vumultan.com/.../CS605Software%20Engineering%20Practitioner's%20Appro.

3. http://originaldocs.net/pdf/software-engineering-book-by-pressman-7th-edition-free-download.pdf

MOOC Courses:

1. MOOC Courses: https://www.classcentral.com/course/swayam-software-engineering-14293

2. NPTEL Courses: https://www.nptelvideos.in/2012/11/software-engineering.html

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PCC-554-MCA : Java and Advance Java Programming		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses: Basic Computer Programming Concept.

Companion Course: Java Programming Laboratory

Course Objectives: The course aims to:

1. To learn the core concept of Java programming.
2. To introduce the working environment of Java Program using the multithreading and AWT, Swing.
3. To study the use of Network programming and JDBC in Java Programming.
4. To gain knowledge of Java Servlet and JSP concepts in Java.
5. To familiarize students with the concepts of Spring.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Illustrate** the concept of Core Java programming.
- **CO2: Apply** the concepts of multithreading and GUI (AWT, Swing) Programming for real world Applications
- **CO3: Explain** the concept of network programming and JDBC.
- **CO4: Develop** server side applications using servlet & JSP.
- **CO5: Demonstrate** the concept of Spring.

Course Contents

Unit I - An Introduction to Java Programming - (09 Hours)

An Introduction to Java: A Short History of Java, Features of Java,Creating and Running Java Programs using Command Line Arguments and IDE, Programming Construct Decision making statement, switch statement, looping statement, Class and Object Defining a class, adding variables, Adding Methods, Creating Objects, Accessing Class Members, Constructors, Object Oriented concepts with respect to Java.

Inheritance: Extending a class, Overriding Method, using super, Final variable and Methods, this keyword, Interfaces, Packages, Exception Handling.

 # Exemplar /Case Studies: Study of Java Program compilation Process, testing and debugging.

Unit II - Multithreading And AWT - (09 Hours)

Multithreading: Multithreading concepts, Thread Life cycle, Creating multithreaded application, Thread priorities, Thread synchronization.

Abstract Windows Toolkit: Components and Graphics, Containers, Frames and Panels, Layout Managers, AWT basic components.

Event delegation Model: Event source and handler, Event categories, Listeners, interfaces, Anonymous classes.

Swing: Containers, JFrames and JPanels, Layout Managers, Swing basic components.

Swing Libraries: Model view Controller design pattern, Different layout, menus dialog boxes, text input.

#Exemplar/Case Studies: Design Multithreaded GUI Application Using AWT & Swing.

Unit III - Networking with java and JDBC - (09 Hours)

Networking basics: Sockets, port, Proxy servers, Internet addressing 7 URL, java.net – networking classes and interfaces, Implementing TCP/IP based Server and Client, Datagrams – Datagram packet, Datagram server and client, URL connections.

JDBC: Java database connectivity, Types of JDBC drivers, Writing first JDBC applications, Types of statement objects ,Statement, Prepared Statement and Callable Statement, Types of resultset, ResultSet Metadata, Inserting and updating records, Connection pooling.

#Exemplar/Case Studies: Developing a Networked File Management System with Database Support

Unit IV - Servlet And JSP - (09 Hours)

Servlet: Introduction, Web application Architecture, Life cycle of servlet, Handling HTTP Get Request, Handling HTTP Post Request.

Servlet Communication: Servlet-Browser CommunicationsendError, setHeader, sendRedirect, Web-Component Communication- Forward, Include, Tracking Mechanisms: Session Object, Cookies, URL-Rewriting, Hidden Form Fields.

Introduction to JSP: Getting Familiar with JSP Server, First JSP Page, Adding Dynamic contents via expressions, Scriptlets, Mixing Scriptlets and HTML Directives, Declaration, Tags and Session.

#Exemplar/Case Studies : Study of Web Servers.

Unit V - Spring - (09 Hours)

Spring: Overview of spring, Spring MVC Architecture, Spring MVC Flow, the core spring module, wiring beans, aspect-oriented spring, database and managing transactions with swing, building web applications with spring MVC, Spring Form Handling, working with remote services, Managing spring with JMX.

#Exemplar/Case Studies : Building Web Applications with Spring MVC.

Learning Resources

Text Books:

1. E. Balagurusamy, “Programming with Java A primer”, Sixth Edition, McGraw Hill Education (India) Private Limited, ISBN 93-5134-320-0.
2. Herbert Schildt, “JAVA Complete Reference”, 9th Edition, McGraw Hill Education (India) Private Limited, ISBN: 93-392-1209-6.
3. Steven Horlzner, “Java 2 programming black book”, Dreamtech Press, ISBN 978-93-5119-940-3.

Reference Books:

1. Eckel B., "Thinking in Java", 4th Edition, Pearson Education,ISBN 0-13-187248-6
2. Herbert Schildt,“Complete Reference Java2” ,5th edition,McGraw-Hill, DOI: 10.1036/007222858X
3. Cay S Horstmann, Gary Cornell,”Core Java 2 Volume - I-Fundamentals”,9th edition,ISBN-13: 978-0- 13-708189-9
4. Cay S Horstmann, Gary Cornell,”Core Java 2 Volume - II-Advanced Features” ,9th edition,ISBN-13: 978-0-13-708160-8
5. Chad Darby, John Griffin & others.,”Beginning Java Networking”,ISBN 13: 9781861005601

e-Books:

1. http://tutorialpoint.com
2. https://www.tpointtech.com/java-tutorial

MOOC Courses:

1. https://nptel.ac.in/courses/106105191
2. https://moocfi.github.io/courses/2013/programming-part-1/
3. https://java-programming.mooc.fi/

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PEC-570A-MCA : Elective II-Machine Learning		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses: Mathematical foundation for Computer Application (MJC-501-MCA).

Companion Course: NA

Course Objectives: The course aims to:

1. To build a strong foundation in machine learning principles and techniques.
2. To explore and understand different machine learning algorithms.
3. To gain an understanding of logic-based and algebraic models used in machine learning.
4. To stay informed about emerging trends and developments in machine learning.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Explain** fundamental concepts and terminologies of Machine Learning.
- **CO2: Apply** different regression and generalization techniques.
- **CO3: Compare** and contrast different classification algorithms and apply them to appropriate problems.
- **CO4: Utilize** unsupervised learning methods to discover hidden patterns and structure in unlabeled data.
- **CO5: Analyze** current trends and advancements in Machine Learning and discuss their implications.

Course Contents

Unit I - Introduction to Fundamentals of Machine Learning - (09 Hours)

Introduction: What is Machine Learning, Examples of Machine Learning applications, Training versus Testing, Positive and Negative Class, Cross-validation. **Types of Learning:** Supervised, Unsupervised and Semi- Supervised Learning. Dimensionality Reduction: Introduction to Dimensionality Reduction, Subset Selection, Introduction to Principal Component Analysis.

 # Exemplar /Case Studies: Reducing Dimensionality in Sensor Data.

Unit II - Supervised Learning: Regression - (09 Hours)

Introduction to Regression: Definition, types, and use cases, **Linear Regression:** Theory, cost function, gradient descent, and assumptions, **Polynomial Regression:** Adding polynomial terms, degree selection, overfitting, underfitting, Lasso and Ridge Regression: Regularization techniques for controlling model complexity, Support Vector Regression (SVR), Evaluation metrics for regression models: Mean Squared Error (MSE), R-squared, and Mean Absolute Error (MAE).

#Exemplar/Case Studies: Predicting Marketing ROI.

Unit III - Supervised Learning: Classification - (09 Hours)

Introduction to Classification: Definition, types, and use cases, **Logistic Regression:** Theory, logistic function, binary and multiclass classification , **Classification algorithms:** Naïve Bayes classifier, k-Nearest Neighbor (KNN), Support vector machines, Decision Trees: Construction, splitting criteria, pruning, and visualization, Random Forests: Ensemble learning, bagging, and feature importance, Evaluation metrics for classification models: Accuracy, Precision, Recall, F1-score, and ROC curves.

#Exemplar/Case Studies: Email Spam Detection System or Sentiment Analysis of Customer Reviews.

Unit IV - Unsupervised Learning - (09 Hours)

Unsupervised Learning: Definition, applications, challenges, Clustering algorithms: K-means clustering, Agglomerative hierarchical clustering- DBSCAN, Anomaly detection: K nearest neighbor (KNN), Association rule: Apriori algorithm, FP- growth algorithm, Dimensionality Reduction, PCA., Supervised learning vs Unsupervised learning vs Semi- Supervised learning.

#Exemplar/Case Studies : Customer Segmentation using K-means clustering.

Unit V - Current trends in Machine Learning - (09 Hours)

Reinforcement Learning: Exploration, Exploitation, Rewards, Penalties, **Deep Learning:** The Neuron, Expressing Linear Perceptron as Neurons, Feed Forward Neural Networks, Linear Neurons and their Limitations, Sigmoid, Tanh and ReLU Neurons, Generative Models: GAN and its applications, Natural Language Processing (NLP): transformer models and their application, Explainable AI (XAI).

#Exemplar/Case Studies : Transfer Learning and Pretrained Models.

Learning Resources

Text Books:

1. Introduction to Machine Learning by Ethem Alpaydin, ISBN- 978-0262028189, PHI.
2. Machine Learning by Tom Mitchell. ISBN- 0070428077, McGraw- Hill.

Reference Books:

1. Machine Learning, Saikat Dutt, S. Chjandramouli, Das, Pearson
2. An Introduction to Statistical Learning, Gareth James Daniela Witten Trevor Hastie Robert Tibshirani
3. Machine Learning: A Probabilistic Perspective by Kevin P. Murphy
4. Hands-On Unsupervised Learning with Python by Ankur A. Patel

5. Reinforcement Learning: An Introduction by Richard S. Sutton and Andrew G. Barto
6. Pattern Recognition and Machine Learning" by Christopher M. Bishop

e-Books:

1. Machine Learning for Absolute Beginners: A Plain English Introduction (Third Edition) (mrce.in)
2. Machine - Learning - Tom Mitchell.pdf (cmu.edu)
3. Machine Learning - The Art and Science of Algorithms that Make Sense of Data (put.poznan.pl)

MOOC Courses:

NPTEL Swayam Course:

1. https://onlinecourses.nptel.ac.in/noc20_cs29/preview
2. https://onlinecourses.nptel.ac.in/noc24_cs81

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PEC-570B-MCA : Elective II-Internet of Things		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Hardware Basics
- Basic programming skills.

Companion Course: NA

Course Objectives: The course aims to:

1. Understand the core concepts, architecture, and components of IoT systems.
2. Develop hands-on skills in IoT project design using Arduino and Raspberry Pi.
3. Explore cloud integration, data analytics, and secure communication in IoT.
4. Analyze real-world IoT applications, challenges, and emerging trends

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Demonstrate** a clear understanding of IoT architecture, components, and communication protocols.
- **CO2: Develop** functional IoT prototypes using Arduino and Raspberry Pi for real-world applications.
- **CO3: Understand** IoT systems with cloud platforms to collect, analyze, and visualize data securely.
- **CO4: Design** and **Develop** secured IoT applications.
- **CO5: Evaluate** IoT applications across industries and propose innovative solutions to emerging challenges.

Course Contents

Unit I - Fundamentals of IoT - (09 Hours)

Overview of IoT: Definition and Characteristics, IoT Architecture, Components of IoT: Things-devices and sensors, Physical Design of IoT & Logical Design of IoT, Communication technologies, Data processing and storage, Overview of protocols in IoT -MQTT, CoAP, HTTP.

 # Exemplar /Case Studies: Smart Manufacturing.

Unit II - IoT Devices and Connectivity - (09 Hours)

IoT Sensors and Actuators: Types and functionalities, Interfacing sensors with devices,

Connectivity and Communication: IoT networking technologies-Wi-Fi, Bluetooth, Zigbee, LoRa, 5G, Role of gateways in IoT networks, Challenges in IoT connectivity, Power management in IoT devices.

#Exemplar/Case Studies: Supply Chain Management.

Unit III - IoT Data and Cloud Integration - (09 Hours)

Data Collection and Processing: Data types and formats in IoT, Data analytics in IoT, IoT and Cloud Computing: Cloud platforms for IoT-AWS IoT, Google Cloud IoT, Microsoft Azure IoT, Edge computing and its importance Security and **Privacy Challenges:** Common vulnerabilities in IoT systems, best practices for securing IoT networks.

#Exemplar/Case Studies: Transportation/ Logistics.

Unit IV - Security and Privacy - (09 Hours)

Challenges and threats to IoT security, IoT Security Requirements, IoT Three-Domain Architecture, Cloud Domain Attacks and Counter measures, Fog Domain Attacks and Counter measures, Sensing Domain Attacks and Counter measures, Summary and Future Directions.

#Exemplar/Case Studies : Health Care.

Unit V - IoT Applications and Future Trends - (09 Hours)

Key IoT Application Areas: Smart homes and cities, Industrial IoT-IIoT and smart manufacturing, Healthcare and wearable's, Agriculture, and environmental monitoring Challenges and Opportunities in IoT Development.

Emerging Trends: AI in IoT, Block chain for IoT security, IoT in sustainability.

#Exemplar/Case Studies : Smart Governance.

Learning Resources

Text Books:

1. Charles Crowell, "IoT-Internet of Things for Beginners: An Easy-to-Understand Introduction to IoT",ISBN-13 : 979-8613100194.
2. Internet of Things (A Hands-on-Approach) ISBN: 978-0996025515 -by ArshdeepBahga and Vijay Madisetti.

Reference Books:

1. David Hanes, Gonzalo Salgueiro, Robert Barton, Jerome Henry, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", Cisco Press, ISBN-13: 978-1- 58714-456-1 ISBN-10: 1-58714-456-5.

2. Olivier Hersent, David Boswarthick, Omar Elloumi , “The Internet of Things – Key applications and Protocols”, Wiley, 2012, ISBN:978-1-119-99435-0.

3. David Hanes, IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the

4. Internet of Things, Cisco Press, ISBN-13: 978-1-58714-456-1, ISBN-10: 1-58714-456-5, 2017.

e-Books:

1. https://nibmehub.com/opac-service/pdf/read/IoT%20Fundamentals.pdf

2. https://www.iotforall.com/ebooks/an-introduction-to-iot

3. http://www.cse.wustl.edu/~jain/cse570-15/ftp/iot_prot/index.html

MOOC Courses:

1. https://onlinecourses.nptel.ac.in/noc25_cs44/preview

2. https://nptel.ac.in/courses/106/105/106105166/

3. https://www.udemy.com/course/a-complete-course-on-an-iot-system-design- anddevelopment/

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025 Pattern)		
PEC-570C-MCA : Elective II-Data Mining and Data warehousing		
Data warehouse	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 50 Marks **End-Semester**: 50 Marks
Comprehensive Continuous Evaluation	**Unit Test (UT)**- 20 Marks **Assignments / Case Study** - 20 Marks **Seminar Presentation / Open Book Test/ Quiz** -10 Marks	

Prerequisite Courses:

- Basics of Computers
- Basic Data Base Management System and mathematics

Companion Course: NA

Course Objectives: The course aims to:

1. To Gain knowledge of the basic concepts of data mining and data preprocessing.
2. To explain the concept of Data Warehousing and OLAP.
3. To compare different data mining techniques including association rule mining, classification, clustering, and outlier detection.
4. To learn the performance of various Classification techniques.
5. To explain the performance of various Clustering techniques.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Define** the fundamental concepts of data mining and data preprocessing techniques.
- **CO2: Explain** the architecture, design, and components of a Data Warehouse and OLAP.
- **CO3: Develop** frequent pattern mining techniques, including association rule mining and Apriori algorithm
- **CO4: Apply** classification and prediction techniques such as Decision Trees, Bayesian Classifiers, and SVM
- **CO5: Implement** clustering techniques and outlier detection methods.

Course Contents

Unit I - Fundamentals of data mining and Data Pre-processing- (09 Hours)

Introduction to Data Mining- Motivation, Importance, Definition of Data Mining, Kind of Data. Data Mining Functionalities and Applications. Architecture of Data Mining Systems

Data Preprocessing – Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization and Concept Hierarchy Generation.

 # Exemplar /Case Studies: Netflix – Personalized Recommendation System.

Unit II - Data Warehouse and OLAP - (09 Hours)

Data warehousing – Basic concepts, Components, Building a Data warehouse. Data Warehouse Architecture and DBMS Schemas for Decision Support. Data Extraction, Transformation and Loading (ETL). Data Quality and Metadata, Query tools and Applications.

Online Analytical Processing (OLAP) – OLAP and Multidimensional Data Analysis.

#Exemplar/Case Studies: Walmart – Retail Data Warehousing for Inventory Management.

Unit III - Association Rules - (09 Hours)

Association Rule Mining: Concepts and Applications, Efficient and Scalable Frequent Item set Mining Methods, Market Basket Analysis and Apriori Algorithm, Tree-Based Algorithms for Association Mining, Association Mining to Correlation Analysis Constraint-Based Association Mining.

#Exemplar/Case Studies: Amazon – Market Basket Analysis for Product Recommendations.

Unit IV - Classification and Prediction - (09 Hours)

Classification and Prediction: Issues Regarding Classification and Prediction , Classification by Decision Tree Induction , Bayesian Classification , Rule Based Classification , Classification by Back propagation , Support Vector Machines: Associative Classification Other Classification Methods, Prediction – Accuracy and Error Measures, Evaluating the Accuracy of a Classifier or Predictor.

#Exemplar/Case Studies : JPMorgan Chase – Fraud Detection in Banking.

Unit V - Clustering Methods and Outlier Analysis - (09 Hours)

Cluster Analysis: Problem Definition, Clustering Overview, Evaluation of Clustering Algorithms, Partitioning Clustering-K-Means Algorithm, K-Means Additional issues,

Hierarchical Clustering-Agglomerative Methods and divisive methods, Basic Agglomerative Hierarchical Clustering Algorithm, Specific techniques, Key Issues in Hierarchical Clustering, Strengths and Weakness; Outlier Detection.

#Exemplar/Case Studies : Google – User Segmentation for Targeted Advertising.

Learning Resources

Text Books:

1. Jiawei Han & Micheline Kamber, Data Mining – Concepts and Techniques, 3rd Edition, Morgan Kaufmann Publishers, Elsevier, 2012. ISBN-13 : 978-0123814791

2. Margaret H Dunham, Data Mining Introductory and Advanced Topics, 2nd edition, Pearson Education, New Delhi, India,2006. ISBN-13 : 978-8177587852

3. Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press. Zaki, M. J., & Meira Jr., W. (2014). ISBN: 978-0521766333.

Reference Books:

1. Arun K Pujari, Data Mining Techniques, 3rd edition, Universities Press, 2013. ISBN : 978-938623505
2. Introduction to Data Mining (2nd ed.). Pearson. an, P.-N., Steinbach, M., Karpatne, A., & Kumar, V ISBN: 978-0133128901.

e-Books:

1. https://charuaggarwal.net/Data-Mining.pdf
2. https://theswissbay.ch/pdf/Gentoomen%20Library/Data%20Mining/Dunham%20-%20Data%20Mini

MOOC Courses:

1. https://onlinecourses.nptel.ac.in/noc21_cs06/preview
2. https://onlinecourses.swayam2.ac.in/cec19_cs01/preview

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
PCC-555-MCA : Java Programming Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 04 Hours/Week	02	**Term Work** : 25 Marks **Practical** : 50 Marks

Prerequisite Courses: Basic Computer Programming Concept

Companion Course: Java and Advance Java Programming

Course Objectives: The course aims to:

1. To learn the core concept of Java programming.
2. To introduce the working environment of Java Program using the multithreading and AWT, Swing.
3. To study the use of Network programming and JDBC in Java Programming.
4. To gain knowledge of Java Servlet and JSP concepts in Java.
5. To familiarize students with the concepts of Spring

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Apply** core concept of Java programming to write program
- **CO2: Construct** application by using multithreading and AWT.
- **CO3: Develop** java application using networking and JDBC.
- **CO4: Design** server side applications using servlet & JSP.
- **CO5: Demonstrate** the concept of Spring.

List of Laboratory Experiments/Assignments (Any 10 laboratory assignments) based on Programming

Sr. No.	Practical Assignments
1	Create a hierarchy of Employee, Manager, Sales Manager, they should have the following functionality: a) Employee: Display name, date of birth and id of Employee. b) Manager: Display all above information with salary drawn. Sales Manager: Display all above information and commission if applicable

2	Write a Java program to calculate salary using packages. Creates a package employee and create a class Emp. Data members are name, employee id, category, basic pay, HRA, DA, net pay, provident fund, gross pay, income tax, and allowance. Calculate the values in methods. Call the methods to perform and print values.
3	Write a program to accept roll no. Marks from users. Throw user defined exception Marks out of Bound if marks are 100.
4	Write a program to implement student information in a file and perform the operations on it.
5	Write a program to design an admission enquiry form using Swing.
6	Write a client-server program using TCP sockets to echo the message send by the client.
7	Write a program to connect to any database and to execute the SQL query operation using GUI Interface.
8	Write a program of servlet code to demonstrate GET and POST Methods with suitable example.
9	Create a JSP page that displays the current date and time using JSP scripting elements.
10	Create a user registration system using JSP and Servlet with MySQL database.
11	Write a program to demonstrate how the process tasks asynchronously in a Spring Boot application.
12	Write a program to set up Spring Boot with JPA and connect to an in-memory database like

Mini Project Topic (Any one project/ concept)

1. Calculator with basic functions. Add more functionality such as graphic user interface and
2. Complex calculations.
3. Design Registration page using AWT and JDBC.
4. Design Admission Enquiry Form Using Swing
5. Client-Server Program Using TCP Sockets

Learning Resources

Text Books:

1. E. Balagurusamy, “Programming with Java A primer”, Sixth Edition, McGraw Hill Education (India) Private Limited, ISBN 93-5134-320-0.
2. Herbert Schildt, “JAVA Complete Reference”, 9th Edition, McGraw Hill Education (India) Private Limited, ISBN: 93-392-1209-6.
3. Steven Horlzner, “Java 2 programming black book”, Dreamtech Press, ISBN 978-93-5119-940-3.

Reference Books:

1. Eckel B., "Thinking in Java", 4th Edition, Pearson Education,ISBN 0-13-187248-6
2. Herbert Schildt,“Complete Reference Java2” ,5th edition,McGraw-Hill, DOI: 10.1036/007222858X
3. Cay S Horstmann, Gary Cornell,”Core Java 2 Volume - I-Fundamentals”,9th edition,ISBN-13: 978-0- 13-708189-9
4. Cay S Horstmann, Gary Cornell,”Core Java 2 Volume - II-Advanced Features” ,9th edition,ISBN-13: 978-0-13-708160-8
5. Chad Darby, John Griffin & others.,”Beginning Java Networking”,ISBN 13: 9781861005601

MOOC Courses:

1. https://nptel.ac.in/courses/106105191
2. https://java-programming.mooc.fi/

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
PCC-556-MCA : Database Laboratory		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week	01	**Oral** : 50 Marks

Prerequisite Courses:

- Basics of Computers
- Basic Mathematics

Companion Course: Database Management System

Course Objectives: The course aims to:

1. To understand the principles of database design using Entity-Relationship modeling and schema normalization.
2. To develop proficiency in SQL for data definition, manipulation, and retrieval using queries, joins, and set operations.
3. To apply procedural extensions of SQL such as PL/SQL for creating stored procedures, functions, triggers, and cursors.
4. To study and implement techniques for transaction management, concurrency control, deadlock handling, and database recovery.
5. To evaluate modern database technologies including NoSQL systems and big data frameworks like Hadoop for large-scale data processing

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Design** and implement Entity-Relationship (ER) models and convert them into normalized relational schemas.
- **CO2: Develop** and execute SQL queries including SELECT statements with various clauses and joins, as well as nested queries and set operations.
- **CO3: Create** and manage PL/SQL components such as stored procedures, functions, triggers, and cursors for automated database operations.
- **CO4: Apply** concepts of transaction management, concurrency control, and recovery techniques to ensure ACID properties and data consistency.
- **CO5: Evaluate** NoSQL and big data technologies by implementing CRUD operations in MongoDB and processing large datasets using Hadoop MapReduce.

List of Laboratory Experiments (Any 6 to 8 laboratory experiments) based on Programming

Sr. No.	Practical Assignments
1	ER Diagram Design: • Create an ER model for a university management system including students, courses, instructors, and enrollments. • Convert the ER model into relational tables.
2	Basic SQL Queries: • Create a database for an online shopping system. • Define tables with primary keys, foreign keys, constraints. • Perform basic insert, update, and delete operations.
3	SQL Queries – SELECT and Joins: • Create a database for an online shopping system. • Implement SELECT queries using WHERE, ORDER BY, GROUP BY, HAVING clauses. • Perform INNER, OUTER, LEFT, RIGHT JOINs on related tables.
4	Nested Queries and Set Operations: • Create a database for an online shopping system. • Write nested queries using subqueries. • Perform set operations like UNION, INTERSECT, and MINUS.
5	Views, Indexing, and Aggregation: • Create and manage views for employee and department databases. • Use indexes to improve query performance. • Implement aggregate functions like COUNT, SUM, AVG, MAX, MIN.

6	PL/SQL – Stored Procedures & Functions: • Develop stored procedures for an attendance system to mark attendance automatically. • Create functions for calculating total sales in a retail database.
7	PL/SQL – Triggers & Cursors: • Implement before and after triggers for an inventory management system. • Use cursors to process multiple records row-by-row.
8	Normalization and Schema Refinement: • Design a student course database and normalize it to 3NF/BCNF. • Decompose a given denormalized relation into smaller relations.
9	Implementation of Functional Dependencies: • Identify functional dependencies in a library database. • Apply decomposition using FD rules.
10	Transaction Management – ACID Properties: • Implement a bank transaction system with withdraw, deposit, and transfer operations. • Ensure ACID compliance.

Mini Project Topic (Any one project/ concept)

1. Concurrency Control – Locking Mechanisms:
   - Simulate a concurrent transaction scenario.
   - Implement 2PL (Two-Phase Locking) protocols for handling concurrent transactions.
2. Deadlock Detection and Prevention:
   - Simulate a deadlock scenario in a hospital database.
   - Implement deadlock prevention techniques.
3. Database Recovery Techniques:
   - Implement shadow paging and log-based recovery mechanisms.

- Simulate crash recovery for an airline reservation system.

4. NoSQL Databases – MongoDB/Cassandra:

- Implement a document-oriented database using MongoDB.
- Perform CRUD operations (Create, Read, Update, Delete) on JSON documents.

5. Big Data Processing – Hadoop HDFS & MapReduce:

- Load and retrieve unstructured social media data into HDFS.
- Write a MapReduce job to count word frequency in a dataset.

Learning Resources

Text Books:

1. Database System Concepts Authors: Abraham Silberschatz, Henry F. Korth, S. Sudarshan Edition: 7th Edition Publisher: McGraw-Hill ISBN: 978-0078022159
2. Fundamentals of Database Systems Authors: Ramez Elmasri, Shamkant B. Navathe Edition: 7th Edition Publisher: Pearson ISBN: 978-0133970777
3. Database Management Systems Authors: Raghu Ramakrishnan, Johannes Gehrke Edition: 3rd Edition Publisher: McGraw-Hill ISBN: 978-0072465631
4. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence Authors: Pramod J. Sadalage, Martin Fowler Edition: 1st Edition Publisher: Addison-Wesley ISBN: 978-0321826626
5. Hadoop: The Definitive Guide Author: Tom White Edition: 4th Edition Publisher: O'Reilly Media, ISBN: 978-1491901632

Reference Books:

1. An Introduction to Database Systems Author: C.J. Date Edition: 8th Edition Publisher: Pearson ISBN: 978-0321197849
2. SQL: The Complete Reference Author: James R. Groff, Paul N. Weinberg Edition: 3rd Edition Publisher: McGraw-Hill ISBN: 978-0071592567
3. Oracle PL/SQL Programming Author: Steven Feuerstein Edition: 6th Edition Publisher: O'Reilly Media ISBN: 978-1449324452
4. MongoDB: The Definitive Guide Authors: Kristina Chodorow Edition: 3rd Edition Publisher: O'Reilly Media ISBN: 978-1491954461
5. Principles of Distributed Database Systems Authors: M. Tamer Özsu, Patrick Valduriez Edition: 4th Edition Publisher: Springer ISBN: 978-3319184626

E-Books Links

1. Database System Concepts (7th Edition) Authors: Abraham Silberschatz, Henry F. Korth, S. Sudarshan eBook ISBN: 9781260084504 Available at: McGraw-Hill Education Connect, VitalSource
2. Fundamentals of Database Systems (7th Edition) Authors: Ramez Elmasri, Shamkant B. Navathe eBook ISBN: 9780133970777 Available at: Pearson eText, Amazon Kindle, VitalSource

3. NoSQL Distilled Authors: Pramod J. Sadalage, Martin Fowler eBook ISBN: 9780133036121 Available at: Safari Books Online (O'Reilly), Kindle

4. Hadoop: The Definitive Guide (4th Edition) Author: Tom White eBook ISBN: 9781491901687 Available at: O'Reilly Online, Google Books, Amazon Kindle

5. SQL: The Complete Reference (3rd Edition) Authors: James R. Groff, Paul N. Weinberg eBook ISBN: 9780071592567 Available at: McGraw-Hill eBook Library, Amazon Kindle

MOOC Courses:

1. https://nptel.ac.in/courses/106105175
2. https://nptel.ac.in/courses/106101061
3. https://www.coursera.org/learn/sql-data-science
4. https://www.edx.org/course/database-system
5. https://www.udemy.com/course/the-complete-sql-bootcamp/

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
PCC-557-MCA : Project Based Learning		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week	02	**Term Work :** 25 Marks

Prerequisite Courses:

- Data Structures and Algorithms Laboratory
- OOP Laboratory
- Python Programming Laboratory

Companion Course: Data Science Laboratory, Lab Practice-I

Course Objectives: The course aims to:

1. **Develop Analytical and Problem-Solving Skills** - Apply critical thinking to design innovative, technology-driven solutions for real-world challenges.
2. **Understand and Apply the Software Development Life-cycle (SDLC)** - Use structured approaches, tools, and best practices to create efficient and scalable systems.
3. **Enhance Teamwork and Professional Skills** - Work collaboratively, demonstrating leadership, effective communication, and workplace readiness.
4. **Foster Innovation and Research Orientation** - Integrate entrepreneurial thinking and research-based approaches to create impactful solutions.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Identify** and **define** socially or industrially relevant problems.
- **CO2: Design** and **implement** effective, sustainable, and scalable solutions.
- **CO3: Exhibit** the ability to collaborate productively, uphold professional ethics, and convey ideas clearly.
- **CO4: Develop** and deliver industry-standard technical documentation and reports.

Course Execution details

Preamble

Project-Based Learning (PBL) is an instructional methodology that enables students to acquire in-depth knowledge and industry-relevant skills through the execution of meaningful, real-world projects. More than just completing assignments, PBL immerses learners in the process of investigating, analyzing, and solving authentic, complex challenges with sustained engagement. It is fundamentally "learning by doing", preparing students for a professional environment where successful project execution is a critical driver of innovation and progress.

PBL transforms the teacher's role from a traditional lecturer to that of a mentor, initiator, and facilitator, guiding students in collaborative, inquiry-driven learning. Under this model, students are challenged with open-ended problems that require research, ideation, design, implementation, and presentation of solutions, mirroring professional project cycles in the IT industry.
To ensure effective implementation, PBL involves:

1. **Formulation of Team and Topic Finalization:**
   (a) Students should form a group of 3 to 4 members
   (b) Staff and Students should discuss the relevant problem statement.(Prefer real world problems having some social impact and application)
   (c) Each team should be allocated a guide.
   (d) Students should submit Synopsis(should contain Flowchart, Usage of the logic, algorithm, functions and suitable data structure for implementing the solution)
2. **Development:**
   (a) Select any suitable programming platform (Open source, window, web, mobile applications or any other suitable)
   (b) Prefer open source technologies for development.
   (c) Students can select any programming language they have learnt or in which they are competent.
3. **Design and Documentation:**
   (a) SDLC has to be followed for design and development
   (b) Prepare Analysis Specification Document, Input Specification and Design Specification Documents(use Data Design, DFD, Flowcharts, UML diagrams, Data Dictionary, ER dig etc)
   (c) Follow SDD, SRS
   (d) Provide Test Specifications(test cases, test results, test methodology etc)
   (e) Report Generations if needed.
4. **Report and Presentation:**
   (a) Students should present the working model of the project to the guide and panel of the college.
   (b) They should prepare a report comprising the above mentioned terminologies.
   (c) Submit Hard copy/Soft copy of the report which should contain certificate signed by guide , HOD and principal (prefer soft copy)

Selection of Project/Problem

The problem-based project-oriented model for learning is recommended. The model begins with the identifying of a problem, often growing out of a question or "wondering". This formulated problem then stands as the starting point for learning. Students design and analyze the problem/project within an articulated interdisciplinary or subject frame. A problem can be theoretical, practical, social, technical, symbolic, cultural, and/or scientific and grows out of students' wondering within different disciplines and professional environments. A chosen problem has to be exemplary. The problem may involve an interdisciplinary approach in both the analysis and solving phases. By exemplary, a problem needs to refer back to a particular practical, scientific, social and/or technical domain. The problem should stand as one specific example or manifestation of more general learning outcomes related to knowledge and/or modes of inquiry. There are no commonly shared criteria for what constitutes an acceptable project. Projects vary greatly in the depth of the questions explored, the clarity of the learning goals, the content, and structure of the activity.

• A few hands-on activities that may or may not be multidisciplinary.

• Use of technology in meaningful ways to help them investigate, collaborate, analyse, synthesize, and present their learning.

• Activities may include- Solving real life problem, investigation, /study and Writing reports of in depth study, field work.

Assessment

The institution/head/mentor is committed to assessing and evaluating both student performance and program effectiveness. Progress of PBL is monitored regularly on weekly basis. Weekly review of the work is necessary. During process of monitoring and continuous assessment and evaluation of the individual and the team performance is to be measured. PBL is monitored and continuous assessment is done by supervisor /mentor and authorities. Students must maintain an institutional culture of authentic collaboration, self-motivation, peer learning and personal responsibility. The institution/department should support students in this regard through guidance/orientation programs. Supervisor/mentor and Students must actively participate in assessment and evaluation processes. Group may demonstrate their knowledge and skills by developing a public product and/or report and/or presentation.

1. Individual assessment for each student (Understanding individual capacity, role and involvement in the project).

2. Group assessment (roles defined, distribution of work, intra-team communication and togetherness) . Documentation and presentation Evaluation and Continuous Assessment: It is recommended that all activities should to be recorded regularly, regular assessment of work need to be done and proper documents need to be maintained at college end by both students as well as mentor (PBL work book). Continuous Assessment Sheet (CAS) is to be maintained by all mentors/department and institutes. Recommended parameters for assessment/evaluation and weightage.

3. Idea Inception and Awareness /Consideration of -Environment/ Social /Ethics/ Safety measures/Legal aspects (10%).

4. Outcomes of PBL/ Problem Solving Skills/ Solution provided/ Final product (Individual assessment and team assessment) (40%).

5. Documentation (Gathering requirements, design and modeling, implementation/execution, use of technology and final report, other documents) (15%).

6. Demonstration (Presentation, User Interface, Usability) (20%).

7. Contest Participation/ publication (15%) PBL workbook will serve the purpose and facilitate the job of students, mentor and project coordinator. It will reflect accountability, punctuality, technical writing ability and work flow of the work undertaken.

Note: While planning for the assessment, choose a valid method based on your context. It should be able to understand by both the students as well as the faculty. The student group must follow the principles of Software Engineering (Scoping out the problem, the solution implementation and related documentation). Researching the problem and outlining various approaches is key here and should be emphasized by the tutor and the mentor. Aspects of design thinking (from the point of view of the person facing the problem) are very important. Students should not jump into the technology aspects first. The team can follow the principles of Agile Software Development. The weekly meetings could be used as a Scrum meeting. The tutor and mentor should actively help the students to scope the work and the approach. They must validate the technology choices. If the implementation code is well documented, the project can be continued by subsequent batch, which will help solve a bigger problem.

Student's Role in PBL

Before initiating PBL sessions, students should be oriented and equipped with the mindset and skills required for independent and innovative problem-solving. The objective is to nurture original thinkers and solution creators, not mere replicators of existing ideas.

In the PBL framework, students are active drivers of their own learning journey, taking full responsibility for initiating tasks, conceptualizing ideas, and progressing from problem identification to solution delivery. Key expectations include:

1. **Idea Initiation & Ownership:** Students must proactively propose and refine ideas or problems, avoiding dependency on external direction.

2. **Problem Analysis & Application:** Define, investigate, and critically analyze the challenge, formulate learning objectives, and apply acquired knowledge to design feasible solutions.

3. **Self-Directed Learning:** Manage their own learning process, moving away from passive instruction to active engagement and continuous improvement.

4. **Collaborative Knowledge Construction:** Work in teams to co-create knowledge, share expertise, and enhance collective understanding of the project context.

5. **Interpersonal & Conflict-Resolution Skills:** Develop communication, active listening, constructive feedback, and creative conflict resolution capabilities essential for professional teamwork.

6. **Inquiry & Research Skills:** Cultivate strong inquiry abilities, including framing research questions, sourcing credible data, and applying analytical tools to validate solutions.

Through this active participation model, students gain technical expertise, leadership qualities, and workplace-ready competencies, making them prepared for complex, interdisciplinary, and real-world professional challenges.

Text Books:

1. A new model of problem-based learning. By Terry Barrett. All Ireland Society for higher education (AISHE). ISBN:978-0-9935254-6-9; 2017
2. Problem Based Learning. By Mahnazmoallem, woei hung and Nada Dabbagh, Wiley Publishers. 2019.
3. Stem Project based learning and integrated science, Technology, Engineering and mathematics approach By Robert Capraro, Mary Margaret Capraro

Reference Books:

1. De Graff E, Kolmos A, red: Management of change: Implementation of problem-based and project-based learning in engineering. Rotterdam: Sense Publishers. 2007. 2. Gopalan, “Project management core text book”.
2. Indian Edition James Shore and Shane Warden, “The Art of Agile Development”

MOOC / NPTEL/YouTube Links:

1. https://onlinecourses.nptel.ac.in/noc19_mg30/preview

Savitribai Phule Pune University First Year of MCA (Under Engineering) (2025Pattern)		
OJT-581-MCA : Internship		
Teaching Scheme	**Credits**	**Examination Scheme**
Practical: 06 Hours/Week	03	**Term Work** : 50 Marks **Oral/Presentation** : 50 Marks

Course Objectives: The course aims to:

Internship provides an excellent opportunity to learner to see how the conceptual aspects learned in classes are integrated into the practical world. Industry/on project experience provides much more professional experience as value addition to classroom teaching.

1. To encourage and provide opportunities for students to get professional/personal experience through internships.
2. To learn and understand real life/industrial situations.
3. To get familiar with various tools and technologies used in industries and their applications.
4. To nurture professional and societal ethics.
5. To create awareness of social, economic and administrative considerations in the working environment of industry organizations

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO1: Demonstrate** professional competence through industry internship.
- **CO2: Apply** knowledge gained through internships to complete academic activities in a professional manner.
- **CO3: Choose** appropriate technology and tools to solve given problem.
- **CO4: Demonstrate** abilities of a responsible professional and use ethical practices in day to day life.
- **CO5: Creating** network and social circle, and developing relationships with industry people.
- **CO6: Analyze** various career opportunities and decide carrier goals.

Guidelines

Internships are educational and career development opportunities, providing practical experience in a field or discipline. Internships are far more important as the employers are looking for employees who are properly skilled and having awareness about industry environment, practices and culture. Internship is structured, short-term, supervised training often focused around particular tasks or projects with defined time scales.

Core objective is to expose technical students to the industrial environment, which cannot be simulated/experienced in the classroom and hence creating competent professionals in the industry and

to understand the social, economic and administrative considerations that influence the working environment of industrial organizations.

The following guidelines are proposed to give academic credit for the internship undergone as a part of the First year MCA curriculum.

Duration

Internship is to be completed after semester 1 and before commencement of semester 2 and it is to be assessed and evaluated in semester 2.

Internship work Identification

Student may choose to undergo Internship at Industry/Govt. Organizations/NGO/MSME/Rural Internship/ Innovation/IPR/Entrepreneurship. Student may choose either to work on innovation or entrepreneurial activities resulting in start-up or undergo internship with industry/NGO's/Government organizations/Micro/Small/ Medium enterprises to make themselves ready for the industry. Student can take internship work in the form of the following but not limited to:•

- Working for consultancy/ research project,
- Contribution in Incubation/ Innovation/ Entrepreneurship Cell/ Institutional Innovation Council/ startups cells of institute /
- Learning at Departmental Lab/Tinkering Lab/ Institutional workshop,
- Development of new product/ Business Plan/ registration of start-up,
- Industry / Government Organization Internship,
- Internship through Internshala,
- In-house product development, intercollegiate, inter department research internship under research lab/group, micro/small/medium enterprise/online internship,
- Research internship under professors, IISC, IIT's, Research organizations,
- NGOs or Social Internships, rural internship,
- Participate in open source development.

Internship Diary/ Internship Workbook

Students must maintain Internship Diary/ Internship Workbook. The main purpose of maintaining diary/workbook is to cultivate the habit of documenting. The students should record in the daily training diary the day-to-day account of the observations, impressions, information gathered and suggestions given, if any. The training diary/workbook should be signed every day by the supervisor.

Internship Diary/workbook and Internship Report should be submitted by the students along with attendance record and an evaluation sheet duly signed and stamped by the industry to the Institute immediately after the completion of the training.

Internship Work Evaluation

Every student is required to prepare a maintain documentary proofs of the activities done by him as internship diary or as workbook. The evaluation of these activities will be done by Programme Head/Cell In-charge/ Project Head/ faculty mentor or Industry Supervisor based on- Overall compilation of internship activities, sub-activities, the level of achievement expected, evidence needed to assign the points and the duration for certain activities.

Evaluation through Seminar Presentation/Viva-Voce at the Institute

The student will give a seminar based on his training report, before an expert committee constituted by the concerned department as per norms of the institute. The evaluation will be based on the following criteria:

- Depth of knowledge and skills
- Communication & Presentation Skills
- Team Work
- Creativity
- Planning & Organizational skills
- Adaptability
- Analytical Skills
- Attitude & Behavior at work
- Societal Understanding
- Ethics
- Regularity and punctuality
- Attendance record
- Diary/Work book
- Student's Feedback from External Internship Supervisor

After completion of Internship, the student should prepare a comprehensive report to indicate what he has observed and learnt in the training period.

Internship Diary/workbook may be evaluated on the basis of the following criteria:

- Proper and timely documented entries
- Adequacy & quality of information recorded
- Data recorded
- Thought process and recording techniques used
- Organization of the information

The report shall be presented covering following recommended fields but limited to :

- Title/Cover Page
- Internship completion certificate
- Internship Place Details- Company background-organization and activities/Scope and object of the study / Supervisor details
- Index/Table of Contents
- Introduction
- Title/Problem statement/objectives
- Motivation/Scope and rationale of the study
- Methodological details
- Results / Analysis /inferences and conclusion
- Suggestions / Recommendations for improvement to industry, if any
- Attendance Record
- Acknowledgement
- List of reference (Library books, magazines and other sources)

Task Force for Curriculum Design and Development

Programme Coordinator

Dr. Suhasini Itkar

Team Members for Course Design

Dr. Amit Bhusari	Trinity Academy of Engineering, Pune
Dr. Atul Nevase	Anantrao Pawar College of Engineering and Research, Pune
Prof. Mukta Deshpande	G.S. Moze College of Engineering, Balewadi
Dr. Priyanka Abhang	Sandip Institute of Engineering and Management, Nashik
Prof. Pankaj Sathe	SND College of Engineering & Research Center , Yeola
Prof. Prashant Londhe	PDEA's College of Engineering , Manjari
Prof. P. D. Jadhav	MET's Institute of Engineering, Nashik
Dr. Poonam Choudhary	GES, R.H. Sapat College of Engg, MS & R, Nashik
Prof. Manjusha Khond	MET's Institute of Engineering, Nashik
Dr. Kapil Misal	Trinity Academy of Engineering, Pune
Dr. Rucha Samant	GES, R.H. Sapat College of Engg, MS & R, Nashik
Prof. Javed Attar	MET's Institute of Engineering, Nashik
Prof. Gayatri Raut	GES, R.H. Sapat College of Engg, MS & R, Nashik
Prof. Pramod Jadhao	Trinity Academy of Engineering, Pune
Prof. Swati Andhale	Parvatibai Genba Moze College Of Engineering , Wagholi
Prof. Bisweswar Thakur	Trinity Academy of Engineering, Pune
Prof. Sujata Patil	Trinity Academy of Engineering, Pune
Prof. Priyanka T. yeole	G.S.Moze College of Engineering , Balewadi
Prof. Vibha Upadhya	Trinity Academy of Engineering, Pune
Prof. Dhanashree R. Kolpe	G.S.Moze College of Engineering, Balewadi
Prof. Vaishali Hatkar	Trinity Academy of Engineering, Pune
Prof. Supriya S. Gaikwad	Sandip Institute of Engineering and Management, Nashik
Prof. Ganesh S. Ragade	Sandip Institute of Engineering and Management, Nashik
Prof. Meharaj I. Khan	MET's Institute of Engineering, Nashik
Prof. Munmun P. Puranik	G.S. Moze College of Engineering, Balewadi
Prof. Shubhangi Vitalkar	Trinity Academy of Engineering, Pune
Prof. Umesh Jadhav	S B Patil College of Engineering, Indapur
Dr. Nita Shinde	MET's Institute of Engineering, Nashik
Prof. Dipali A. Bhusari	Trinity Academy of Engineering, Pune
Prof. Sachin V. Shinde	PDEA's College of Engineering , Manjari
Prof. Sonali Vidhate	MET's Institute of Engineering, Nashik
Prof. Sonali Mutha	Anantrao Pawar College of Engineering and Research, Pune
Prof. Neha Bhatambarekar	GES, R.H. Sapat College of Engg, MS & R, Nashik

Chairman

Dr. Nilesh Uke - Board of Studies Computer Engineering

Savitribai Phule Pune University, Pune

Dean

Dr. Pramod Patil - Dean – Science and Technology

Savitribai Phule Pune University, Pune