

ANNA UNIVERSITY
CENTRE FOR DISTANCE EDUCATION
REGULATIONS – 2020
CURRICULUM AND SYLLABUS - TENTATIVE (DISTANCE MODE)
MASTER OF COMPUTER APPLICATIONS - (MCA)

SEMESTER I

SL. NO.	COURSE TITLE	L	T	P	C
1.	Matrices, Probability and Statistics	3	2	0	4
2.	Advanced Data Structures and Algorithms	3	0	0	3
3.	Advanced Database Technology	3	0	0	3
4.	Object Oriented Software Engineering	3	0	0	3
5.	Python Programming	3	0	0	3
6.	Research Methodology and Intellectual Property Rights	2	0	0	2
7.	Advanced Database Technology Laboratory	0	0	4	2
8.	Advanced Data Structures and Python Programming Laboratory	0	0	4	2
9.	Communication Skills Enhancement I	0	0	2	1
TOTAL		17	2	10	23

TOTAL CREDITS: 23

REFERENCE BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 43rd Edition, New Delhi, 2015.
2. R.K. Jain and S.R.K Iyenger, Advanced Engineering Mathematics, Narosa Publishing House, New Delhi, 2002.
3. Devore, J.L., Probability and Statistics for Engineering and Sciences, Cengage Learning, 8th Edition, New Delhi, 2014.
4. Miller and M. Miller, Mathematical Statistics, Pearson Education Inc., Asia 7th Edition, New Delhi, 2011.
5. Richard Johnson, Miller and Freund's Probability and Statistics for Engineer, Prentice Hall of India Private Ltd., 8th Edition, New Delhi, 2011.

TENTATIVE

OUTCOMES:

- Implement a program using stack, queue, linked list data structures
- Design and Implement Tree data structures and Sets
- Apply the Graph Data structure and to find shortest path among the several possibilities
- Perform analysis of various algorithms
- Analyze and design algorithms to appreciate the impact of algorithm design in practice.

REFERENCES:

1. AnanyLevitin "Introduction to the Design and Analysis of Algorithms" 3rdEdition Pearson Education, 2015.
2. Jean Paul Tremblay and Paul G. Sorensen. "An Introduction to Data Structures with Applications", 2nd Edition, Tata McGraw Hill, New Delhi, 2017
3. Peter Drake, "Data Structures and Algorithms in Java", 4th Edition, Pearson Education 2014
4. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms", 3rd Edition, PHI Learning Private Ltd, 2012
5. V. Aho, J. E. Hopcroft, and J. D. Ullman, "Data Structures and Algorithms", 1stEdition, Pearson Education, 1983.
6. Michael T. Goodrich, "Algorithm Design: Foundations, Analysis and Internet Examples", 2nd Edition, Wiley India Pvt. Ltd, 2006.

TENTATIVE

OUTCOMES:

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

1. Design a distributed database system and execute distributed queries.
2. Use NoSQL database systems and manipulate the data associated with it.
3. Design a data warehouse system and apply OLAP operations.
4. Design XML database systems and validating with XML schema.
5. Apply knowledge of information retrieval concepts on web databases.

REFERENCES:

1. Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database System Concepts", 6th Edition, McGraw Hill, 2011.
2. R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", Seventh Edition, Pearson Education/Addison Wesley, 2017.
3. C. J. Date, A. Kannan, S. Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006.
4. Jiawei Han, MichelineKamber ,Jian Pei, "Data Mining: Concepts and Techniques", Third Edition, Morgan Kaufmann, 2012.
5. Brad Dayley, "Teach Yourself NoSQL with MongoDB in 24 Hours", Sams Publishing, First Edition, 2014.
6. ShashankTiwari, "Professional NoSQL", O'Reilly Media, First Edition, 2011.
7. Vijay Kumar, "Mobile Database Systems", John Wiley & Sons, First Edition, 2006

TENTATIVE

OBJECT ORIENTED SOFTWARE ENGINEERING

L T P C

3 0 0 3

OBJECTIVES:

- To understand the phases in object oriented software development
- To gain fundamental concepts of requirements engineering and analysis.
- To know about the different approach for object oriented design and its methods
- To learn about how to perform object oriented testing and how to maintain software
- To provide various quality metrics and to ensure risk management.

UNIT I SOFTWARE DEVELOPMENT LIFE CYCLE 9

Introduction – Object Orientation - Object Oriented Methodologies – Terminologies - Software Development Life Cycle – Conventional Software Life Cycle Models – Build and Fix Model – Waterfall Model – Prototyping Model – Iterative Enhancement Model – Spiral Model – Extreme Programming - Object Oriented Software Life Cycle Models – Selection of Software Development Life Cycle Models

UNIT II OBJECT ORIENTED REQUIREMENTS ELICITATION & ANALYSIS 9

Software Requirement - Requirements Elicitation Techniques – Initial Requirements Document – Use Case Approach – Characteristics of a Good Requirement – SRS Document – Requirements Change Management – Object Oriented Analysis : Identification of Classes and Relationships, Identifying State and Behavior – Case Study LMS - Managing Object Oriented Software Engineering: Projection Selection and Preparation – Product Development Organization – Project Organization and Management – Project Staffing

UNIT III OBJECT ORIENTED SOFTWARE DESIGN 9

Object Oriented Design – Interaction Diagrams – Sequence Diagram – Collaboration Diagrams – Refinement of Use Case Description – Refinement of Classes and Relationships – Construction of Detailed Class Diagram – Development of Detailed Design & Creation of Software Design Document - Object Oriented Methods : Object Oriented Analysis (OOA / Coad-Yourdon), Object Oriented Design (OOD/Booch) , Hierarchical Object Oriented Design (HOOD), Object Modeling Technique (OMT), Responsibility – Driven Design Case Studies : Warehouse Management System, Telecom

UNIT IV OBJECT ORIENTED TESTING AND MAINTENANCE 9

Software testing: Software Verification Techniques – Object Oriented Checklist :- Functional Testing – Structural Testing – Class Testing – Mutation Testing – Levels of Testing – Static and Dynamic Testing Tools - Software Maintenance – Categories – Challenges of Software Maintenance – Maintenance of Object Oriented Software – Regression Testing

UNIT V SOFTWARE QUALITY & METRICS 9

Need of Object Oriented Software Estimation – Lorenz and Kidd Estimation – Use Case Points Method – Class Point Method – Object Oriented Function Point – Risk Management – Software Quality Models – Analyzing the Metric Data – Metrics for Measuring Size and Structure – Measuring Software Quality - Object Oriented Metrics

TOTAL: 45 PERIODS

OUTCOMES:

- Able to identify the appropriate process model to develop the object oriented software
- Gain knowledge about requirement elicitation and analyzing techniques
- Able to choose and design suitable UML diagrams and methods
- Able to apply correct testing methods and maintain software systems.
- Able to estimate the object oriented application by applying metric data.

REFERENCES:

1. Yogesh Singh, Ruchika Malhotra, “ Object – Oriented Software Engineering”, PHI Learning Private Limited ,First edition,2012
2. Ivar Jacobson. Magnus Christerson, Patrik Jonsson, Gunnar Overgaard, “Object Oriented Software Engineering, A Use Case Driven Approach”, Pearson Education, Seventh Impression, 2009
3. Craig Larman, “Applying UML and Patterns, an Introduction to Object-Oriented Analysis and Design and Iterative Development”, Pearson Education, Third Edition, 2008.
4. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen, Kelli A. Houston, “Object Oriented Analysis & Design with Applications, Third Edition, Pearson Education,2010
5. Roger S. Pressman, “Software Engineering: A Practitioner’s Approach, Tata McGraw-Hill Education, 8th Edition, 2015.
6. Timothy C. Lethbridge and Robert Laganier, “Object – Oriented Software Engineering, Practical Software Development using UML and Java”, Tata McGraw Hill Publishing Company Limited, Second Edition, 2004

TENTATIVE

OBJECTIVES:

- To develop Python programs with conditionals and loops.
- To define Python functions and use function calls.
- To use Python data structures – lists, tuples, dictionaries.
- To do input/output with files in Python.

UNIT I PYTHON BASICS 10

Introduction to Python Programming – Python Interpreter and InteractiveMode– Variables and Identifiers – Arithmetic Operators – Values and Types – Statements.Operators – Boolean Values – Operator Precedence – Expression – Conditionals: If-ElseConstructs – Loop Structures/Iterative Statements – While Loop – For Loop – BreakStatement-Continue statement – Function Call and Returning Values – Parameter Passing – Local and GlobalScope – Recursive Functions.

UNIT II DATA TYPES IN PYTHON 10

Lists, Tuples, Sets, Strings, Dictionary, Modules: Module Loading and Execution – Packages – Making Your OwnModule – The Python Standard Libraries

UNIT III FILE HANDLING AND EXCEPTION HANDLING 8

Files: Introduction – File Path – Opening and Closing Files – Reading and Writing Files –File Position – Exception: Errors and Exceptions, Exception Handling, Multiple Exceptions

UNIT IV MODULES, PACKAGES 9

Modules: Introduction – Module Loading and Execution – Packages – Making Your OwnModule – The Python Libraries for data processing, data mining and visualization- NUMPY, Pandas, Matplotlib, Plotly

UNIT V OBJECT ORIENTED PROGRAMMING IN PYTHON 8

Creating a Class, Class methods, Class Inheritance, Encapsulation, Polymorphism, class method vs. static methods, Python object persistence

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, students will be able to

- Develop algorithmic solutions to simple computational problems
- Structure simple Python programs for solving problems.
- Read and write data from/to files in Python Programs.
- Represent compound data using Python lists, tuples, dictionaries.
- Decompose a Python program into functions.

REFERENCES:

1. ReemaThareja, "Python Programming using Problem Solving Approach", Oxford University Press, First edition, 2017
2. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", SecondEdition, Shroff,O'Reilly Publishers, 2016(<http://greenteapress.com/wp/thinkpython/>)
3. Guido van Rossum, Fred L. Drake Jr., "An Introduction to Python – Revised andUpdatedforPython 3.2, Network Theory Ltd., First edition, 2011
4. John V Guttag, "Introduction to Computation and Programming Using Python",Revised and Expanded Edition,MIT Press, 2013
5. Charles Dierbach, "Introduction to Computer Science using Python", Wiley IndiaEdition, First Edition, 2016
6. Timothy A. Budd, "Exploring Python", Mc-Graw Hill Education (India) Private Ltd.,First edition,2011
7. Kenneth A. Lambert, "Fundamentals of Python: First Programs", Cengage Learning,second edition,2012

TENTATIVE

RESEARCH METHODOLOGY AND INTELLECTUAL PROPERTY RIGHTS

L T P C
2 0 0 2

COURSE OBJECTIVES:

The course should enable the students to:

- Identify an appropriate research problem in their interesting domain.
- Understand ethical issues; understand the Preparation of a research project thesis report.
- Understand the Preparation of a research project thesis report
- Understand the law of patent and copyrights.
- Acquire adequate knowledge of IPR.

UNIT I RESEARCH METHODOLOGY 6

Research Methodology – An Introduction, Objectives, Types of research, Research approaches, Significance, Research methods versus Methodology, Research and Scientific method, Importance, Research process, Criteria, Problems encountered by researchers.

Defining the research problem – Research problem, Selecting the problem, Necessity, Technique involved, An illustration.

Reviewing the Literature – The place of the literature review in research, How to review the Literature, Writing about the literature reviewed.

UNIT II RESEARCH DESIGN 6

Research Design – Meaning, Need, Features, Different research design, Basic principles of experimental designs, Important experimental designs.

Measurement & Scaling techniques – Sampling Design, Measurement in research, Measurement scales, Error, Measurement tools, Scaling, Meaning, Scale classification, Scale construction techniques

Data Collection – Collection of primary data, Collection of secondary data, Selection of appropriate method for data collection.

UNIT III RESEARCH TECHNIQUE AND TOOLS 6

Testing of Hypothesis – Basic concepts, Procedure, Test of Hypothesis, Important parametric Tests, Hypothesis Testing unifications.

Interpretation & Report writing – Meaning, techniques, Precaution in Interpretation, Significance of Report writing, steps, Layout, types, mechanics, precautions.

Use of Tools/ Techniques for research – Use of Encyclopedias, Research Guides, Handbook etc., Academic Databases for Computer Science Discipline, Use of tools / techniques for Research methods to search required information effectively, Reference Management Software like Zotero/Mendeley, Software for paper formatting like LaTeX/MS Office, Software for detection of Plagiarism

UNIT IV INTELLECTUAL PROPERTY RIGHTS 6

Intellectual Property – The concept, IPS in India, development, Trade secrets, utility Models, IPR & Bio diversity, CBD, WIPO, WTO, Right of Property, Common rules, PCT, Features of Agreement, Trademark, UNESCO.

UNIT V PATENTS 6

Patents – Learning objectives, Concept, features, Novelty, Inventive step, Specification, Types of patent application, E-filing, Examination, Grant of patent, Revocation, Equitable Assignments, Licences, Licencing of related patents, patent agents, Registration of patent agents.

REFERENCE BOOKS:

1. Research Methodology: Methods and Techniques by C.R.Kothari, GauravGarg, New Age International 4th Edition 2018 (UNIT I to UNIT III)
2. Research Methodology a step-by-step guide for beginners by Ranjit Kumar, SAGE publications Ltd 3rd Edition 2011 (For the topic Reviewing the Literature under Unit I)
3. Stuart Melville and Wayne Goddard, "Research Methodology: An Introduction for Science & engineering students.Juta and Co., Limited, 1996, First edition
4. Research methods: The concise knowledge base-Trochim, Atomic Dog publishing, First edition,2005..
5. John W. Best & James V. Khan, "Research in Education", Pearson 8thEdition'year.
6. Professional Programme Intellectual Property Rights, Law and practice, The Institute of Company Secretaries of India, Statutory body under an Act of parliament, September 2013 (UNIT IV & UNIT V)

OUTCOMES: (Cos)

On completion of the course the student would be able to :

CO1: Understand the research problem and Literature review.

CO2: Understand the various research designs and their characteristics.

CO3: Prepare a well-structured research paper and scientific presentations.

CO4: Explore on various IPR Components and process of filing.

CO5 Develop awareness the patent law and procedural mechanism in obtaining a patent.

TENTATIVE

OBJECTIVES:

The student should be able:

- To understand the concepts of Open Source DBMS.
- To understand the process of distributing tables across multiple systems
- To understand the process of storing, retrieving spatial and temporal data
- To understand the process of storing, retrieving objects in a database
- To understand the process of storing and retrieving data from a XML Database
- To use the open source database for building a mobile application

EXPERIMENTS IN THE FOLLOWING TOPICS:

1. NOSQL Exercises
 - a. MongoDB – CRUD operations, Indexing, Sharding, Deployment
 - b. Cassandra: Table Operations, CRUD Operations, CQL Types
 - c. HIVE: Data types, Database Operations, Partitioning – HiveQL
 - d. OrientDB Graph database – OrientDB Features
2. MySQL Database Creation, Table Creation, Query
3. MySQL Replication – Distributed Databases
4. Spatial data storage and retrieval in MySQL
5. Temporal data storage and retrieval in MySQL
6. Object storage and retrieval in MySQL
7. XML Databases , XML table creation, XQuery FLWOR expression
8. Mobile Database Query Processing using open source DB (MongoDB/MySQL etc)

TOTAL: 60 PERIODS

OUTCOMES:

Upon completion of this course, the student should be able to:

- Design and Implement databases.
- Formulate complex queries using SQL
- Design and Implement applications that have GUI and access databases for backend connectivity
- To design and implement Mobile Databases
- To design and implement databases to store spatial and temporal data objects

ADVANCED DATA STRUCTURES AND PYTHON PROGRAMMING LABORATORY

L T P C
0 0 4 2

OBJECTIVES:

- To learn the basic programming constructs in Python.
- To implement Recursive programming in Python
- To implement Divide and Conquer algorithmic technique in Python
- To implement Tree Data structures in Python
- To implement Graphs in Python
- To deploy the standard libraries in Python

EXPERIMENTS:

1. Towers of Hanoi using Recursion
2. To implement Binary Search
3. Merge Sort
4. To implement AVL Trees using Python
5. To implement Splay Trees using Python
6. To implement Red black Trees using Python
7. To implement Graphs using Python
8. Implementing programs using written modules and Python Standard Libraries.
9. Implementing real-time/technical applications using Files and Exception handling.

TOTAL: 60 PERIODS

OUTCOMES:

On completion of the course, students will be able to:

1. Develop algorithmic solutions to simple computational problems
2. Develop and execute Python programs.
3. Decompose a Python program into functions.
4. Represent compound data using Python data structures.
5. Apply Python features in developing software applications.

COMMUNICATION SKILLS - I

L T P C
0 0 2 1

OBJECTIVES:

- To provide opportunities to learners to practice English and thereby make them proficient users of the language.
- To enable learners to fine-tune their linguistic skills (LSRW) with the help of Technology.
- To enhance the performance of students listening, speaking, reading and writing and thereby develop their career opportunities.

LIST OF ACTIVITIES:

1. Listening:
 - Listening and practicing neutral accents
 - Listening to short talks and lectures and completing listening comprehension exercises
 - Listening to TED Talks
2. Speaking:
 - Giving one minute talks
 - Participating in small Group Discussions
 - Making Presentations
3. Reading:
 - Reading Comprehension
 - Reading subject specific material
 - Technical Vocabulary
4. Writing:
 - Formal vs Informal Writing
 - Paragraph Writing
 - Essay Writing
 - Email Writing

TOTAL: 30 PERIODS

REFERENCES / MANUALS / SOFTWARE: Open Sources / websites

OUTCOMES:

On completion of the course the students will be able to:

- Listen and comprehend Lectures in English
- Articulate well and give presentations clearly
- Participate in Group Discussions successfully
- Communicate effectively in formal and informal writing
- Write proficient essays and emails