Faculty of Science and Technology

Savitribai Phule Pune University

Curriculum for
First Year Master of Computer Applications(MCA)

(Course 2019)

(with effect from June 2019)

Savitribai Phule Pune University
Master of Computer Applications

Program Educational Objectives

PEO 1: To prepare globally competent graduates having knowledge of computer applications and emerging technologies to provide effective solutions for computing problems.

PEO 2: To prepare committed and motivated graduates by inculcating professional ethics and values with knowledge of legal and environmental issues.

PEO 3: To prepare graduates with research attitude, analytical skills, lifelong learning ability and multidisciplinary thinking.

PEO 4: To prepare graduates with managerial and soft skills to work effectively as an individual and in teams.

Program Outcomes

Students are expected to know and be able to-

PO1. Apply knowledge of mathematics, computer science, computing specializations appropriate for real world applications.

PO2. Identify, formulate, analyze and solve *complex* computing problems using relevant domain disciplines.

PO3. Design and evaluate solutions for *complex* computing problems that meet specified needs with appropriate considerations for real world problems.

PO4. Find solutions of complex computing problems using design of experiments, analysis and interpretation of data.

PO5. Apply appropriate techniques and modern computing tools for development of complex computing activities.

PO6. Apply professional ethics, cyber regulations and norms of professional computing practices.

PO7. Recognize the need to have ability to engage in independent and life-long learning in the broadest context of technological change.

PO8. Demonstrate knowledge and understanding of the computing and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO9. Communicate effectively with the computing community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO10. Assess societal, environmental, health, safety, legal and cultural issues within local and global contexts, and the consequent responsibilities relevant to the professional computing practices.

PO11. Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary environments.

PO12. Identify a timely opportunity and use innovation, to pursue opportunity, as a successful entrepreneur/professional.

Course Structure for Semester_1

Course Code	Course	Teaching Scheme Hours/Week		Examination Scheme						Credit	
		TH	PR	In-Sem	End _Sem	TW	OR	PR	Total Marks	TH	PR
310901	Discrete Mathematics	3	-	30	70	-	-	-	100	3	
310902	Data Structures	4	-	30	70	-	-	-	100	4	
310903	Object Oriented Programming	3	-	30	70	-	-	-	100	3	
310904	Principles of Programming	3	-	30	70	-	-	-	100	3	
310905	Management Theory and Practices	3	-	30	70	-	-	-	100	3	
310906	Data Structures Laboratory	-	4	-	-	50	-	50	100		2
310907	Object Oriented Programming Laboratory	-	4	-	-	50	-	50	100		2
310908	Programming Language Laboratory	-	2	-	-	50		50	100		1
	Total	16	10	150	350	150	-	150	800	21	
310909	**Audit Course -1:** AC1-I Humanities and Social Sciences, AC1-II Road Safety									Grade	

Course Structure for Semester_2

Course Code	Course	Teaching Scheme Hours/Week		Examination Scheme						Credit	
		TH	PR	In-Sem	End _Sem	TW	OR	PR	Total Marks	TH	PR
310910	Probability & Statistics	3	-	30	70	-	-	-	100	3	
310911	Systems Programming &Operating System (SPOS)	4	-	30	70	-	-	-	100	4	
310912	Database Management System	3	-	30	70	-	-	-	100	3	
310913	Java Programming	3	-	30	70	-	-	-	100	3	
310914	Computer Organization	3		30	70	-	-	-	100	3	
310915	Database Management System Laboratory	-	4	-	-	50	-	50	100		2
310916	SPOS Laboratory	-	2	-	-	50	-	-	50		1
310917	Java Programming Laboratory	-	4	-	-	50	-	50	100		2
310918	Project Based Learning	-	2	-	-	50	-	-	50		1
	Total	16	12	150	350	200	--	100	800	22	
310919	**Audit Course -2:** AC2-I Foreign Language, AC2-II Environmental Studies									Grade	

SEMESTER I

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310901: Discrete Mathematics

Teaching Scheme:	Credit	Examination Scheme:
TH: 03 Hours/Week	03	In_Sem: 30 Marks End_Sem : 70 Marks

Prerequisites: Basic Mathematics

Course Objectives:

- To study discrete objects and relationships among them
- To use appropriate set, function and relation models to understand practical examples, and interpret the associated operations and terminologies in context
- To learn logic and proof techniques to expand mathematical maturity
- To determine number of logical possibilities of events
- Determine number of logical possibilities of events
- To formulate problems precisely, solve the problems, apply formal proof techniques, and explain the reasoning clearly
- To demonstrate how these concepts can be applied to solve nontrivial real life problems

Course Outcomes:

On completion of the course, student will be able to–

- Solve real world problems logically by using set and induction approaches.
- Describe and implement relations and functions.
- Apply logical reasoning to solve a variety of problems.
- Use the basic properties of graphs and trees to model simple applications.
- Analyze and synthesize the real world problems using Algebraic structures.

Course Contents

Unit I	Set Theory and Logic	08 Hours

Discrete Mathematics, Significance of Discrete Mathematics in Computer Engineering, Sets–Naïve Set Theory (Cantorian Set Theory), Need for Sets, Representation of Sets, Set Operations, cardinality of set, principle of inclusion and exclusion, Types of Sets –Countable and Uncountable Sets, Finite and Infinite Sets, Countably Infinite and Uncountably Infinite Sets, power set. Propositional Logic-logic, Propositional Equivalences, Application of Propositional Logic-Translating English Sentences, Proof by Mathematical Induction.

Unit II	Relations and Functions	07 Hours

Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations , Closures of Relations, Equivalence Relations, Partial Orderings, partitions, Hasse Diagram, Lattices, Transitive Closure and Warshall‘s Algorithm, n-Ary Relations and their Applications. Functions-Surjective, Injective and Bijective functions, Inverse Functions, The Pigeonhole Principle.

Unit III	Permutations and Combinations	07 Hours

The Basics of Counting, rule of Sum and Product, Permutations and Combinations, Generalized Permutations and Combinations.

Unit IV	Graph Theory	07 Hours

Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Planar Graphs, Graph Colouring. Case Study-Web Graph, Google map.

Unit V Trees 07Hours

Introduction, properties of trees, Binary search tree, decision tree, prefix codes and Huffman coding, cut sets, Spanning Trees and Minimum Spanning Tree, Kruskal's and Prim's algorithms, Case Study-Game Tree, Mini-Max Tree

Unit VI Algebraic Structures and Coding Theory 06 Hours

The structure of algebra, Algebraic Systems, Semi Groups, Monoids, Groups, Homomorphism and Normal Subgroups, and congruence relations, Rings, Integral Domains and Fields, Polynomial Rings and polynomial Codes.

Books:

Text:

1. Kenneth H. Rosen, "Discrete Mathematics and its Applications", Tata McGraw-Hill, ISBN 978-0-07-288008-3, 7th Edition.
2. C. L. Liu, "Elements of Discrete Mathematics", TMH, ISBN 10:0-07-066913-9.

Reference:

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. Narsingh Deo, "Graph with application to Engineering and Computer Science", Prentice Hall of India, 1990, 0 –87692 –145 –4.
4. Dr. K. D. Joshi, "Foundations of Discrete Mathematics", New Age International Limited, Publishers, January 1996, ISBN: 8122408265, 9788122408263
5. Sriram P & Steven S, "Computational Discrete Mathematics", Cambridge University Press, ISBN 13: 978-0-521-73311-3
6. J. Tremblay, Manohar R., "Discrete Mathematical Structures with application to Computer Science", Tata McGraw-Hill, 2002 ISBN 0-07-463113-6

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310902: Data Structures

Teaching Scheme:	Credit	Examination Scheme:
TH: 04 Hours/Week	04	In_Sem: 30 Marks End_Sem : 70 Marks

Prerequisites: Basics of programming language

Course Objectives:

- To understand the standard and abstract data representation methods.
- To understand the memory requirement for various data structures.
- To operate on the various structured data.
- To understand various data searching and sorting methods with pros and cons

Course Outcomes:

On completion of the course, student will be able to–

- To learn how data structure concepts are useful in problem solving.
- To implement different ways of data structures such as stacks, linked lists, trees & graphs.
- To analyse and design notation of algorithm.

Course Contents

Unit I	Introduction to Data Structures and Array Concepts	08 Hours

Introduction to data structures: Concept of data, Data types, Data Object, Data structure, Notation of Data Structure, Abstract Data types (ADT), Linear data structures using sequential organization: Concept of sequential organization, Concept of Linear data structures, arrays as ADT, Multidimensional arrays, Storage representations (row major and column major and their address calculation). Polynomial representation using arrays, Application of array in sparse matrix representation, addition and transpose

Unit II	Linked Lists	08 Hours

Concept, Comparison of sequential and linked organizations, Primitive operations, Realization of Linked Lists, Realization of linked list using arrays, Dynamic Memory Management, Linked list using dynamic memory management, Linked List Abstract Data Type, Linked list operations, Head pointer and header node, **Types of linked list-** Singly, Doubly Linked List and operations, Circular Linked List, Singly circular linked list, Doubly circular linked list**, Polynomial Manipulations -** Polynomial addition, Multiplication of two polynomials using linked list

Unit III	Stacks	08 Hours

Stacks- concept, Primitive operations, Stack Abstract Data Type, Representation of Stacks Using Sequential Organization, stack operations, Multiple Stacks, Applications of Stack- Expression Evaluation and Conversion, Polish notation and expression conversion, Need for prefix and postfix expressions, Postfix expression evaluation, Linked Stack and Operations**.**

Recursion- concept, **variants of recursion-** direct, indirect, tail and tree, Backtracking algorithmic strategy, use of stack in backtracking.

Unit IV	Queues	08 Hours

Concept, Queue as Abstract Data Type, Realization of Queues Using Arrays , Circular Queue, Advantages of using circular queues, Multi-queues, Deque, Priority Queue, Array implementation of priority queue, Linked Queue and operations.

<table>
<tr><td><b>Unit V</b></td><td><b>Non Linear Data Structures</b></td><td><b>08 Hours</b></td></tr>
<tr><td colspan="3">Tree : Trees and binary trees-concept and terminology, Sequential & Linked representation of binary trees, Algorithm for tree traversals, Conversion of general tree to binary tree, Binary search trees, Applications of binary tree : expression tree, decision tree.
Graph: Representation of graph -Adjacency matrix and Adjacency list, Graph traversals, application of graph: connected components , Spanning tree, Minimum cost spanning tree, shortest path computation</td></tr>
<tr><td><b>Unit VI</b></td><td><b>Searching and Sorting</b></td><td><b>08 Hours</b></td></tr>
<tr><td colspan="3"><b>Searching-</b> Search Techniques, Sequential search, variant of sequential search- sentinel search, Binary search, Fibonacci search.
<b>Sorting-</b> 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, Heap sort, Shell sort, Bucket sort, Radix sort, Comparison of All Sorting Methods.</td></tr>
<tr><td colspan="3"><b>Books:</b></td></tr>
<tr><td colspan="3"><b>Text Books:</b>
<b>1.</b> Aho A., Hopcroft J., Ulman J., “Data Structures and Algorithms”, Pearson Education, ISBN-0-201-43578-0
<b>2.</b> Brassard & Bratley, “Fundamentals of Algorithmics”, Prentice Hall India/Pearson Education, ISBN 13-9788120311312.
<b>3.</b> Richard F. Gilberg, Behrouz A. Forouzan, “Data Structures: A Pseudocode Approach with C”, Cengage Learning, ISBN: 9788131503140.</td></tr>
<tr><td colspan="3"><b>Reference Books:</b>
<b>1.</b> Horowitz and Sahani, “Fundamentals of Data Structures in C++”, University Press, ISBN 10: 0716782928 ISBN 13: 9780716782926.
<b>2.</b> Yedidyah Langsam, Moshe J Augenstein, Aron M Tenenbaum, “Data Structures using C and C++”, Pearson Education, ISBN 81-317-0328-2.
<b>3.</b> A Michael Berman, “Data Structures via C++: Objects by Evolution”, Oxford University Press, ISBN:0-19-510843-4.
<b>4.</b> Trembley, J.P. and Sorenson P.G., “An Introduction to Data Structures with Applications”, McGrawHill
<b>5.</b> Donald Knuth “Art of Programming”, Vol. 1, 2, 3, Pearson Education, ISBN 10: 0830604553 / ISBN 13: 9780830604555.</td></tr>
</table>

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310903: Object Oriented Programming

		Examination Scheme:
Teaching Scheme:	**Credit**	**In_Sem: 30 Marks**
TH: 03 Hours/Week	**03**	**End_Sem : 70 Marks**

Prerequisites: Basics of programming languages

Course Objectives:

- To study basics of Object Oriented Programming (OOP).
- To understand object-oriented concepts such as data abstraction, encapsulation, inheritance, dynamic binding, and polymorphism.
- To use the object-oriented paradigm in program design.
- Provide programming insight using OOP constructs.

Course Outcomes:

On completion of the course, student will be able to–

- Analyse the strengths of object oriented programming
- Design and apply OOP principles for effective programming
- Develop programming application using object oriented programming language C++
- Achieve applicability of OOP

Course Contents

Unit I	Basics of Programming	07 Hours

Basics of C++ - Need of Object-Oriented Programming (OOP), Object Oriented Programming Paradigm, Basic Concepts of Object-Oriented Programming and advantages of it, Benefits of OOP, C++ as object oriented programming language, Syntax & Structure of C++ Programming Comments, header files.

Unit II	Classes, Objects and Functions	07 Hours

Classes and Objects -Data Types and Variables, Operators, Control-flow Statements, Looping Statements, Arrays, Strings, Structures, Enumerations, Class, Object, class and data abstraction, class scope and accessing class members, separating interface from implementation, controlling access to members

Functions- Function, function prototype, accessing function and utility function, Constructors and destructors, Copy Constructor, Objects and Memory requirements, Static Class members, data abstraction and information hiding, inline function.

Unit III	Inheritance and Polymorphism	07 Hours

Operator Overloading- concept of overloading, operator overloading, Overloading Unary Operators, Overloading Binary Operators, Data Conversion, Type casting (implicit and explicit), Pitfalls of Operator Overloading and Conversion.

Inheritance- Base Class and derived Class, protected members, relationship between base Class and derived Class, Constructor and destructor in Derived Class, Overriding Member Functions, Class Hierarchies, Inheritance, Public and Private Inheritance, Levels of Inheritance, Multiple Inheritance.

Polymorphism- concept, relationship among objects in inheritance hierarchy, abstract classes,

<table>
<tr><td colspan="3">polymorphism</td></tr>
<tr><td><b>Unit IV</b></td><td><b>Virtual Functions</b></td><td><b>07 Hours</b></td></tr>
<tr><td colspan="3"><b>Virtual Function-</b> Need for virtual function, Friend Functions, Static Functions, Assignment and Copy Initialization, this Pointer, virtual function, dynamic binding, Virtual destructor, this Pointer</td></tr>
<tr><td><b>Unit V</b></td><td><b>Templates and Exception Handling</b></td><td><b>07 Hours</b></td></tr>
<tr><td colspan="3"><b>Templates-</b> function templates, Function overloading, overloading Function templates, class templates, class template, template and inheritance, template and friends Generic Functions, Applying Generic Function, Generic Classes, The type name and export keywords, The Power of Templates.
<b>Exception Handling-</b> Fundamentals, C++ Standard Exceptions</td></tr>
<tr><td><b>Unit VI</b></td><td><b>Files handling</b></td><td><b>07 Hours</b></td></tr>
<tr><td colspan="3">Ifstream, of stream, istream, ostream and fstream classes and their hierarchy. Input and output operation - open() ,get(), getline(), read(), seekg() and tellg() AND put(), seekp(), tellp(),and write() functions, Command-Line Arguments, Printer output, Early vs. Late Binding, Error Handling in File I/O</td></tr>
<tr><td colspan="3"><b>Books:</b></td></tr>
<tr><td colspan="3"><b>Text :</b>
<b>1.</b> Bjarne Stroustrup, “The C++ Programming language”, Third edition, Pearson Education. ISBN 9780201889543.
<b>2.</b> Deitel, “C++ How to Program”, 4thEdition, Pearson Education, ISBN:81-297-0276-2</td></tr>
<tr><td colspan="3"><b>Reference :</b>
<b>1.</b> Robert Lafore, “Object-Oriented Programming in C++”, fourth edition, Sams Publishing, ISBN:0672323087 (ISBN 13: 9780672323089)
<b>2.</b> E. Balgurusamy, “Object oriented programming in C++ “, Tata McGraw Hill, ISBN: 9780071072830,
<b>3.</b> Herbert Schildt, “C++ The complete reference”, Eighth Edition, McGraw Hill Professional, ISBN:978-00-72226805
<b>4.</b> Matt Weisfeld, “The Object-Oriented Thought Process”, Third Edition, Pearson ISBN-13:075-2063330166
<b>5.</b> Cox Brad, Andrew J. Novobilski, “Object Oriented Programming: An Evolutionary Approach”, Second Edition, Addison–Wesley, ISBN: 13:978-020-1548341.</td></tr>
</table>

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310904: Principles of Programming

Teaching Scheme:	Credit	Examination Scheme:
TH: 03 Hours/Week	03	In_Sem: 30 Marks End_Sem : 70 Marks

Prerequisites:

Course Objectives:

- To develop analytical and logical thinking and problem solving capabilities.
- To develop understanding of how a programming problem is recognized and how a solution to the problem can be designed
- To develop understanding of how programs can be tested.
- To develop understanding of how programs can be documented.

Course Outcomes:

On completion of the course, student will be able to–

- Define fundamental concept of Software development, syntax and semantics of computer languages.
- Describe structuring of data and computations.
- Define programming structure with different aspects.
- Explain general principles of computer programming such as simple loops , decision structures and functions
- Design simple and complex algorithms and determine their time complexity
- Explain the importance of Searching and sorting with arrays.

Course Contents

Unit I	Introduction to Programming Language and Semantics	07 Hours

Introduction

Software development process, language and software development environments, language and software design methods,

Syntax and semantics-language definition, syntax, abstract syntax, concrete syntax, and pragmatics, semantics, an introduction to formal semantics, languages, language processing, interpretation, translation, the concept of binding, variables, name and scope, Type, l-value, r-value, reference and unnamed variables, routines, generic routines, aliasing and overloading, an abstract semantic processor, run time structure

Unit II	Structuring of Data and Computations	07 Hours

Structuring of Data- Built in and primitive types, Data aggregates and type constructors, Cartesian product, Finite mapping User-defined types and abstract data types, Type systems, Static versus dynamic program checking, Strong typing and type checking, Type compatibility, Type conversions, Types and subtypes, Generic types, monomorphic versus polymorphic type systems.

Structuring of Computations- Structuring the computation, Expressions and statements, Conditional execution and iteration, Routines, Style issues: side effects and aliasing, Exceptions.

<table>
<tr><th>Unit III</th><th>Programming Structure</th><th>07 Hours</th></tr>
<tr><td colspan="3">Introduction to programming structure, modular programming, top down/bottom up design approach, modules and their function cohesion and coupling, logical and global variable, parameters, return values, variable names and data dictionary , naming conventions, Implementation of Sequential, selection and iterative structures , Recursive approach</td></tr>
<tr><th>Unit IV</th><th>Flow Chart and Algorithms</th><th>07 Hours</th></tr>
<tr><td colspan="3">Flow Charts
Notations of Flow Charts and its implementation, Top-Down-step wise refinement, Implementation of algorithms, Program Verification, Efficiency of algorithms, Analysis of Algorithm,
Basic Algorithms: Exchange of values of two variables with or without third variable, Summation of set of numbers, Factorial Computation, Sine function computation, Generation of Fibonacci Sequence, Reversing of digits of an integer, Base conversions, Character to Number conversion, Finding Square Root, smallest divisor, Factorial, GCD, Generating Prime numbers, prime factors of integers, pseudo random number generation, Pascal triangle, Perfect number.</td></tr>
<tr><th>Unit V</th><th>Analysis of Algorithms</th><th>07 Hours</th></tr>
<tr><td colspan="3"><b>Analysis of Algorithms:</b> Introduction of analysis of an algorithm, frequency count and its importance of analysis of an algorithm, Complexity of algorithm, asymptotic notations - big O, Φ, Ω notations, how to estimate running time of an algorithm- counting number of iterations, counting the frequency of basic operations, using recurrence relation, Best, worst and average case analysis using some of the above examples</td></tr>
<tr><th>Unit VI</th><th>Algorithms using Arrays</th><th>07 Hours</th></tr>
<tr><td colspan="3"><b>Algorithm using Array:</b> Processing Array- one dimensional, multidimensional arrays, table lookup technique, pointer technique, Array technique- Maximum and minimum of array, reversing of an array, mean and median of n-numbers, Row major and column major form of array representation,
<b>Sorting and searching:</b> Linear search, binary search, Insertion sort, bubble sort, selection sort.
<b>Business data processing:</b> What is data processing? Standard methods of organizing data, File management system, Database management system</td></tr>
<tr><th colspan="3">Books:</th></tr>
<tr><td colspan="3"><b>Text :</b>
<b>1.</b> Carlo Ghezzi, Mehdi Jazayeri, “Programming Language Concepts”,3rd Edition, Wiley Publication ISBN : 978-81-265-1861-6.
<b>2.</b> Sebesta R., "Concepts of Programming Languages", 4th Edition, Pearson Education, ISBN- 81-7808-161-X. 2.</td></tr>
<tr><td colspan="3"><b>Reference:</b>
<b>1.</b> T. W. Pratt, M. V. Zelkowitz, "Programming Languages Design and Implementation”, 4th Edition, PHI, ISBN 81-203-2035-2.
<b>2.</b> Maureen Sprankle & Jim Hubbard, “Problem Solving and Programming Concepts”, ISBN-13: 978-0132492645.
<b>3.</b> R. G. Dromey, “ How to solve it by Computer”, Prentice-Hall, ISBN:10: 8131705625, 13: 978-8131705629</td></tr>
</table>

Savitribai Phule Pune University, Pune
First year of MCA (2019 Course)
310905: Management Theory and Practices

Teaching Scheme:	Credit	Examination Scheme:
TH: 03 Hours/Week	03	In_Sem: 30 Marks End_Sem : 70 Marks

Course Objectives:

- To understand Management in the Organization.
- Students should be able to understand significance of Team Building and Conflicts in the Organization.
- Students should be able get knowledge of Management Information Systems, Customer Relationship Management and Supply Chain management using real life examples.
- To understand the Managerial Decision Techniques and Managerial Ethics.

Course Outcomes:

On completion of the course, student will be able to–

- Describe the Functions and activities of the Management.
- Explain the Organizational structure.
- Describe the Leadership, Team Building Models and Managerial Ethics.
- Elaborate the Conflict Management, Total Quality Management and Re-engineering Process.
- Explain Management Information System, Customer Relationship Management and Supply Chain Management and ERP.
- Elaborate Managerial Decision Making Models

Course Contents

Unit I	Introduction to Management	07 Hours

Definition of Management: its nature and purpose, Management as a Science or an Art? The need, scope, purpose of Management, The Systems approach to Operational Management, Functions, Skills of Managers, Functions and activities of Management, planning, organizing, staffing, directing and controlling

Unit II	Organization	07 Hours

Introduction -Definition, Need for Organization, Process of Organizing, Organizational Structure Functional organization, Product Organization, MOA and AOA, Organizational Structures, Definition, types, merits and demerits of each of structures (Line, Functional, Line and staff, Committee, Matrix and Project structure)

Unit III	Organizational Behaviour	07 Hours

Definition / Concepts, Need, Group and Group Dynamics, Team Building, Leadership: -Definition - its importance to the organization - leadership style approaches to the study of Leadership - trait, Behavioural and Situational approaches - Fiedler's contingency model, Hersey and Blanchard's Theory, Black and Moutan's Theory, Path and Goal Theory, Managerial Ethics.

Unit IV	Conflict Management & TQM	07 Hours

Conflict Management, Motivation: Concept Theory X, Y and Z, Total Quality Management: Techniques of TQM, Re-engineering

Unit V Management Information System 07 Hours

Definitions, Role of MIS, Applications of MIS in Academics, Structure of MIS based on Management , Activity and functions, Information Systems concepts to MIS, CRM: Customer Relationship Management, SCM: Supply Chain Management, ERP: Enterprise Resource Planning.

Unit VI Managerial Decision Making 07 Hours

Introduction, Decision making environment: Open Systems, Closed system, Decision Making under certainty, Decision making under uncertainty, Decision making under Risk, Decision Types/Models: Structured decisions, unstructured decisions, Programmable decisions, Non programmable Decisions Classical Model, Administrative Model, Decision making tools: Autocratic, Participative, and Consultative Decision Making Tools, Herbert Simpson's Model, Principle of Rationality / Bounded Rationality

Books:

Text :

1. Stephen P Robbins, Timothy A Judge, NeharikaVohra, "Organizational Behavior", Pearson Education Limited, ISBN-13 9789332542228.
2. Jawadekar W, "Management Information Systems", Tata McGraw-HillPublishing Company Limited, *ISBN*-13: 978-0070445758.

Reference :

1. GordanDevis, Margrethe H. Oison, "Management Information System",Tata McGraw-Hill,*ISBN* 0-07-044555-96.
2. Shukla M., "Business Organization and management", S.Chand (G/L) & Company Ltd., ISBN 81-219-08132.
3. Koontz H &Weitrich, "Essential of management", McGraw Hill Education; 9th Edition ISBN-13: 978-1259005121.
4. Burton & Thakur, "Management Today Principles and Practices", Tata McGraw-Hill Publishing Company Limited, ISBN-0-07-462094-0.
5. Ivancevich&Gibson, Donnelly "Management Principles and Functions" A.I.T.B.S. Publishers, 2005, 4th Edition, ISBN 025606671X, 9780256066715.
6. Keith Davis & John W. Newstrom, "Human Behavior at Work: Organizational Behavior", McGraw-Hill, ISBN 13: 9780070155749.
7. Fred Luthans "Organizational Behavior", TMH, 2013, ISBN-13:978-0073530352 12th Edition, TMH.
8. Robert Murdick, Joel e. Ross, "Information Systems for Modern Management"
PHI, 3rd Edition, ISBN: 9788120303973.
9. Efraim Turban, "Decision Support & Intelligent System", Pearson, 7th Edition, ISBN 9780130461063
10. Poonam Sharma, Kanika T Bhal, "Managerial Ethics", SAGE Publications, 2004, ISBN 0761932496, 9780761932499.
11. Daniel E. O'Leary, "Enterprise Resource Planning Systems: Systems, Life Cycle, Electronic Commerce, and Risk", Cambridge University Press, 1 edition, ISBN-13: 978-0521791526

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310906: Data Structures Laboratory

Teaching Scheme:	Credit	Examination Scheme:
PR: 04 Hours/Week	02	TW: 50 Marks PR : 50 Marks

Guidelines for Instructor's Manual

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

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of 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, algorithm, flowchart, Design, test cases, conclusion/analysis. **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 journal may be avoided. Use of DVD 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 Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of student. 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 Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student's for advanced learning, understanding of the fundamentals, effective and efficient implementation.

So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

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 need to 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 among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned.

Suggested List of Laboratory Assignments

1.	A. Implement application of array in sparse matrix to perform simple and fast transpose B. Implement application of array in sparse matrix to perform matrix manipulation. C. Implement application of array in polynomial expression.
2.	A. Write a menu driven program to perform following operations on singly linked list: Create, Insert, Delete, and Display B. Write a menu driven program to perform following operations on singly linked list: Create, reverse, search, count and Display
3.	Create two doubly linked lists. Sort them after creation using pointer manipulation. Merge these two lists into one list so that the merged list is in sorted order. (No new Node should be created.) B. Write a menu driven program to perform operations on doubly linked list.
4.	A. Implement circular linked list and perform operations on it. *B.* Represent polynomial as a circularly linked list and write a menu driven program to perform addition and evaluation.
5.	A. Implement stack as an ADT. Use this ADT to perform expression conversion and evaluation. (Infix – Postfix) B. Implement stack as an ADT. Use this ADT to perform expression conversion (Infix – Prefix).
6.	A. Implement circular queue using arrays. B. Implement job scheduling algorithm using queue.
7.	A. Write a program to implement Merge sort method. B. Write a program to implement Heap sort method. C. Implement Fibonacci Search.
8.	A. Create binary tree and perform recursive traversals. B. Create binary tree. Find height of the tree and print leaf nodes. Find mirror image, print original and mirror image using level-wise printing.
9.	A. Represent graph using adjacency list/adjacency matrix and perform Depth First Search. B. Represent graph using adjacency list/adjacency matrix and perform Breadth First Search.
10.	A. Implement minimum cost spanning tree algorithm. B. Implement shortest path algorithm.

Savitribai Phule Pune University, Pune
First year of MCA (2019 Course)
310907: Object Oriented Programming Laboratory

Teaching Scheme: PR: 04 Hours/Week	Credit 02	Examination Scheme: TW: 50 Marks PR: 50 Marks

Guidelines for Instructor's Manual

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

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of 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, algorithm, flowchart, Design, test cases, conclusion/analysis. **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 journal may be avoided. Use of DVD 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 Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of student. 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 Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student’s for advanced learning, understanding of the fundamentals, effective and efficient implementation.

So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

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 need to 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 among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned.

Suggested List of Laboratory Assignments

1.	Implement a class Complex which represents the Complex Number data type. Implement the following operations: 1. Constructor (including a default constructor which creates the complex number0+0i). 2. Overloaded **operator+** to add two complex numbers. 3. Overloaded **operator*** to multiply two complex numbers. 4. Overloaded << and >> to print and read Complex Numbers.
2.	Implement a class Quadratic that represents degree two polynomials i.e., polynomials of type ax^2+bx+c. The class will require three data members corresponding to a, b and c. Implement the following operations: 1. A constructor (including a default constructor which creates the 0polynomial). 2. Overloaded **operator+** to add two polynomials of degree2. 3. Overloaded << and >> to print and read polynomials. To do this, you will need to decide what you want your input and output format to looklike. 4. A function eval that computes the value of a polynomial for a given value ofx. 5. A function that computes the two solutions of the equation$ax^2+bx+c=0$.
3.	Implement a class CppArray which is identical to a one-dimensional C++ array (i.e., the index set is a set of consecutive integers starting at 0) except for the following : 1. It performs range checking. 2. It allows one to be assigned to another array through the use of the assignment operator (e.g. cp1=cp2) 3. It supports a function that returns the size of thearray. 4. It allows the reading or printing of array through the use of **cout**and**cin**.
4.	Write a C++ program create a calculator for an arithmetic operator (+, -, *, /). The program should take two operands from user and performs the operation on those two operands depending upon the operator entered by user. Use a switch statement to select the operation. Finally, display the result. Some sample interaction with the program might look like this: *Enter first number, operator, second number:* *10 / 3 Answer =3.333333* *Do another (y/n)?y* *Enter first number, operator, second number: 12* *+ 100 Answer = 112* *Do another (y/n)? n*
5.	Develop an object oriented program in C++ to create a database of student information system containing the following information: Name, Roll number, Class, division, Date of Birth, Blood group, Contact address, telephone number, driving license no. etc

	Construct the database with suitable member functions for initializing and destroying the data viz constructor, default constructor, Copy constructor, destructor, static member functions, friend class, this pointer, inline code and dynamic memory allocation operators-new and delete.
6.	Create a class template to represent a generic vector. Include following member functions: □ To create the vector. □ To modify the value of a given element □ To multiply by a scalar value □ To display the vector in the form (10,20,30,…)
7.	Create a class Rational Number (fractions) with the following capabilities: a) Create a constructor that prevents a 0 denominator in a fraction, reduces or simplifies fractions that are not in reduced form and avoids negative denominators. b) Overload the addition, subtraction, multiplication and division operators for this class. c) Overload the relational and equality operators for this class.
8.	Imagine a publishing company which does marketing for book and audiocassette versions. Create a class publication that stores the title (a string) and price (type float) of a publication. From this class derive two classes: book, which adds a page count (type int), and tape, which adds a playing time in minutes (type float). Write a program that instantiates the book and tape classes, allows user to enter data and displays the data members. If an exception is caught, replace all the data member values with zero values.
9.	Write a function in C++ to count and display the number of lines not starting with alphabet 'A' present in a text file "STORY.TXT". Example: If the file "STORY.TXT" contains the following lines, The roses are red. A girl is playing there. There is a playground. An aeroplane is in the sky. Numbers are not allowed in the password. The function should display the output as 3.
10.	Write C++ Program with base class convert declares two variables, val1 and val2, which hold the initial and converted values, respectively. It also defines the functions getinit( ) and getconv( ), which return the initial value and the converted value. These elements of convert are fixed and applicable to all derived classes that will inherit convert. However, the function that will actually perform the conversion, compute( ), is a pure virtual function that must be defined by the classes derived from convert. The specific nature of compute( ) will be determined by what type of conversion is taking place.

Savitribai Phule Pune University, Pune
First year of MCA (2019 Course)
310908: Programming Language Laboratory

Teaching Scheme:	Credit	Examination Scheme:
PR: 02 Hours/Week	01	TW: 50 Marks PR: 50 Marks

Operating System recommended :- 64-bit Open source Linux or its derivative
Programming tools recommended: -Open Source C Programming tool like GCC.

Prerequisites: Basic Programming Skills

Course Objectives:

- To learn and acquire art of computer programming
- To learn to program in C
- To study basic Linux/Unix commands

Guidelines for Instructor's Manual

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

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of 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, algorithm, flowchart, Design, test cases, conclusion/analysis. **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 journal may be avoided. Use of DVD 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 Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of student. 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 Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student’s for advanced learning, understanding of the fundamentals, effective and efficient implementation.

So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

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 need to 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 among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned.

Suggested List of Laboratory Assignments

PART I – Basics of C-Programming

1.	Write a C program to accept five numbers from console and then to display them back on console in ascending order.
2.	Write a C program to calculate the sum of all numbers from 0 to 100 (both inclusive) that are divisible by 4
3.	Write a C program to accept the length of three sides of a triangle from console and to test and print the type of triangle – equilateral, isosceles, right angled, none of these.
4.	Write a C program to accept a string from console and to display the following on console: (a) Total number of characters in the string (b) Total number of vowels in the string (c) Total number of occurrence of character ‘a’ in the string. (d) Total number of occurrence of string ‘the’ in the string.
5.	Write a program in C to reverse the digits of a given integer.
6.	Write a program in C to read an integer and display each of the digit of the integer in English.
7.	Write a program in C to generate first 20 Fibonacci numbers
8.	Write a program in C to generate prime numbers between 1 and *n*.
9.	Write a program in C to compute the GCD of the given two integers
10.	Write a program in C to compute the factorial of the given positive integer using recursive function.
11.	Write a program in C to compute the roots of a quadratic equation.
12.	Write a program in C to sort *n* integers using bubble sort.
13.	Write a program in C to compute addition/subtraction/multiplication of two matrices. Use functions to read, display and add/subtract/multiply the matrices.
14.	Write a program in C to carry out following operations on strings using library functions a. To concatenate a string S_2 to string S_1. (a) To find the length of a given string (b) To compare two strings S_1 and S_2. (c) To copy a string S_2 to another string S_1.
15.	A data file contains a set of examination scores followed by a trailer record with a value of -1. Write a C program to calculate and print the average of the scores.
	PART II- Study of Basic Linux/Unix commands
16.	General Purpose commands: date, who, who am I, uname, cal, tty, stty, echo, printf, bc, script, passwd ,finger
17.	File Handling utilities: pwd, mkdir, cd, rmdir,ls,cat, cp, mv, rm, chmod, chown, chgrp, file, find, ln, ulink, ulimit,umask,touch
18.	Process Related Commands: ps, kill, nohup, at, batch, crontab, fg, bg, jobs
19.	Filters: cat, head, tail, cut, paste, cmp, comm, diff, sort, more, less, pg, tr, uniq etc….
20.	Network Related commands :telnet, ftp, rlogin, arp
21.	Disk and backup utilities: df, du, cpio ,tar
22.	Advanced filters (grep, sed,awk)grep: (grep, egrep,fgrep)

Savitribai Phule Pune University

FIRST YEAR OF MCA (2019 Course)

310909 AC1-I: Humanities and Social Sciences

Objective of Humanities and Social Science (HSS) is to produce well-rounded engineers, not only having good technological skills but also with the ability to interact with different organs of an organization.
HSS is concerned with society and the relationships among individuals within a society. It in turn has many branches, each of which is considered a "social science". The main social sciences include economics, political science, human geography, demography and sociology. In a wider sense, social science also includes some fields in the humanities such as anthropology, archaeology, history, law and linguistics.

Course Objectives:

- Human and social development;
- Contemporary national and international affairs;
- Emergence of Indian society and Economics

Course Outcomes:

On completion of the course, student will be able to–

- Make engineering and technology students aware of the various issues concerning human and society.
- These issues will help to sensitize students to be broader towards the social, cultural, economic and human issues, involved in social changes.
- Able to understand the nature of the individual and the relationship between the self and the community
- Understanding major ideas, values, beliefs, and experiences that have shaped human history and cultures

Course Contents:

1. **Indian Society :** Structure of Indian Society, Indian Social Demography– Social and Cultural, Differentiations: caste, class, gender and tribe; Institutions of marriage, family and kinship-Secularization –Social Movements and Regionalism- Panchayatraj Institutions; Indian constitution; Affirmative Action Programme of the Government- various reservations and commissions.
2. **Social Development:** Scientific approach to the study of human beings. Evolution of human kind, social change and evolution. National policy on education, health and health care and human development.
3. **Sectorial Development:** Agriculture: Technology changes, Green revolutions, Employment Rural & Urban, Government Schemes. Industrial Development: Strategies, Public & Private Sectors, Categories, infrastructure, Consumer Awareness.
4. **Environment & Ecology:**Ecosystems: Structure, Working, components. Pollution: Water & Air Pollution, Global Warming, Control Strategies, International Treaties. Energy Sources: Renewable & Non Renewable, Hydro power, Biomass, Ocean, Geothermal & Tidal. Global Environmental Issues: Population Growth.

Reference Books:

1. Krugman, “International Economics”, Pearson Education, ISBN-13:000-01334-23646
2. Prakash, “The Indian Economy”, Pearson Education, ISBN-8131758931
3. Thursen Gerald, “Engineering Economics”, Prentice Hall, ISBN-10:0138221227
4. C.S. Rao, “Environmental Pollution Control Engineering”, New Age International Pvt. Ltd, ISBN-812241835X
5. Rangarajan, “Environmental Issues in India”, Pearson Education, ISBN-10:8131708101
6. University of Delhi, “The Individual & Society”, Pearson Education. ISBN-8131704173
7. Wikipedia.org / wiki /social studies.

Savitribai Phule Pune University
FIRST YEAR OF MCA (2019 Course)
310909AC1-II: Road Safety

Road transport remains the least safe mode of transport, with road accidents representing the main cause of death of people. The boom in the vehicle population without adequate road infrastructure, poor attention to driver training and unsatisfactory regulation has been responsible for increase in the number of accidents. India‘s vehicle population is negligible as compared to the World statistics; but the comparable proportion for accidents is substantially large. The need for stricter enforcement of law to ensure greater safety on roads and an environment-friendly road transport operation is of paramount importance. Safety and security are growing concerns for businesses, governments and the traveling public around the world, as also in India. It is, therefore, essential to take new initiatives in raising awareness, skill and knowledge of students as one of the ibid stake holders who are expected to follow the rules and policies of the government in order to facilitate safety of individual and safe mobility of others.

Course Contents:

1. Existing Road Transport Scenario
2. Accident Causes & Remedies
3. Road Accident Investigation & Investigation Methods.
4. Regulatory / Legislative Provisions for Improving Road Safety
5. Behavioural Training for Drivers for Improving Road Safety
6. Road Safety Education

Books:

1. Road Accidents in India Issues & Dimensions , Ministry of Road Transport & Highways Government of India (www.unescap.org/sites/default/files/2.12.India_.pdf)
2. Road Safety in India- Insights and analysis, http://indiatransportportal.com/wp-content/uploads/2012/11/Road_safety_2012.pdf
3. Road User‘s Handbook, ROADS & MARITIME PUBLICATIONS
4. Improving Road Safety in Developing Countries‖, The national Academic Press

SEMESTER II

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310910: Probability & Statistics

Teaching Scheme: TH: 03 Hours/Week	Credit 03	Examination Scheme: In_Sem: 30 Marks End_Sem : 70 Marks

Prerequisites: Basic knowledge of Mathematics and Statistics

Course Objectives:

- To provide an introduction to probability and statistics
- To make an inference about a population of interest based on information obtained from a sample of measurements from that population
- Be able to distinguish between discrete and continuous random variables
- Be able to compute and interpret the expected value, variance, and standard deviation for a discrete random variable

Course Outcomes:

On completion of the course, student will be able to–

- Apply statistical concepts to solve basic problems.
- Solve the problems of Discrete Distributions and Continuous Distributions.
- Explain various Descriptive Statistical concepts
- Describe Hypothesis and its estimates.
- Analyze Categorical Data using Statistical Quality Control techniques

Course Contents

Unit I	Introduction To Probability	07 Hours

Introduction to probability, sample space and events, permutations and combinations, Axioms of probability, conditional probability, Bayes' Theorem

Unit II	Descriptive Statistics	07 Hours

Concept of Population, sample, Types of Sampling, Random Sampling Frequency distributions: Mean, Median, Mode, Variance and Standard Deviation. Co-relation, regression and their methods

Unit III	Discrete Distributions	07 Hours

Random Variables, Discrete probability densities, cumulative Distribution, mathematical Expectations, Mean, Variance, Geometric Distribution, Binomial Distribution

Unit IV	Continuous Distributions	07 Hours

Continuous Random Variable and Probability Density, Cumulative Distribution, Expected Value, Mean and Variance, Normal Distribution, Joint Distributions, Joint Density and Marginal Density (Discrete and Continuous), Independence

Unit V	Hypothesis Testing	07Hours

Student-t Distribution, Hypothesis Testing, Significance Testing, Hypothesis and significance Tests on the Mean. Inference on proportions: Estimating proportions, Confidence interval on p, Sample size for estimating p, Testing Hypothesis on a proportion

Chi-squared Goodness of Fit Tests, Testing for independence: r * c Test for Independence

Unit VI	Statistical Quality Control	07 Hours
Statistical Quality Control: Properties of control charts, Shewhart control charts: Sample Mean chart, R chart, P chart, C charts. Acceptance Sampling		

Books:

Text :

1. Kishor S. Trivedi, “Probability and Statistics with Reliability, Queuing and Computer Science Applications”, Wiley , Second Edition, ISBN:9781119285441
2. Sheldon Ross, “A first Course in Probability”, Prentice Hall, ISBN-13: 978-0-13-603313-4
3. Spiegel, Schiller, “Probability and Statistics” 2nd Edition, TMH, ISBN: 0-07-058610-1

Reference :

1. Papoulis, Pillai, “Probability, Random Variables and Stochastic Processes”, 4th Edition ISBN:0-07-048658-1.
2. Veerarajanl, “Probability, Statistics And Random”, Tata McGraw-Hill, ISBN:0-07-049482-7
3. S.C. Gupta, V.K Kapoor, S. Chand, “Fundamentals of Mathematical Statistics”, Sultan *Chand* & Sons, ISBN-10: 8180545288.
4. D .P. Apte , “Probability and Combinatorics”, Excel Books, ISBN-13: 978-8174465207
5. G. Haribhaskaran, “Probability Queuing Theory & Reliability Engineering”, Laxmi publication, ISBN: *ISBN* 13: 9788170089445.
6. S. Palaniammal, “Probability and Queuing Theory”, PHI, 10: 8120342445 ISBN 13: 9788120342446

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310911: Systems Programming and Operating System

Teaching Scheme:	Credit	Examination Scheme:
TH: 04 Hours/Week	04	In_Sem: 30 Marks End_Sem : 70 Marks

Prerequisites: Fundamental of Computer Organisation

Course Objectives:

- To introduce basic concepts of system programming
- To learn and understand the concept of operating system and management policies adopted by operating system as pertaining with processes
- To provide the knowledge of basic concepts towards concurrency control and deadlock.
- To get familiar with different approaches of memory management
- To understand structure and organisation of file system and disk management

Course Outcomes:

On completion of the course, student will be able to–

- Describe the structure and design of assemblers, linker, loader and compiler
- Explain knowledge of operating system and process management concept
- Appreciate role of process synchronization towards increasing throughput of system
- Interpret the mechanisms adopted for memory management
- Differentiate file management and disk scheduling methods

Course Contents

Unit I	System Programming	08 Hours

Introduction : Components of System Software, Language Processing Activities, Fundamentals of Language Processing

Assemblers: System program and application program, Features of assembly language, types of statement**,** Advanced assembler directive**,** Structure of an assembler, Data structures used, Two-Pass Assemblers, One-Pass Assemblers, Forward reference problem, Cross assembler.

Macro Processors: Concept and need, Features of MASM (No design aspects expected)

Unit II	Linkers, Loaders and Compilers	08 Hours

Linker and Loader : Linking , Concept of binding : static and dynamic, program relocation, loader schemes: compile and go, general loader scheme, absolute loaders, direct linking loaders, case study of MS-DOS Linker and Debugger,

Compiler : Introduction to Compilers, Structure of compiler – Lexical analysis, syntax analysis, semantic analysis, intermediate code generation, code optimization, code generation

Unit III	Operating System	08 Hours

Operating Systems: Introduction, Evolution of OS, User view and system view, Operating system functions, Mainframe systems -Batch systems, Multi-programming Systems, Time sharing systems, Desktop systems, Multiprocessor systems, Real Time Systems, Distributed systems ,OS Structure-System Calls, system program, interrupt mechanism .

<table>
<tr><td colspan="3"><b>Process Management</b> : Process Concept, Process states, Process control block, Threads, CPU scheduler, Preemptive and Non-preemptive Scheduling, Scheduling criteria, Types of scheduling algorithms: First Come First Served, Shortest Job Scheduling, Round Robin, Priority, Multilevel queue scheduling and Multilevel feedback queue scheduling</td></tr>
<tr><td><b>Unit IV</b></td><td><b>Concurrency Control and Deadlock</b></td><td><b>08 Hours</b></td></tr>
<tr><td colspan="3"><b>Concurrency Control:</b> Critical section problem<b>,</b> Concurrency and Race Conditions, Mutual exclusion requirements, Software and hardware solutions, Semaphores, Monitors, Classical IPC problems and solutions.
<b>Deadlock</b> : Characterization, Detection, Recovery, Avoidance and Prevention.</td></tr>
<tr><td><b>Unit V</b></td><td><b>Memory Management</b></td><td><b>08 Hours</b></td></tr>
<tr><td colspan="3"><b>Memory management:</b> Contiguous and non-contiguous, Memory partitioning: Fixed and Variable Partitioning, Swapping and overlap swapping, Paging, Segmentation and Demand Paging, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit),Page Replacement Policies (FIFO, LRU, Optimal, Other Strategies) ,Concept of Virtual Memory, Management of Virtual memory</td></tr>
<tr><td><b>Unit VI</b></td><td><b>File Management</b></td><td><b>08 Hours</b></td></tr>
<tr><td colspan="3"><b>File Management:</b> Concept, Access methods, Directory Structure, Protection, File system implementation, Directory implementation, Allocation methods, Free space management, efficiency and performance.
I/O Management : Disk structure , disk scheduling</td></tr>
<tr><td colspan="3"><b>Books:</b></td></tr>
<tr><td colspan="3"><b>Text:</b>
<b>1.</b> John Donovan, “System Programming”, McGraw Hill, ISBN 978-0--07-460482-3.
<b>2.</b> Silberschatz, Galvin, Gagne, "Operating System Principles", 9th Edition, Wiley, ISBN 978-1-118-06333-0</td></tr>
<tr><td colspan="3"><b>References:</b>
<b>1.</b> Dhamdhere D., "Systems Programming and Operating Systems", McGraw Hill, ISBN 0 - 07 - 463579 – 4
<b>2.</b> Randal Bryant and David O'Hallaron, “Computer Systems: A Programmer's Perspective” , Pearson, ISBN 10: 0-13-610804-0
<b>3.</b> Stallings W., "Operating Systems", 6th Edition, Prentice Hall, ISBN-978-81-317-2528-3.
<b>4.</b> John. R. Levine, Tony Mason and Doug Brown, “Lex and Yacc”, O'Reilly, 1998, ISBN: 1-56592-000-7</td></tr>
</table>

Savitribai Phule Pune University, Pune
First year of MCA (2019 Course)
310912: Database Management Systems

Teaching Scheme: **TH: 03 Hours/Week**	**Credit** **03**	**Examination Scheme:** **In_Sem: 30 Marks** **End_Sem : 70 Marks**

Prerequisites: 1.Data structures
2. Discrete structures

Course Objectives:

- To understand the fundamental concepts of database management.
- To expose the students to SQL and PL/SQL.
- To make the students understand the relational model.
- To describe the fundamentals of Query Processing and Optimization.
- To familiarize with the basic issues of transaction processing and concurrency control.
- To learn and understand various Database Architectures and Applications

Course Outcomes:

On completion of the course, student will be able to –

- Define basic functions of DBMS & RDBMS.
- Populate and query a database using SQL DML/DDL commands.
- Design and implement a database schema for a given problem-domain.
- Design the Query Processor.
- Explain transaction management in relational database System.
- Describe different database architecture and analyses the use of appropriate architecture in real time environment.

Course Contents

Unit I	**Introduction To DBMS**	**07 Hours**

Introduction: Database Concepts, Database System Architecture, Data Modelling: Data Models, Basic Concepts, entity, attributes, relationships, constraints, keys.

E-R and EER diagrams: Components of E-R Model, conventions, converting E-R diagram into tables, EER Model components, converting EER diagram into tables, legacy system model.

Relational Model: Basic concepts, Attributes and Domains, Codd's Rules.

Relational Integrity: Domain, Entity, Referential Integrities, Enterprise Constraints, Schema Diagram.

Relational Algebra: Basic Operations, Selection, projection, joining, outer join, union, difference, intersection, Cartesian product, division operations (examples of queries in relational algebraic using symbols).

Unit II	**SQL And PL/SQL**	**07 Hours**

SQL: Characteristics and advantages, SQL Data Types and Literals, DDL, DML, DCL, TCL,SQL Operators, Tables: Creating, Modifying, Deleting, Views: Creating, Dropping, Updating using Views, Indexes, SQL DML Queries: SELECT Query and clauses, Set Operations, Predicates and Joins, Set membership, Tuple Variables, Set comparison, Ordering of Tuples, Aggregate Functions, Nested

Queries, Database Modification using SQL Insert, Update and Delete Queries. **PL/SQL:** concept of Stored Procedures & Functions, Cursors, Triggers, Assertions, roles and privileges, Embedded SQL, Dynamic SQL.		
Unit III	**Relational Database Design**	**07 Hours**
Database Design: Features of Good Relational Designs, Normalization, Atomic Domains and First Normal Form, Decomposition using Functional Dependencies, Algorithms for Decomposition, 2NF, 3NF, BCNF, Modelling Temporal Data.		
Unit IV	**Advanced Databases**	**07 Hours**
Database Architectures: Centralized and Client-Server Architectures, 2 Tier and 3 Tier Architecture, Introduction to Parallel Databases, Key elements of Parallel Database Processing, Architecture of Parallel Databases, **Distributed Database:** Introduction to Distributed Databases, Architecture of Distributed Databases, Distributed Database Design		
Unit V	**Object Oriented Databases and XML**	**07 Hours**
Structure types and inheritance in SQL: Structure types, Type inheritance, Table inheritance, Array and multiset types in SQL: creating and accessing collection values, Querying collection, nesting and unnesting, Object identity and reference types in SQL, implementing object relational features **XML:** Relational Database Tables and XML, Generating XML pages using Basic SQL, Oracle Database and XML		
Unit VI	**NoSQL: HBASE**	**07 Hours**
NoSQL: Concept of Big Data, The Problem with Relational Database Systems,Non-relational Database Systems, Not-Only SQL or NoSQL? **HBASE:** Building Blocks : Tables, Rows, Columns, and Cells, Auto-Shading, Introduction to HBASE Architecture **Introduction to Data Models** - Graph Databases, Schema-less Databases. **Introduction to Distribution Models** – single server, sharding, master-slave replication, peer to peer replication		
Books:		
Text : **1.** Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 6th edition **2.** Connally T, Begg C., "Database Systems", Pearson Education, ISBN 81-7808-861-4		
Reference: **1.** C J Date, “An Introduction to Database Systems”, Addison-Wesley, ISBN: 0201144719. **2.** Atul Kahate, “Introduction to Database Management Systems”, Pearson Education, Pearson Education, ISBN: 9788131788417, 8131788415. **3.** Raghu Ramakrishnan, “Database Management Systems”, Fourth Edition, Tata McGraw Hill, 2010 **4.** Joy A. Kreibich, “Using Sqlite”, O'REILLY, ISBN: 13:978-93-5110-934-1 **5.** Garrett Grolemund, “Hands-on Programming with R”, O'REILLY, ISBN : 13:978-93-5110-728-6		

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310913: Java Programming

		Examination Scheme:
Teaching Scheme:	**Credit**	**In_Sem: 30 Marks**
TH: 03 Hours/Week	**03**	**End Sem: 70 Marks**

Prerequisites: Object oriented programming concepts.

Course Objectives:

- Understand fundamentals Java.
- Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
- Be able to use the Java SDK environment to create, debug and run simple Java programs.

Course Outcomes:

On completion of the course, student will be able to–

- Know the structure and model of the Java programming language.
- Use the Java programming language for various programming technologies.
- Develop software in the Java programming language (application).

Course Contents

Unit I	Introduction to Java	07 Hours

Fundamentals of Object-oriented Programming, Evolution of Java, Overview of Java Language: Data types in Java, Arrays, Strings and Vectors: Creating one dimensional and multidimensional array, Strings, Vectors, Wrapper classes, Enumerated types, Annotations.

Unit II	Object Oriented Programming	07 Hours

Classes ,Objects And Methods: Defining class , Methods, Creating objects , Accessing Class members, Static Methods , Finalize Methods, Visibility Control, Method overloading, Method Overriding, Recursion. Interfaces, Constructors and finalizes Methods.

Java API Packages, Using System Packages, Naming conventions, Creating Packages and Jar Files, Accessing and using a package, Hiding Classes

Unit III	Multithreading and Exception Handling	07 Hours

Creating threads, Extending Thread Class, Stopping and Blocking a thread, Life cycle of a thread, Using thread method, Thread exceptions, implementing the Run able interface, Inter-thread communication. Managing Errors and Exceptions: Types of errors, Exceptions, Syntax of exception handling code, multiple catch statements, Throwing your own exception, Using exceptions for debugging.

Unit IV	Graphics Programming	07 Hours

The Graphics class, Lines and Rectangles, Circles, Arc and ellipses, Polygons, Drawing Bar charts, AWT Package and Swings, Applet Programming

Unit V	Managing Files & I/O Handling	07 Hours

Files and Streams, Stream classes, Byte Stream Classes , Character Stream Classes, Using Streams, Reading / writing bytes and characters , Interactive Input and Output, Other Stream classes.

Unit VI	**J2EE & JDBC**	**07 Hours**

Introduction to J2EE: J2EE Overview, Why J2EE?, J2EE Architecture, J2EE Container. JDBC: JDBC Introduction, JDBC Architecture, Types of JDBC Drivers, JDBC versus ODBC and other API's, The java.sql package, Connecting to databases, Manipulating records of a Result Set object through User Interface, The JDBC Exception classes, Data Manipulation (using Prepared Statements, Joins, Transactions, Stored Procedures), Data navigation, Result set Metadata.

Books:

Text :

1. Steven Holzner et al. "Java 2 Programming", Black Book, Dreamtech Press.
2. H.M. Deitel, P.J. Deitel, "Java - How to Program", PHI Publication, 6th Edition.

Reference:

1. E. Balagurusamy, "Programming with Java – A Primer", Tata – McGraw-Hill Publication, 4th Edition, 2010.
2. Jim Keogh, "The Complete Reference- J2EE",TMH, ISBN 0-07-222710-9.
3. Bruce Eckel, "Thinking in Java", PHI Publication, ISBN *0-13-187248*-6.
4. Patric Naughton, Michael Morrison, "The Java Handbook", McGraw Hill Publication, ISBN 007235447x
5. Tim Lindholm, Frank Yellin, Bill Joy, KathiWalrath, "The Java Virtual Machine Specification", Addison Wesley Publication.

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310914: Computer Organization

Teaching Scheme:	Credit	Examination Scheme:
TH: 03 Hours/Week	03	In_Sem: 30 Marks End_Sem : 70 Marks

Course Objectives:

- To understand basic concepts required for digital computer
- To understand the structure, function and characteristics of computer systems.
- To explain the function of each element of a memory hierarchy, identify and compare different methods for computer I/O.
- To understand the design of the various I/O functions components of digital computers.
- To compare simple computer architectures and organizations based on established performance metrics.
- To identify the elements of modern computer and explain their impact.

Course Outcomes:

On completion of the course, student will be able to–

- Demonstrate computer architecture concepts related to design of modern processors, memories and I/Os.
- Analyse the principles of computer architecture using examples drawn from commercially available computers.
- Explain various I/O functional components of digital computer.
- Explain various modern concepts in digital computer.

Course Contents

Unit I	Introduction & Data Representation , Boolean Algebra	07 Hours

Concept of Digital Computer, Types of Software – System software / Application software / Utility Software., Compilers, Interpreters, Assemblers, Linker, Loader. Binary, Octal, Hexadecimal and their inter-conversion, 1's and 2's complement. Binary Arithmetic. & Number Systems – BCD, EBCDIC, ASCII, De-Morgan's Theorem, Duality Theorem, K-Map, Sum of product, Product of Sum, Algebra Rules, Laws, Logic Circuits, NOT,AND, OR, NAND,NOR, XOR, XNOR, Gated diagrams.

Unit II	Structure of Computers	07 Hours

Functional units of Computer , Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance parameters (Equation), The Arithmetic and Logic Unit, addition and subtraction of signed numbers, design of adder and fast adder, carry look ahead addition, multiplication of positive numbers, signed operand multiplication, booths algorithm, fast multiplication, integer division. Floating point representation and operations.

Unit III	Memory System	07 Hours

Memory Hierarchy, Primary Memory – DRAM, SDRAM, DDR, RDRAM. ROM, PROM, EPROM, EEPROM, Cache memory Structure, Cache memory principles, Elements of cache design- cache address, size, mapping functions, replacement algorithms, write policy, line size, number of cache, one level and two level cache, performance characteristics of two level cache- locality & operations. Case Study- Pentium IV onwards cache organization.

DMA, DMA interfacing with processor. Memory Locations and Addresses, **Case Study - Advanced processor (Pentium IV onwards)**

Unit IV | Input / Output & Processor Organization | 07 Hours

Instruction and Instruction Sequencing, Addressing Modes, Basic Input / Output Operations. Instruction types, Interrupt driven I/O- interrupt processing, design issues. Processor organization, Register organization- user visible registers, control and status registers, Instruction Execution cycle. **Case Study-** registers organization of microprocessor 8086.

Unit V | Superscalar Concept | 07 Hours

I/O Instruction Execution – with timing diagram, Instruction **Cycle-** The machine cycle and Data flow. **Instruction Pipelining-** Pipelining Strategy, pipeline performance, pipeline hazards, dealing with branches, **Case Study**- pipelining in Pentium. Pipelining and Superscalar Operation, Clock Rate, Instruction set: CISC and RISC.

Unit VI | Parallel Processing | 07 Hours

Parallel Processing, Concept and Block Diagram, Types (SISD, SIMD, MIMD, MISD), **Case Study** Multi processor Organisation and performance measure Clusters Concept, Cluster Architecture.

Books:

Text :

1. W. Stallings, "Computer Organization and Architecture: Designing for performance", Pearson Education/ Prentice Hall of India, 2003, ISBN 978-93-325-1870-4, 7th Edition.
2. Zaky S, Hamacher, "Computer Organization", 5th Edition, McGraw-Hill Publications, 2001, ISBN- 978-1- 25-900537-5, 5th Edition.

Reference :

1. John P Hays, "Computer Architecture and Organization", McGraw-Hill Publication, 1998, ISBN:978-1-25-902856-4, 3rd Edition.
2. Miles Murdocca and Vincent Heuring, "Computer Architecture and Organization- an integrated approach", Wiley India Pvt. Ltd, ISBN:978-81-265-1198-3, 2nd Edition
3. A. Tanenbaum, "Structured Computer Organization", Prentice Hall of India, 1991 ISBN: 81 – 203 – 1553 – 7, 4th Edition
4. Patterson and Hennessy, "Computer Organization and Design", Morgan Kaufmann Publishers In, ISBN 978-0-12-374750-1, 4th Edition.

Savitribai Phule Pune University, Pune
First year of MCA (2019 Course)
310915: Database Management Systems Laboratory

Teaching Scheme: PR: 04 Hours/Week	Credit 02	Examination Scheme: TW: 50 Marks PR : 50 Marks

Companion Course: Database Management System

Guidelines for Instructor's Manual

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

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of 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, algorithm, flowchart, Design, test cases, conclusion/analysis. **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 journal may be avoided. Use of DVD 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 Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of student. 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 Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student’s for advanced learning, understanding of the fundamentals, effective and efficient implementation.

So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

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 need to 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 among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a

mini-project based on the concepts learned.	
Suggested List of Laboratory Assignments **(Instructor may design a newer one)**	
Group A- Database Programming Languages – SQL, PL/SQL	
1.	Study of Open Source Relational Databases : MySQL
2.	Design and Develop SQL DDL statements which demonstrate the use of SQL objects such as Table, View, Index, Sequence, Synonym
3.	Design at least 10 SQL queries for suitable database application using SQL DML statements: Insert, Select, Update, Delete with operators, functions, and set operator.
4.	Design at least 10 SQL queries for suitable database application using SQL DML statements: all types of Join, Sub-Query and View.
5.	Unnamed PL/SQL code block: Use of Control structure and Exception handling is mandatory. Write a PL/SQL block of code for the following requirements:- Schema: 1. Borrower(Rollin, Name, Date of Issue, Name of Book, Status) 2. Fine(Roll_no, Date,Amt) • Accept roll_no & name of book from user. • Check the number of days (from date of issue), if days are between 15 to 30 then fine amount will be Rs 5per day. • If no. of days>30, per day fine will be Rs 50 per day & for days less than 30, Rs. 5 per day. • After submitting the book, status will change from I to R. • If condition of fine is true, then details will be stored into fine table. **Frame the problem statement for writing PL/SQL block inline with above statement.**
6.	Cursors: (All types: Implicit, Explicit, Cursor FOR Loop, Parameterized Cursor) Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped. **Frame the separate problem statement for writing PL/SQL block to implement all types of Cursors inline with above statement. The problem statement should clearly state the requirements.**
7.	PL/SQL Stored Procedure and Stored Function. Write a Stored Procedure namely proc_Grade for the categorization of student. If marks scored by students in examination is <=1500 and marks>=990 then student will be placed in distinction category if marks scored are between 989 and900 category is first class, if marks 899 and 825 category is Higher Second Class Write a PL/SQL block for using procedure created with above requirement. Stud_Marks(name, total_marks) Result(Roll,Name, Class) **Frame the separate problem statement for writing PL/SQL Stored Procedure and function, inline with above statement. The problem statement should clearly state the requirements.**
8.	Database Trigger (All Types: Row level and Statement level triggers, Before and After Triggers). Write a database trigger on Library table. The System should keep track of the records that are being updated or deleted. The old value of updated or deleted records should be added in Library_Audit table. **Frame the problem statement for writing Database Triggers of all types, in-line with above**

	statement. The problem statement should clearly state the requirements.
Group B	**Large Scale Databases**
1.	Study of Open Source NOSQL Database: MongoDB (Installation, Basic CRUD operations, Execution)
2.	Design and Develop MongoDB Queries using CRUD operations. (Use CRUD operations, SAVE method, logical operators)
3.	Implement aggregation and indexing with suitable example using MongoDB.
4.	Implement Map reduces operation with suitable example using MongoDB.
5.	Design and Implement any 5 query using MongoDB
6.	Create simple objects and array objects using JSON
7.	Encode and Decode JSON Objects using Java/Perl/PHP/Python/Ruby

Group C Mini Project : Database Project Life Cycle

Write a program to implement MogoDB database connectivity with PHP/ python/Java Implement Database navigation operations (add, delete, edit etc.) using ODBC/JDBC.

Implement MYSQL/Oracle database connectivity with PHP/ python/Java Implement Database navigation operations (add, delete, edit,) using ODBC/JDBC.

Using the database concepts covered in Part-I & Part-II & connectivity concepts covered in Part C, students in group are expected to design and develop database application with following details:

Requirement Gathering and Scope finalization

Database Analysis and Design:

- Design Entity Relationship Model, Relational Model, Database Normalization

Implementation :

- Front End : Java/Perl/PHP/Python/Ruby/.net
- Backend : MongoDB/MYSQL/Oracle
- Database Connectivity : ODBC/JDBC

Testing : Data Validation

Group of students should submit the Project Report which will be consist of documentation related to different phases of Software Development Life Cycle: Title of the Project, Abstract, Introduction, scope, Requirements, Data Modeling features, Data Dictionary, Relational Database Design, Database Normalization, Graphical User Interface, Source Code, Testing document, Conclusion. Instructor should maintain progress report of mini project throughout the semester from project group and assign marks as a part of the term work

Savitribai Phule Pune University, Pune
First year of MCA (2019 Course)
310916: Systems Programming and Operating System Laboratory

Teaching Scheme: **TH: 02 Hours/Week**	**Credit** **01**	**Examination Scheme:** **TW: 50 Marks**

Companion Course: Systems Programming and Operating System

Course Objectives:

- To implement basic language translator by using various needed data structures
- To implement basic Macro-processor
- To design and implement Dynamic Link Libraries
- To implement scheduling schemes

Course Outcomes:

On completion of the course, student will be able to–

- Understand the internals of language translators
- Handle tools like LEX & YACC.
- Understand the Operating System internals and functionalities with implementation point of view.

Guidelines for Instructor's Manual

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

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of 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, algorithm, flowchart, Design, test cases, conclusion/analysis. **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 journal may be avoided. Use of DVD 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 Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of student. 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 Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student’s for advanced learning, understanding of the fundamentals, effective and efficient implementation.

So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

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 need to 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 among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned.

Suggested List of Laboratory Assignments

Group A

1. Design suitable data structures and implement pass-I of a two-pass assembler for pseudo-machine in Java using object oriented feature. Implementation should consist of a few instructions from each category and few assembler directives.
2. Implement Pass-II of two pass assembler for pseudo-machine in Java using object oriented features. The output of assignment-1 (intermediate file and symbol table) should be input for this assignment.
3. Design suitable data structures and implement pass-I of a two-pass macro-processor using OOP features in Java
4. Write a Java program for pass-II of a two-pass macro-processor. The output of assignment-3 (MNT, MDT and file without any macro definitions) should be input for this assignment.

Group B

1. Write a program to create Dynamic Link Library for any mathematical operation and write an application program to test it. (Java Native Interface / Use VB or VC++).
2. Write a program using Lex specifications to implement lexical analysis phase of compiler to generate tokens of subset of ‘Java’ program.
3. Write a program using Lex specifications to implement lexical analysis phase of compiler to count no. of words, lines and characters of given input file.
4. Write a program using YACC specifications to implement syntax analysis phase of compiler to validate type and syntax of variable declaration in Java.
5. Write a program using YACC specifications to implement syntax analysis phase of compiler to recognize simple and compound sentences given in input file.

Group C

1. Write a Java program (using OOP features) to implement following scheduling algorithms: FCFS , SJF (Preemptive), Priority (Non-Preemptive) and Round Robin (Preemptive)
2. Write a Java program to implement Banker’s Algorithm
3. Implement UNIX system calls like ps, fork, join, exec family, and wait for process management (use shell script/ Java/ C programming).
4. Study assignment on process scheduling algorithms in Android and Tizen.

Group D

Write a Java Program (using OOP features) to implement paging simulation using

1. Least Recently Used (LRU)
2. Optimal algorithm

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310917: Java Programming Laboratory

Teaching Scheme: TH: 04 Hours/Week	Credit 02	Examination Scheme: TW: 50 Marks PR: 50 Marks

Prerequisites: Object oriented programming concepts.

Course Objectives:

- Understand fundamentals Java.
- Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
- Be able to use the Java SDK environment to create, debug and run simple Java programs.

Course Outcomes:

On completion of the course, student will be able to–

- Know the structure and model of the Java programming language.
- Use the Java programming language for various programming technologies.
- Develop software in the Java programming language (application).

Guidelines for Instructor's Manual

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

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of 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, algorithm, flowchart, Design, test cases, conclusion/analysis. **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 journal may be avoided. Use of DVD 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 Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of student. 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 Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student’s for advanced learning, understanding of the fundamentals, effective and efficient implementation.

So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

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 need to 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 among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned.

Suggested list of Experiments
(Instructor may design based one these)

1.	Write a program in Java with class Rectangle with the data fields width, length, area and colour. The length, width and area are of double type and colour is of string type. The methods are get_length(), get_width(), get_colour() and find_area(). Create two objects of Rectangle and compare their area and colour. If the area and colour both are the same for the objects then display " Matching Rectangles", otherwise display " Non-matching Rectangle".
2.	Write a Java program which imports user defined package and uses members of the classes contained in the package.
3.	Write a Java program which implements interface.
4.	Create an applet with three text Fields and four buttons add, subtract, multiply and divide. User will enter two values in the Text Fields. When any button is pressed, the corresponding operation is performed and the result is displayed in the third Text Fields.
5.	Write a java program to crete User defined exception to check the following conditions and throw the exception if the criterion does not meet. a. User has age between 18 and 55 b. User stays has income between Rs. 50,000 – Rs. 1,00,000 per month c. User stays in Pune/ Mumbai/ Bangalore / Chennai d. User has 4-wheeler Accept age, Income, City, Vehicle from the user and check for the conditions mentioned above. If any of the condition not met then throw the exception.
6.	Implement Java program to implement a base class consistingof the data members such as name of the student, roll number and subject. The derived class consists of the data members subject code, internal assessment and university examination marks. The program should have the facilities. i) Build a master table ii) List a table iii) Insert a new entry iv) Delete old entry v) Edit an entry vi) Search for a record. Use virtual functions.
7.	Write a program to create multiple threads and demonstrate how two threads communicate with each other.
8.	Write a java program that creates an output file, writes information to it, closes the file and open it again as an input file and read the information from the file.
9.	Write a java program to create simple application to access data base using JDBC.For example, write a program to create a database for reservation system using information such as Name, sex, age, starting place of journey and destination. Program should have following facilities a) To display entire passenger list b) To display particular record c) To update record d) To delete and sort record. Use Exception Handling for data verification

Savitribai Phule Pune University, Pune
First Year of MCA (2019 Course)
310918: Project Based Learning-I

Teaching Scheme: TH: 02 Hours/Week	Credit 01	Examination Scheme: TW : 50 Marks

For better learning experience, along with traditional classroom teaching and laboratory learning; project based learning has been introduced with objective to motivate students to learn by working in group cooperatively to solve a problem.

Project-based learning (PBL) is a student-centric pedagogy that involves a dynamic classroom approach in which it is believed that students acquire a deeper knowledge through active exploration of real-world challenges and problems. Students learn about a subject by working for an extended period of time to investigate and respond to a complex question, challenge, or problem. It is a style of active learning and inquiry-based learning. (Reference: Wikipedia).

Problem based learning will also redefine the role of teacher in learning process.

Along with communicating knowledge to students, often in a lecture setting, the teacher will also to act as an initiator and facilitator in the collaborative process of knowledge transfer and development.

Course Objectives:

1. To emphasizes learning activities that are long-term, interdisciplinary and student-centric.
2. To inculcate independent learning by problem solving with social context.
3. To engages students in rich and authentic learning experiences.
4. To provide every student the opportunity to get involved either individually or as a group so as to develop team skills and learn professionalism.

Course Outcomes:

CO1: Project based learning will increase their capacity and learning through shared cognition. CO2: Students able to draw on lessons from several disciplines and apply them in practical way. CO3: Learning by doing approach in PBL will promote long-term retention of material and replicable skill, as well as improve teachers' and students' attitudes towards learning.

Group Structure:

Working in faculty-supervised groups, the students plan, manage and complete a project which addresses the stated problem. Taken together, the elements of problem, project and group that are the cornerstones of the problem

- There should be team/group of 4-5 students
- A supervisor/mentor teacher

Selection of Project/Problem:

The problem-based project oriented model for learning is recommended. The model begins with the identifying 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 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 exemplarity, 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, analyze, 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 project slot/project day. Weekly review the work is appreciated. During process of monitoring and continuous assessment AND evaluation the individual and team performance is to be measured. PBL is monitored and continuous assessment is done by 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 and the provision of appropriate resources and services. 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.

- Individual assessment for each student (Understanding individual capacity, role and involvement in the project)
- Group assessment (roles defined, distribution of work, intra-team communication and togetherness)
- Documentation and presentation

Evaluation and Continuous Assessment:

It is recommended that the all activities are to be record and regularly, regular assessment of work to be done and proper documents are to be maintained at college end by both students as well as mentor (you may call it PBL work book).

Continuous Assessment Sheet (CAS) is to be maintained by all mentors/department and institutes.

Recommended parameters for assessment, evaluation and weightage:

- Idea Inception **(5%)**
- Outcomes of PBL/ Problem Solving Skills/ Solution Provided/ Final product **(50%)**
- (Individual assessment and team assessment)
- Documentation (Gathering requirements, design & modeling, implementation/execution, use of technology and final report, other documents) **(25%)**
- Demonstration (Presentation, User Interface, Usability etc) **(10%)**
- Contest Participation/ publication **(5%)**
- Awareness /Consideration of -Environment/ Social /Ethics/ Safety measures/Legal aspects **(5%)**

PBL workbook will serve the purpose and facilitate the job of students, guide and project coordinator. This workbook will reflect accountability, punctuality, technical writing ability and work flow of the work undertaken.

References:

- Project-Based Learning, Edutopia, March 14, 2016 .What is PBL? Buck Institute for Education.
- www.schoology.com .www.wikipedia.org

Savitribai Phule Pune University
FIRST YEAR OF MCA (2019 Course)
310919:AC2-I: Foreign Language- Japanese (Module 1)

About course:

With changing times, the competitiveness has gotten into the nerves and Being the Best at all times is only the proof of it. Nonetheless, being the best differs significantly from Communicating the best'. The best can merely be communicated whilst using the best suited Language!

Japanese is the new trend of 21st century. Not only youngsters but even the professionals seek value in it. It is the engineer's companion in current times with an assertion of a thriving future. Pune has indisputably grown to become a major center of Japanese Education in India while increasing the precedence for Japanese connoisseurs.

Japanese certainly serves a great platform to unlock a notoriously tough market & find a booming career. While the companies prefer candidates having the knowledge of the language, it can additionally help connect better with the native people thus prospering in their professional journey. Learning Japanese gives an extra edge to the resume since the recruiters consciously make note of the fact it requires real perseverance and self-discipline to tackle one of the most complex languages.

It would be easy for all time to quit the impossible; however it takes immense courage to reiterate the desired outcomes, recognize that improvement is an ongoing process and ultimately soldier on it. The need of an hour is to introduce Japanese language with utmost professionalism to create awareness about the bright prospects and to enhance the proficiency and commitment. It will then prove to be the ultimate path to the quest for professional excellence!

Course Objectives:

- To meet the needs of ever growing industry with respect to language support.
- To get introduced to Japanese society and culture through language.

Course Outcomes:

On completion of the course student will

- Have ability of basic communication.
- Have the knowledge of Japanese script.
- Get introduced to reading , writing and listening skills
- Will develop interest to pursue professional Japanese Language course.

Course Contents:

1: Introduction to Japanese Language. Hiragana basic Script, colors, Days of the week

2: Hiragana : modified Kana, double consonant, Letters combined with ya, yu, yo Long vowels, Greetings and expressions

3: Self Introduction, Introducing other person, Numbers, Months, Dates, Telephone numbers, Stating one's age.

Reference Books:

1. Minna No Nihongo, "Japanese for Everyone", Elementary Main Text book 1-1 (Indian Edition), Goyal Publishers & Distributors Pvt. Ltd.

2. http://www.tcs.com (http://www.tcs.com/news_events/press_releases/Pages/TCS-Inaugurates-Japan-centric-Delivery-Center-Pune.aspx)

Savitribai Phule Pune University
FIRST YEAR OF MCA (2019 Course)
310919:AC2-II: Environmental Studies

Environmental studies are the field that examines this relationship between people and the environment. An environmental study is an interdisciplinary course examining the interplay between the social, legal, management, and scientific aspects of environmental issues.

Course Objectives:

- Understanding the importance of ecological balance for sustainable development.
- Understanding the impacts of developmental activities and mitigation measures.
- Understand and realize the multi-disciplinary nature of the environment, its components, and inter-relationship between man and environment
- Understand the relevance and importance of the natural resources in the sustenance of life on earth and living standard

Course Outcomes:

On completion of the course, student will be able to–

- Comprehend the importance of ecosystem and biodiversity
- To correlate the human population growth and its trend to the environmental degradation and develop the awareness about his/her role towards environmental protection and prevention
- Identify different types of environmental pollution and control measures
- To correlate the exploitation and utilization of conventional and non-conventional resources

Course Contents:

1. Natural Resources: Introduction, Renewable and non-renewable, Forest, water, mineral, food, energy and land resources, Individual and conservation of resources, Equitable use of resources.
2. Ecosystems: Concept, Structure, Function, Energy flow, Ecological succession, Forest, grassland, desert and aquatic ecosystems - Introduction, characteristic features, structure and function.
3. Biodiversity: Genetic, Species and ecological diversity, Bio geographical classification of India, Value and hot spots, Biodiversity at global, national and local levels, India as mega-biodiversity nation, Threats to biodiversity, Endangered and endemic species of India, Conservation of Biodiversity, Endangered and endemic species, Conservation of biodiversity.
4. Pollution: Definition, Causes, effects and control measures of the pollution – Air, soil, Noise, Water, Marine and Thermal and Nuclear Pollution, Solid waste management, Role of Individual in Prevention of Pollution, Pollution case studies, Disaster management.

Reference Books:

1. Bharucha, E., “Textbook of Environmental Studies”, Universities Press (2005), ISBN-10:8173715408

2. Mahua Basu, “Environmental Studies”, Cambridge University Press, ISBN-978-1-107-5317-3