GURU KASHI UNIVERSITY

Master of Computer Applications

Session: 2025-2026

Faculty of Computing

Graduate Attributes of the Programme: -

Type of learning outcomes	**The Learning Outcomes Descriptors**
Graduates should be able to demonstrate the acquisition of:	
Learning outcomes that are specific to disciplinary/interdisciplinary areas of learning	Apply advanced knowledge in programming, software development, operating systems, computer graphics, AI, and cloud computing.
	Design and manage complex systems using data structures, algorithms, database technologies, and web/mobile development platforms.
	Integrate interdisciplinary technologies such as IoT, data mining, machine learning, and big data analytics to solve real-world problems.
	Conduct research using scientific methodologies and contribute to innovative computing solutions for business and societal needs.
Generic learning outcomes	Exhibit critical thinking, problem-solving, and decision-making skills in multidisciplinary environments.
	Demonstrate ethical responsibility, teamwork, adaptability, and commitment to lifelong learning.
	Apply leadership and entrepreneurial skills to create IT-based solutions with societal relevance.

Programme Learning outcomes: Post Graduate Diploma in Computer Application (PGDCA) Certificate is awarded to students who have demonstrated the achievement of the outcomes located at level 4.5:

Element of the Descriptor	**Programme learning outcomes relating to PGDCA**
The graduates should be able to demonstrate the acquisition of:	
Knowledge and understanding	Acquire in-depth knowledge of data structures, algorithms, operating systems, and web technologies.
	Understand software project management principles and the Indian knowledge system.
	Comprehend theoretical and practical aspects of Python programming and advanced graphics.
	Demonstrate foundational understanding of object-oriented programming, advanced database systems, and computer architecture.
General, technical and professional skills required to perform and accomplish tasks	Create and manage databases and applications using advanced DBMS concepts.
	Develop and debug software using C++, Python, and web development tools.
	Implement algorithms and data structures efficiently.
Application of knowledge and skills	Apply programming, analytical, and management skills to solve real-world computing problems. Build and deploy web-based and desktop applications.
Generic learning outcomes	Develop effective communication, documentation, and collaboration skills through lab work and project-based learning. Exhibit adaptability and problem-solving capabilities in varied technical environments.
Constitutional, humanistic, ethical, and moral values	Recognize the importance of professional ethics and moral responsibilities in software development. Demonstrate sensitivity to societal issues through technology use aligned with Indian knowledge and value systems.
Employability and job-ready skills, and entrepreneurship skills and capabilities/qualities and mindset	Gain practical experience with tools and languages used in the software industry (C++, Python, web technologies). Develop a mindset geared toward innovation, entrepreneurship, and continuous learning. Be prepared for roles such as software developer, web programmer, or system analyst.
Credit requirements	A student will be allowed an exit option after passing

	first academic year of the MCA Programme with requisite 46 credits.
Entry requirements	Passed any graduation degree (B.E./B.Tech./B.Sc./B.Com./BA/B/Voc/BCA etc.) preferably with Mathematics at 10+2 level or Graduation level. Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying examination. (for students having no Mathematics background, a compulsory bridge course will be required/additional bridge course related to computer subjects required as applicable.)

Programme Structure of Master of Computer Application

Semester 1st									
Course Code	**Course Title**	**Type of Course**	**L**	**T**	**P**	**Credits**	**Int**	**Ext**	**Total Marks**
MCA1400	Object Oriented Programming using C++	Core	3	0	0	3	30	70	100
MCA1401	Advanced Database Management System	Core	3	0	0	3	30	70	100
MCA1402	Object Oriented Programming using C++ Lab	Core	0	0	2	1	30	70	100
MCA1403	Advanced Database Management System Lab	Core	0	0	2	1	30	70	100
MCA1404	Advanced Computer Architecture	Core	4	0	0	4	30	70	100
MCA1405	Software Project Management	Core	4	0	0	4	30	70	100
IKS0016	Indian Knowledge System	Indian Knowledge System	4	0	0	4	30	70	100
Discipline Elective I (Any one of the following)									
MCA1406	Cloud Computing	Discipline Elective I	4	0	0	4	30	70	100
MCA1407	Big Data								
MCA1408	Machine Learning								
Total			**22**	**0**	**4**	**24**	**240**	**560**	**800**

Semester 2nd									
Course Code	**Course Title**	**Type of Course**	**L**	**T**	**P**	**Credits**	**Int**	**Ext**	**Total Marks**
MCA2450	Advanced Data Structures & Algorithms	Core	3	0	0	3	30	70	100
MCA2451	Programming using Python	Core	3	0	0	3	30	70	100
MCA2452	Advanced Data Structure & Algorithms Lab	Core	0	0	2	1	30	70	100
MCA2453	Programming using Python Lab	Core	0	0	2	1	30	70	100
MCA2454	Advanced Operating System	Core	4	0	0	4	30	70	100
NCA2455	Advanced Computer Graphics	Core	3	0	0	3	30	70	100
MCA2456	Advanced Computer Graphics Lab	Core	0	0	2	1	30	70	100
MCA2457	Advance Web Technologies	Employability and Entrepreneurship Skills	0	0	4	2	30	70	100
Discipline Elective II (Any one of the following)									
MCA2458	IoT and Its Applications	Discipline Elective II	4	0	0	4	30	70	100
MCA2459	Digital Image Processing								
MCA2460	Data Warehousing and Data Mining Techniques								
Total			**17**	**0**	**10**	**22**	**240**	**560**	**800**

Programme Learning outcomes: Master of Computer Application degree is awarded to students who have demonstrated the achievement of the outcomes located at level 4.5:

Element of the Descriptor	**Programme learning outcomes relating to MCA**
The graduates should be able to demonstrate the acquisition of:	
Knowledge and understanding	Understand research methodology and software engineering practices to manage and execute complex projects.
	Demonstrate advanced knowledge in programming paradigms, data structures, algorithms, database management, computer architecture, operating systems, and emerging technologies such as AI, IoT, and mobile computing.
	Acquire deep insight into theoretical foundations of computer science such as Theory of Computation and Software Quality Assurance.
	Explore current trends in information systems and their applications in industry and academia.
General, technical and professional skills required to perform and accomplish tasks	Design, develop, and deploy advanced software systems using modern tools, platforms, and frameworks.
	Conduct technical research and apply statistical and analytical methods to solve real-world IT problems.
	Perform system analysis, manage projects, and apply professional documentation and quality assurance standards.
Application of knowledge and skills	Solve complex computing problems by integrating theoretical knowledge with practical implementation. Apply machine learning and data analytics to drive decision-making and business intelligence.
Generic learning outcomes	Exhibit leadership, collaboration, and problem-solving in multi-disciplinary teams. Engage in self-directed learning and stay updated with evolving technologies.
Constitutional, humanistic, ethical, and moral values	Practice ethical computing, data privacy, and responsible use of information technology. Promote inclusivity, diversity, and sustainability in digital solutions.
Employability and job-ready skills, and entrepreneurship	Be equipped for roles such as software engineer, data analyst, AI/ML developer, cybersecurity specialist, network administrator, and cloud engineer.

skills and capabilities/qualities and mindset	Possess strong entrepreneurial and innovation-driven mindset supported by project and internship experience.
Credit requirements	90 Credits
Entry requirements	Bachelor Degree with Honours/ Honours with Research in Information Technology or Computer Applications as a major course or BCA with Honors/Honours with Research (4 years) OR 1 year PG Diploma in Computer Applications/Information Technology with at least 45% marks or Equivalent CGPA in aggregate, after 3 years Bachelor Degree.

Semester 3rd									
Course Code	**Course Title**	**Type of Course**	**L**	**T**	**P**	**Credits**	**Int**	**Ext**	**Total Marks**
MCA3500	Artificial Intelligence	Core	4	0	0	4	30	70	100
MCA3501	Research Methodology	Core	4	0	0	4	30	70	100
MCA3502	Network Administration	Employability and Entrepreneurship Skills	0	0	4	2	30	70	100
MCA3503	Dissertation (Phase I)	Research Skills	0	0	0	12	30	70	100
Total			**8**	**0**	**4**	**22**	**120**	**280**	**400**

Semester 4th									
Course Code	**Course Title**	**Type of Course**	**L**	**T**	**P**	**Credits**	**Int**	**Ext**	**Total Marks**
MCA4550	Java Programming	Core	3	0	0	3	30	70	100
MCA4551	Java Programming Lab	Core	0	0	2	1	30	70	100
MCA4552	Mobile Application Development	Employability and Entrepreneurship Skills	0	0	4	2	30	70	100
MCA4553	Dissertation (Phase II)	Research Skills	0	0	0	12	30	70	100
Discipline Elective III (Any one of the following)									
MCA4554	Theory of Computation	Discipline Elective III	4	0	0	4	30	70	100
MCA4555	Principle & Practices of Management								
MCA4556	Software Testing and Quality Assurance								
Total			**7**	**0**	**6**	**22**	**150**	**350**	**500**
Grand Total			**54**	**0**	**24**	**90**			

Semester-I

Course Title: Object Oriented Programming using C++	L	T	P	Cr.
Course Code: MCA1400	3	0	0	3

Total Hours : 45

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Describe all the basic concepts of C++ and its features such as composition of objects, Operator overloading.
2. Analyze inheritance with the understanding of early binding and late binding.
3. Classify various object-oriented concepts to solve different problems.
4. Analyze and explore various Stream classes, I/O operations and exception handling.

Course Content

UNIT I **15 Hours**

Programming Basics: Introduction to Programming, Programming Paradigms, Programming Languages and Types. Introduction to C - Basic Program Structure, Execution flow of C Program, Directives, Basic Input /Output Introduction to Object Oriented Programming- OOP concepts, Advantages, Applications, Comparison of C and C++-Data Types, Control Structures, Operators and Expressions.

Introduction to C++: Structure of a C++ program, Classes and Objects, Access modifiers, Data Members, Member Functions, Inline Functions, passing parameters to a Function (pass by Value, pass by Address, pass by Reference), Function Overloading, Object as a Parameter, Returning Object Static data members and functions, Constant Data members and functions

Constructors- Default, Parameterized, Copy, Constructor Overloading, Destructors Arrays, Array as a Class Member, Array of Objects, Strings C style strings and String Class.

UNIT II **10 Hours**

Operator Overloading and Pointers: Operator Functions-Member and Non Member Functions, Friend Functions Overloading Unary operators Overloading binary operators(Arithmetic, Relational, Arithmetic Assignment, equality), Overloading Subscript operator Type Conversion Operators- primitive to Object, Object to primitive, Object to Object Disadvantages of operator Overloading, Explicit and Mutable Pointers, Pointer and Address of Operator, Pointer to an Array and Array of Pointers, Pointer arithmetic, Pointer to a Constant and Constant Pointer, Pointer Initialization, Types of Pointers(void, null and dangling), Dynamic Memory Allocation, Advantages and Applications of pointers

UNIT III **10 Hours**

Inheritance and Polymorphism: Inheritance Concept, protected modifier, Derivation of Inheritance- Public, Private and Protected,
Types of Inheritance-Simple, Multilevel, Hierarchical, Multiple, Hybrid, Constructors and Inheritance, Function Overriding and Member hiding Multiple Inheritance, Multipath inheritance – Ambiguities and solutions Polymorphism, Static and Dynamic Binding, Virtual Functions, Pure Virtual Functions, Virtual destructors, Abstract Classes, Interfaces.

UNIT IV **10 Hours**

Memory management: new delete, object copying copy constructors, assignment operator, this input/output. Streams and Exceptions: Files, Text and Binary Files, Stream Classes, File IO using Stream classes, File pointers, Error Streams, Random File Access, Manipulators, Overloading Insertion and extraction operators Error handling, Exceptions, Throwing and catching exceptions, Custom Exceptions, Built in exceptions
Advanced C++: Templates and standard Template library: template classes, declaration, template functions, containers, algorithms, iterators, manipulating string objects, hashes, iostreams and other type.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Kamthane, A. (2012). Programming in C++, 2/e. Pearson Education India.*
- *Salaria, R. S. (2016). Mastering Object-Oriented Programming with C++. KHANNA PUBLISHING HOUSE.*
- *Balagurusamy, E. (2001). Object-Oriented Programming with C++, 7e. McGraw- Hill Education.*

Course Title: Advanced Database Management System	L	T	P	Cr.
Course Code: MCA1401	3	0	0	3

Total Hours : 45

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Understand advanced database concepts and architectures.
2. Apply normalization techniques and database design principles.
3. Implement and manage complex queries and transactions.
4. Analyze and utilize advanced data models and indexing techniques

Course Content

UNIT I **15 Hours**

Database System Architectures: Centralized, Client-Server, Distributed, Parallel Databases. Data Models: Hierarchical, Network, Object-Oriented, and Object-Relational Models. Introduction to Big Data and NoSQL Databases: Characteristics, Types, and Use Cases.

UNIT II **10 Hours**

Complex SQL Queries: Sub queries, Joins, Set Operations, and Aggregations. Stored Procedures and Functions: Creation, Execution, and Management. Triggers and Cursors: Definition, Types, and Applications. Transaction Management: ACID Properties, Transaction Isolation Levels, Concurrency Control.

UNIT III **10 Hours**

Advanced ER Modeling: Extended ER Models, Subclasses, Super classes, and Inheritance. Normalization: Functional Dependencies, Multivalued Dependencies, Higher Normal Forms (BCNF, 4NF, 5NF). Demoralization Concepts, Techniques, and Trade-offs. Database Tuning: Indexing, Query Optimization, and Performance Tuning

UNIT IV **10 Hours**

Data Warehousing Architecture, ETL Processes, Star and Snowflake Schemas. Data Mining: Techniques, Classification, Clustering, Association Rules. Distributed Databases: Fragmentation, Replication, Distributed Query Processing. Security and Authorization: Database Security Models, SQL Injection, Encryption Techniques. Transaction Management and Concurrency Control, Backup, Recovery, and Cloud Databases

Transactional modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Elmasri, R., & Navathe, S.B. (2015). Fundamentals of Database Systems. Pearson.*
- *Silberschatz, A., Korth, H.F., & Sudarshan, S. (2010). Database System Concepts. McGraw-Hill Education.*
- *Connolly, T., & Begg, C. (2014). Database Systems: A Practical Approach to Design, Implementation, and Management. Pearson.*
- *Ramakrishnan, R., & Gehrke, J. (2002). Database Management Systems. McGraw-Hill Education.*

Course Title: Title: Object Oriented Programming using C++ Lab	L	T	P	Cr.
Course Code: MCA1402	0	0	2	1

Total Marks:30

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Design an algorithmic solution for a given problem.
2. Debug a given Program.
3. Identify solutions to a problem, apply control structures, and use defined functions for solving the problem.
4. Implement Programs with pointers and arrays, per form pointer arithmetic, and use the pre-processor.

List of Experiments:

1. Write a Program to display Names, Roll No., and grades of 3 students who have appeared in the examination. Declare the class of name, Roll No. and grade. Create an array of class objects. Read and display the contents of the array.
2. Write a Program to swap two Characters of different data types using function overloading.
3. Write a program to demonstrate the use of inline, friend functions and this keyword.
4. Write a program to implement static data members and member functions.
5. Write a Program to implement Constructor and Destructor.
6. Write a Program to demonstrate Constructor Overloading.
7. Write a Program to calculate factorial using Copy Constructor.
8. Write a Program to allocate & deallocate memory using new [ ] and delete [ ].
9. Write a Program to demonstrate the use of function overloading.
10. Write a Program to overload comparison operator operator== and operator! = .
11. Write a Program to create an array of pointers.
12. Create a base class containing the data member roll number and name. Also create a member function to read and display the data using the concept of single level inheritance. Create a derived class that contains marks of two subjects and Total Marks as the data members.
13. Write a Program to create multilevel inheritance. (Hint: Classes A1, A2, A3)
14. Write a program to demonstrate the concept of function overriding.
15. Write a Program to demonstrate the use of virtual functions and polymorphism.
16. Write a Program to demonstrate the use of pure virtual functions.
17. Write a Program to demonstrate the concepts of abstract class.
18. Write a Program to perform exception handling.
19. Write a Program to copy the contents of one file to another file.
20. Write a Program to create Generic Functions using Template.

21. Write a Program to Use try-catch blocks to handle built-in and user-defined exceptions.
22. Write a Program to Perform file operations such as read/write using streams (ifstream, ofstream, fstream).

Course Title: Advanced Database Management System lab	L	T	P	Cr.
Course Code: MCA1403	0	0	2	1

Total Hours :30

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Develops an Entity-Relationship model based on user requirements.
2. Implements the role of the database administrator and his responsibilities.
3. Apply Normalization techniques to normalize a database.
4. Declares and enforces integrity constraints on a database.

List of Experiments:

1. Design a database schema for a university management system, including tables for students, courses, instructors, and enrollments, ensuring normalization and integrity constraints.
2. Write an SQL query to retrieve the names of students who have scored above the average marks in a particular subject.
3. Create a stored procedure to calculate the total salary of employees in a department and update the department's budget accordingly.
4. Develop a trigger that automatically updates a log table whenever a new record is inserted into the main table, capturing details like timestamp and user ID.
5. Write PL/SQL code to transfer funds from one bank account to another, ensuring that the transaction is atomic and consistent.
6. Experiment with creating indexes on different columns of a large table and compare query performance using execution plans.
7. Identify and optimize a slow-performing query by rewriting it or adding appropriate indexes to improve execution time.
8. Design a star schema for a retail sales data warehouse, including dimensions like time, product, and customer, and a fact table for sales transactions.
9. Implement an OLAP cube to analyze sales data, allowing users to slice and dice sales metrics by product category, region, and time period.
10. Implement the Priority algorithm to discover frequent item sets from a transaction database and generate association rules.
11. Store and retrieve XML documents in an Oracle database, demonstrating XPath queries to extract specific information from XML data.
12. Implement a locking mechanism to prevent two users from booking the same hotel room simultaneously in a hotel reservation system
13. Develop small-scale database applications that integrate advanced DBMS concepts.

Course Title: Advanced Computer Architecture	L	T	P	Cr.
Course Code: MCA1404	4	0	0	4

Total Hours :60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Analyze the design and functioning of modern computer systems, including instruction set architectures and microarchitectural techniques.
2. Evaluate performance metrics and optimization strategies for parallel processing and multicore systems.
3. Develop an understanding of memory hierarchy and its impact on system performance.
4. Explore advanced topics such as pipelining, superscalar execution, and branch prediction.
5. Apply knowledge of interconnection networks and distributed architectures to solve computational challenges.

Course Content

UNIT- I **15 Hours**

PARALLEL COMPUTER MODELS: Evolution of Computer architecture, system attributes to performance, Multi processors and multi computers, Multi-vector and SIMD computers, PRAM and VLSI models-Parallelism in Programming, conditions for Parallelism-Program Partitioning and Scheduling-program flow Mechanisms-Speed up performance laws-Amdahl‘s law, Gustafson‘s law-Memory bounded speedup Model.

UNIT- II **15 Hours**

MEMORY SYSTEMS AND BUSES: Memory hierarchy-cache and shared memory concepts-Cache memory organization-cache addressing models, Aliasing problem in cache, cache memory mapping techniques-Shared memory organization-Interleaved memory organization, Lower order interleaving, Higher order interleaving. Backplane bus systems-Bus addressing, arbitration and transaction.

UNIT -III **15 Hours**

ADVANCED PROCESSORS Instruction set architectures-CISC and RISC scalar processors-Super scalar processors-VLIW architecture- Multivector and SIMD computers-Vector processing principles-Cray Y-MP 816 system-Inter processor communication

UNIT- IV **15 Hours**

MULTI PROCESSOR AND MULTI COMPUTERS: Multiprocessor system interconnects- Cross bar switch, Multiport Memory-Hot spot problem, Message passing mechanisms-Pipelined Processors-Linear pipeline, on linear pipeline- Instruction pipeline design-Arithmetic pipeline design. Data flow computer architectures-Static, Dynamic-VLSI Computing Structures-Systolic

array architecture, mapping algorithms into systolic arrays, Reconfigurable processor array-VLSI matrix arithmetic processors-VLSI arithmetic models, partitioned matrix algorithms, matrix arithmetic pipelines. Synchronization and memory consistency in parallel systems. I/O devices and data transfer techniques. Direct Memory Access (DMA) and Interrupt handling. System interconnects and bus architectures.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Parallel Computer Architecture: A Hardware/ Software Approach – David Culler & Jaswinder Pal Singh*
- *Computer Architecture: A Quantitative Approach – John L. Hennessy and David A. Patterson*
- *Parallel Processing and Parallel Algorithms – Selim G. Akl*
- *Advanced Computer Architecture – Kai Hwang*

Course Title: Software Project Management	L	T	P	Cr.
Course Code: MCA1405	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Identify the different project contexts and suggest an appropriate project management strategy.
2. Practice the role of project planning, risks associated in successful software development.
3. Identify and describe the key phases of project monitoring and contracts in management.
4. Learn to apply the concept of project management and planning on organizing a team and people's behavior.

Course Content

UNIT I **15 Hours**

Introduction to Software Project Management: Project Definition, Contract Management, Activities Covered by Software Project Management, Overview of Project Planning, plan methods, methodology.

Project Evaluation: Strategic Assessment, Technical Assessment, Cost Benefit Analysis, Cash Flow Forecasting, Cost Benefit Evaluation Techniques, Risk Evaluation, selection of project approach: discussion on models, choice of process models.

UNIT II **15 Hours**

Activity Planning: Objectives, Project Schedule, Sequencing and Scheduling Activities, Network Planning Models, Forward Pass, Backward Pass, Activity Float, Shortening Project Duration, Activity on Arrow Networks,

Risk Management: Nature of Risk, Types of Risk, Managing Risk, Hazard Identification, Hazard Analysis, Risk Planning and Control.

UNIT III **15 Hours**

Monitoring and Control: Creating Framework, Collecting the Data, Visualizing Progress, Cost Monitoring, Earned Value analysis, Prioritizing Monitoring, Getting Project Back to Target, and Change Control.

Managing Contracts: Introduction, Types of Contract, Stages in Contract Placement, Typical Terms of a Contract, Contract Management, Acceptance.

Resource allocation:

introduction and nature of resources, identification of resource requirements, scheduling, creating critical path, cost schedule, counting cost.

UNIT IV **15 Hours**

Effort estimation: basics of software estimation, techniques, COCOMO-II, cost, staffing pattern.

Managing People and Organizing Teams: Introduction, Understanding Behavior, Organizational Behavior: Background, Selecting The Right Person For The Job, Instruction In The Best Methods, Motivation , The Oldman, Hackman Job Characteristics Model, Working In Groups, Becoming A Team, Decision Making, Leadership, Organizational Structures, Stress, Health And Safety

Stakeholder management: Clients, vendors, team members, and users.

Transactional modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Bob Hughes, Mike Cotterell, Software Project Management, Tata McGraw Hill Publishing*
- *Ramesh, Gopalaswamy, Managing Global Projects, Tata McGraw Hill Publishing*
- *Royce, Software Project Management, Pearson Education Publishing*
- *Jalote, Software Project Management in Practice, Pearson Education Publishing*

Course Title: Indian Knowledge System	L	T	P	Cr.
Course Code: IKS0016	4	0	0	4

Total Hours :60

Course Learning Outcome:

1. It aimed to instill in students a sense of rootedness and pride in India, along with an appreciation for its rich, diverse, ancient, and modern culture, knowledge systems, and traditions.
2. This course helps students understand the rich scientific and technological heritage of the country.
3. This course emphasizes and promotes the Indian concept of multidisciplinary learning systems, integrating them with modern science.
4. The course will underscore the importance of intellectual property rights in safeguarding Indian knowledge.

Course Content

Unit-I **15 Hours**

Ancient philosophy of Knowledge: Vedas, Vedangas, sutras – Gurukul parampara

Unit-II **15 Hours**

Astronomy in India: The Beginnings of Indian Astronomy - The Early Historical Period-The Siddhāntic Era - The Kerala School - Aryabhatta - Varahamihira- Bhaskara I – Brahamagupta- Bhaskara II – Brief notes on Astronomical instruments

Unit-III **15 Hours**

Chemistry in India: Early Chemical Techniques, Atomism in Vaiśeṣika - Rishi Kanad- Nagarjuna- Al-Bīrūnī', Vāgbhaṭa- Sushruta- Carak Metallurgy in India - Definition, Metallurgy in Harappan Civilization, Metallurgy of Gold- Copper- Zinc- Bronze - Iron and steel.

Unit-IV **15 Hours**

Developments in Mathematics: Number systems- Geometry- works of Pingala- Baudhayana- Jaina Mahavira-Sridharacharya – Madhava Siddanthas and Calender systems

Transactional Mode

Seminars, Group discussion, Team teaching, Focused group discussion, Assignments, Project-based learning, Simulations, reflection and Self-assessment

Suggested Readings

1. *A Concise History of Science in India – Bose, Sen & Subbarayappa- INSA Publications*
2. *Encyclopedia of Classical Indian Sciences- Roddam Narasimha, Universities Press*
3. *NCERT Modules of KPTI*

Course Title: Cloud Computing	L	T	P	Cr.
Course Code: MCA1406	4	0	0	4

Total Hours :60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Recognize the fundamentals and essentials of Cloud Computing.
2. Describe the Concept of Cloud Infrastructure Model.
3. Analyze the key technical and organizational challenges.
4. Interpret the importance of virtualization in distributed computing.

Course Contents

UNIT I **15 Hours**

Introduction to Cloud Computing: Introduction to Cloud Computing, History and Evolution of Cloud Computing, Types of Cloud Computing, Cloud Computing Architecture, Basics of Cloud Infrastructure.

UNIT II **15 Hours**

Cloud Computing Delivery Models: Introduction, Cloud Computing Delivery Models, Attributes of Cloud Computing, Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), Comparison of Different Services, Combining Different Services, Obstacles for Cloud Technology, Cloud Vulnerabilities, Cloud Challenges, Practical Applications of Cloud Computing.

Migrating to the Cloud: Introduction, Broad Approaches to Migrating to the Cloud, The Seven-step Model of Migration to the Cloud, Service Level Agreements (SLA). Evaluating the Business Need, Cloud vs. Hosted Applications, Cloud vs. Licensed Software Vendors.

UNIT III **15 Hours**

Selection of Cloud Provider: Introduction, A Brief about Leading Cloud Service Providers, Considerations for Selecting a Cloud Solution, Business Considerations, Data Safety and Security, Interoperability, Portability and Integration, Geographical Considerations, Contingency and Recovery Management, Ethical and Legal Considerations, Scalability and Flexibility.

UNIT IV **15 Hours**

Abstraction and Virtualization: Introduction to Virtualization Technologies, Understanding Hypervisors, Scheduling and Load Balancing.

Securing the Cloud: Securing the Cloud, Securing Data, Establishing Identity and Presence.

Case-Studies: Using Google Web Services, Using Amazon Web Services, Using Microsoft Cloud Services. Emerging Trends and Best Practices: Edge and Fog Computing, Serverless Computing and Function as a Service (FaaS), Cloud Automation and Orchestration, Best Practices in Cloud Deployment and Management, Future Directions in Cloud Computing

Transactional modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings:

- *Buyya, R., Broberg, J., &Goscinski, A. M. (Eds.). (2010). Cloud computing: Principles and paradigms. John Wiley & Sons.*
- *Sosinsky, B. (2010). Cloud computing bible. John Wiley & Sons.*
- *Miller, M. (2008). Cloud computing: Web-based applications that change the way you work and collaborate online. Que publishing.*
- *Kiswani, J. H., Dascalu, S. M., & Harris Jr, F. C. (2021). Cloud computing and its applications: A comprehensive survey. International Journal of Computer Applications IJCA, 28.*

Total Hours: 60

Course Title: Big Data	L	T	P	Credits
Course Code: MCA1407	4	0	0	4

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Discuss the building blocks of Big Data.
2. Articulate the programming aspects of cloud computing (map Reduce etc.).
3. Knowledge about the recent research trends related to Hadoop File System, Map Reduce and Google File System etc.
4. Study different types Case studies on the current research and applications of the Hadoop and big data in industry.

Course Content

UNIT I **15 hours**

Introduction to Big Data: Introduction to Big Data Platform – Challenges of Conventional Systems - Intelligent data analysis – Nature of Data - Analytic Processes and Tools - Analysis Vs Reporting - Modern Data Analytic Tools - Statistical Concepts: Sampling Distributions - Re-Sampling - Statistical Inference - Prediction Error.

UNIT II **15 hours**

Mining Data Streams: Introduction to Streams Concepts – Stream Data Model and Architecture - Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness in a Window – Decaying Window - Real Time Analytics Platform(RTAP)Applications –- Case Studies - Real Time Sentiment Analysis, Stock Market Predictions.

UNIT III **15 hours**

Hadoop Environment: History of Hadoop- The Hadoop Distributed File System – Components of Hadoop- Analyzing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design of HDFS-Hadoop file systems- Java interfaces to HDFS- Basics- Developing a Map Reduce Application-How Map Reduce Works-Anatomy of a Map Reduce Job run-Failures-Job Scheduling- Shuffle and Sort – Task execution - Map Reduce Types and Formats- Map Reduce Features - Setting up a Hadoop Cluster - Cluster specification - Cluster Setup and Installation – Hadoop Configuration-Security in Hadoop.

UNIT IV **15 hours**

Data Analysis Systems and Visualization: Link Analysis – Pagerank - Efficient Computation of Pagerank- Topic-Sensitive Page Rank – Link Spam-

Recommendation Systems- A Model for Recommendation Systems- Content-Based Recommendations - Collaborative Filtering- Dimensionality Reduction- Visualizations - Visual data analysis techniques-interaction techniques- Systems and applications.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Chris Eaton, (2012). Dirk deRoos et al., Understanding Big data, McGraw Hill.*
- *Tom White, (2012). HADOOP: The Definitive Guide,OReilly.*
- *Hurwitz, J., Nugent, A., Halper, F., & Kaufman, M. (2013). Big data for dummies (Vol. 336). Hoboken, NJ: John Wiley & Sons.*

Course Title: Machine Learning	L	T	P	Cr.
Course Code: MCA1408	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Describe the basic concepts of Bayesian Decision Theory.
2. Implement the working of perceptron learning algorithm, criterion and Windrow-Hoff learning algorithm.
3. Depict the algorithms like Nearest Neighbor classification, K-nearest neighbor and their applications.Evaluate the models generated from data.

Course Content

UNIT I **15 Hours**

Overview and Introduction to Bayes Decision Theory: Machine intelligence and applications, pattern recognition concepts classification, regression, feature selection, supervised learning class conditional probability distributions, Examples of classifiers bayes optimal classifier and error, learning classification approaches.

UNIT II **15 Hours**

Linear machines: General and linear discriminates, decision regions, single layer neural network, linear separability, general gradient descent, perceptron learning algorithm, mean square criterion and widrow-Hoff learning algorithm; multi-Layer Perceptron: two-layers universal approximates, back propagation learning, on-line, off- line error surface, important parameters.

UNIT III **15 Hours**

Learning decision trees: Inference model, general domains, symbolic decision trees, consistency, learning trees from training examples entropy, mutual information, ID3 algorithm criterion, C4.5 algorithm continuous test nodes, confidence, pruning, learning with incomplete data

Instance-based Learning: Nearest neighbor classification, k-nearest neighbor, nearest neighbor error probability.

UNIT IV **15 Hours**

Machine learning concepts and limitations: Learning theory, formal model of the learnable, sample complexity, learning in zero-bayes and realizable case, VC- dimension, fundamental algorithm independent concepts, hypothesis class, target class, inductive bias, Occam's razor, empirical risk, limitations of inference machines, approximation and estimation errors, Trade Offs.

Unsupervised Learning Algorithms: Clustering Techniques: K-Means, Hierarchical Clustering, DBSCAN, Association Rule Learning: Apriori Algorithm, Eclat Algorithm, Anomaly Detection Methods, Evaluation of Clustering Models.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *alp Aydin, E. (2020). Introduction to machine learning. MIT press.*
- *Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255-260.*
- *Mitchell, T. M., & Mitchell, T. M. (1997). Machine learning (Vol. 1, No. 9). New York: McGraw-Hill.*
- *Bishop, C. M., & Narbada, N. M. (2006). Pattern recognition and machine learning (Vol. 4, No. 4, p. 738). New York: Springer.*

Semester II

Course Title: Advanced Data Structure & Algorithms	L	T	P	Cr.
Course Code: MCA2450	3	0	0	3

Total Hours : 45

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Restate the fundamentals of basic data structures.
2. Implement the basic algorithms for sorting and searching.
3. Experiment the details of stack, queue and linked list operation.
4. Interpret the knowledge of tree and graphs concepts.
5. Apply algorithms and data structures in various real-life software problems.

Course Content

UNIT I **15 Hours**

Introduction to Data Structures & Algorithms Introduction of Data structures, Abstract Data Types, Performance Analysis: Space Complexity, Time Complexity, Asymptotic Notations (Big O, Omega, Theta), Performance measurement, Divide and Conquer, Back Tracking Method, Dynamic programming.

Sorting and searching algorithms Bubble sort, Insertion sort, Radix Sort, Quick sort, Merge sort, Heap sort, Selection sort, shell Sort, Linear Search, Sequential search, Binary search.

UNIT II **10 Hours**

Hashing Different Hashing Techniques, Address calculation Techniques, Common hashing functions, Collision resolution techniques: Linear probe, Quadratic probe, Key offset. Rehashing, Double hashing, Link list addressing.

UNIT III **10 Hours**

Linear Data Structures Stack Definition, Operations, Implementation of Stacks (Array and Linked list) and applications-Evaluation of postfix expression, Balancing of parenthesis

Queue: Definition, Operations, Implementation of simple queue (Array and Linked list) and applications of queue-BFS.

Types of queues: Circular, Double ended, Priority, Implementation using linked list Types of Linked List: Singly, Doubly and Circular Linked list Definition, Operations (Insert, delete, traverse, count, search)

UNIT IV **10 Hours**

Non-linear Data Structures Tree Definition and concepts, General Tree-Definition, Insertion and Deletion into general tree,

Binary Tree- Definition, Insertion and Deletion into binary tree, Traversal of a binary tree, Reconstruction of a binary tree from traversal, Conversion of general tree into binary tree, Huffman tree, Expression tree, Binary threaded three Binary Search Tree- Definion, Operation, Implementation AVL tree-

Definition, AVL tree rotation with examples, Heaps-Definition, Operations (insertion, delete, build) M way Tree- Introduction, B tree-definition and examples and B * 14

Graphs: Graph Representations: Adjacency Matrix and List, Graph Traversal Algorithms: Depth-First Search (DFS), Breadth-First Search (BFS), Applications: Topological Sorting, Shortest Path Algorithms (Dijkstra's, Bellman-Ford, Floyd-Warshall), Minimum Spanning Trees: Prim's and *Kruskal's Algorithms*

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Lipschutz, S. (2011). Data Structures with C (Schaum's Outline Series)*
- *Langsam, Y., Augenstein, M., & Tenenbaum, A. M. (1996). Data Structures using C and C++ (Vol. 2). New Jersey: Prentice Hall.*
- *Samanta, D. (2001). Classic data structures (Vol. 2). Prentice Hall India.*
- *Narahari, Y. (2000). Data structures and algorithms. Retrieved November, 15.2019.*

Course Title: Programming Using Python	L	T	P	Cr.
Course Code: MCA2451	3	0	0	3

Total Hours : 45

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Discuss the variable, expression and statements.
2. Apply conditional and looping constructs.
3. Design and import functions in python programming.
4. Attain the basics of Strings and Dictionaries.
5. Utilize basic operations on File.

Course Content

UNIT I **15 Hours**

Introduction to python Getting Started: Introduction to Python- an interpreted high-level language, interactive mode and script mode.

Variables, Expressions and Statements: Values, Variables and keywords; Operators and Operands in Python: (Arithmetic, relational and logical operators), operator precedence, Expressions and Statements (Assignment statement); Taking input (using raw input () and input ()and displaying output (print statement); Putting Comments.

UNIT II **10 Hours**

Conditional constructs and looping: if else statement While, for (range function), break, continue, else, pass, Nested loops, use of compound expression in conditional constructs and looping

Functions: Importing Modules (entire module or selected objects), invoking built in functions, functions from math module, using random () and randint () functions of random module to generate random numbers, composition.

Defining functions, invoking functions, passing parameters, scope of variables, void functions and functions returning values, flow of execution

UNIT III **10 Hours**

Strings: Creating, initializing and accessing the elements; String operators: +, *, in, not in, range slice [n:m]; Comparing strings using relational operators; String functions & methods: len, capitalize, find, isalnum, isalpha, isdigit, lower, islower, isupper, upper, lstrip, rstrip, isspace, istitile, partition, replace, join, split, count, decode, encode, swapcase, Pattern Matching

Lists: Concept of mutable lists, creating, initializing and accessing the elements, traversing, appending, updating and deleting elements; List operations (joining, list slices)

List functions & methods: len, insert, append, extend, sort, remove, reverse, pop.
Exception Handling and Regular Expressions: try, except, finally blocks, Raising exceptions and custom exceptions, Using regular expressions for pattern matching

UNIT IV **10 Hours**

Dictionaries: Concept of key-value pair, creating, initializing and accessing the elements in a dictionary, traversing, appending, updating and deleting elements. Dictionary functions & Methods: cmp, len, clear (), get (), has key (), items (), keys (), update (), values ()Tuples Immutable concept, creating, initializing and accessing the elements in a tuple; Tuple functions: cmp(), len(), max(), min(), tuple()
Input and Output: Output Formatting, Reading and Writing Files Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Predefined Clean-up Actions
Database Connectivity and Web Development (Advanced Topics):- Connecting to databases using Python (e.g., SQLite, MySQL), Executing CRUD operations, Introduction to web frameworks like Django, Developing simple web applications.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Dawson, M. (2008). Programing With Python.*
- *Harbour, J. S. (2012). More python programming for the absolute beginner. Course Technology, Cengage Learning.*
- *Beazley, D. M. (2009). Python essential reference. Addison-Wesley Professional.*
- *Van Rossum, G. (2007, June). Python Programming Language. In USENIX annual technical conference (Vol. 41, No. 1, pp. 1-36).*

Course Title: Advanced Data Structures & Lab	L	T	P	Cr.
Course Code: MCA2452	0	0	2	1

Total Hours : 30

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Create the applications of data structures.
2. Solve the algorithmic problems like insertion and deletion of data.
3. Interpret the programming code to implement the Link List Structure.
4. Analyze Singly, Doubly, Circular Singly linked lists and its operations.
5. Implement the insertion and deletion on BST and heap sort.

Course Content

1. Program to input 1-D Array
2. Program to perform insertion in Arrays
3. Program to perform deletion in Arrays
4. Program to input 2-D arrays (Matrices)
5. Program to find transpose of a matrix. Multiply 2 matrices.
6. Program to implement sparse matrices.
7. Program to perform linear search
8. Program to perform Binary search
9. Program to reverse array without using another variables.
10. Program to perform Bubble sort.
11. Program to perform sorting using Selection Sort.
12. Program to perform sorting using Insertion Sort.
13. Program to input and traverse N-nodes in a one way linked list.
14. Program to reverse a one way linked list.
15. Program to perform insertion/deletion in linked lists.
16. Program to input and traverse doubly linked list.
17. Program to implement stack operations.
18. Program to implement Queues.
19. Program to find factorial using recursion.
20. Program to print Fibonacci series using recursion.
21. Program to input a BST.
22. Program to perform insertion in a BST.
23. Program to perform deletion in a BST.
24. Program to implement min-heaps.
25. Program to implement max-heaps.
26. Program to implement AVL trees.
27. Program to perform rotations in AVL trees.
28. Program to perform rotations in AVL trees.
29. Program to input a graph.
30. Program to print adjacency list of a graph.
31. Program to perform traversal in graphs using DFS.

32. Program to perform traversal in graphs using BFS.
33. Program to implement shortest path methods.
34. Programs to perform Dynamic memory allocation.
35. Programs to perform sorting on data stored in a file.
36. Programs to delete duplicates in arrays and linked lists.

Course Title: Python Lab	L	T	P	Credits
Course Code: MCA2453	0	0	2	1

Total Marks:30

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Write, Test and Debug Python Programs.
2. Implement Conditionals and Loops for Python Programs.
3. Use functions and represent Compound data using Lists.
4. Implement the basic conditional and looping constructs.

Course Content

Program 1: Print hello world
Program 2: add numbers and concatenate strings
Program 3: input from user
Program 4: using loops (for,while)
Program 5: Loop control statements (break, continue, pass)
Program 6: if-else - conditional checking
Program 7: functions
Program 8: math library
Program 9: strings
Program 10: exceptional handling
Program 11: random numbers/string
Program 12: demo of data structure - list
Program 13: demo of data structure – dictionary
Program 14: demo of data structure – tuple
Program 15: command line argument
Program 16: Filter Even Numbers
Program 17: Convert Decimal to Binary
Program 18: Read a File
Program 19: Find the Largest Number in a List
Program 20: Sort a List

Course Title: Advanced Operating Systems	L	T	P	Cr.
Course Code: MCA2454	4	0	0	4

Total Marks:60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. On completion of this course the students will able to
2. Discuss the fundamentals of Operating System concepts.
3. Attain the mechanisms of OS to handle processes and threads.
4. Classify the role of paging, segmentation and virtual memory in operating systems.
5. Implement various Scheduling Algorithms.
6. Execute all the Deadlock Detection Algorithms.

Course Content

Unit I **15 Hours**

Introductory Concepts: Operating system functions and characteristics, Operating System classification, historical evolution of operating system, Real time system, Distributed system, Methodologies for implementation of O/S service, system calls, system programs, Interrupt mechanisms.

Processes: Process Concept, Process model, Process scheduling, Process states, process hierarchies, Process Synchronization, implementation of Processes, data structures used such as Process table, PCB creation of processes, context switching, exit of Processes.

UNIT II **15 Hours**

Inter-process communication: Race conditions, critical sections, problems of mutual exclusion, Peterson's solution, producer-customer problem, Reader Writer's Problem, Dining Philosophers Problem, semaphores, monitors, message passing, and Semaphores.

Process scheduling: objective, Scheduling Criteria, Scheduling Algorithms, preemptive vs. non-preemptive scheduling, comparative assessment of different algorithms such as round robin, priority bases scheduling. FCFS, SJF, multiple queues with feedback

Deadlocks: Deadlock Characteristics, Conditions, modeling, detection and recovery, deadlock avoidance, deadlock prevention, deadlock hand.

UNIT III **15 Hour**

The Linux: Linux History, Design Principles, Kernel Modules, Process Management, Scheduling, File Systems, Input and Output, Inter process Communication, Security.

Memory Management: Memory Allocation, concepts of Virtual Memory, Logical Versus Physical addresses space, Multiprogramming with fixed partition, variable partitions, virtual memory, paging, demand paging, design and

implementation issues in paging such as page tables, inverted page tables, page replacement algorithms, page fault handling, working set model, local vs. global allocation, page size, segmentation with paging, Frames.

UNIT IV **15 Hours**

File systems: File type, attributes, access and security, file operations, directory structures, path names, directory operations, implementation of file systems, implementation of file and file operation calls, implementation of directories, sharing of files, Allocation methods: Contiguous, linked and index allocation, block allocation, disk space management, free space management, logical file system, physical file system, Efficiency and Performance. Device management: Techniques for device management, dedicated devices, shared devices, virtual devices; device characteristics, hardware considerations: input & output devices, Storage devices: independent device operation, buffering, multiple paths, device allocation considerations.

Operating System Security: Authentication, Authorization, and Access Control, Intrusion Detection and Prevention, Virtualization and Virtual Machines (VMware, KVM, Hyper-V), Cloud Operating Systems and Containerization (Docker, Kubernetes)

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Tanenbaum, A. S. Andrew S. Tanenbaum. Distributed computing,*
- *Qureshi, I. (2014). Cpu scheduling algorithms: A survey. International Journal of Advanced Networking and Applications, 5(4), 1968.*
- *Mohan, I. C. (2013). Operating Systems. PHI Learning Pvt. Ltd.*
- *Duffy, K. P., Davis Jr, M. H., & Sethi, V. (2010). Demonstrating operating system principles via computer forensics exercises. Journal of Information Systems Education, 21(2), 195-202.*
- *Stalling, W. (2012). Operating system.*

Course Title: Advanced Computer Graphics	L	T	P	Cr.
Course Code: MCA 2455	3	0	0	3

Total Marks:45

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Be able to Compare various graphics algorithm used in 2D and 3D
2. Be able to understand fundamentals of graphics used in various real-life applications.
3. Be able to understand and identify the performance characteristics of graphics algorithms.
4. Employ algorithm to model engineering problems, when appropriate.

Course Content

Unit I **10 Hours**

Review of two-dimensional graphics: Transformations Windowing, ClippingThree Dimensions:3D geometry, primitives and transformations. Rotation about an arbitrary axis, Parallel and perspective projection Viewing parameters
3D clipping and viewing transformation, Curves and Fractals: Polygon Meshes Parametric Cubic curves: B-spline, Bezier, Hermite. Parametric Bicubic Surfaces Quadric surfaces Fractals: fractal lines and surfaces Applications

Unit II **10 Hours**

Solid Modeling: Representing solids, Regularized Boolean Set Operations Primitive Instancing, Sweep and Boundary Representations Spatial-partitioning, Representations Constructive Solid Geometry, User Interface for Solid Modeling, Achromatic and Colored Light: Achromatic light, Gamma correction, Halftone approximation, Chromatic Color,CIE chromaticity diagram, Color models for Raster Graphics., Using Color in Computer Graphics

Unit III **10 Hours**

Hidden Lines and Surfaces: Algorithms for Visible-Line and Surface determination: z- buffer, List priority, Scan line, Area Subdivision, Ray Tracing, Illumination and Shading: Surface detail, shadows and Transparency Inter object Reflections
Illumination Models Extended Light Sources, Recursive Ray Tracing

Unit IV **15 Hours**

Image based Rendering: Introduction, comparison with geometry-based rendering, applications, Graphics APIs: OpenGL, DirectX, Vulkan (overview and comparison), GPU Programming with GLSL or HLSL, Real-Time Rendering and Game Engines, Physically Based Rendering (PBR), Non-Photorealistic Rendering (NPR), Animation Techniques: Keyframing, Skeletal Animation, Motion Capture,

Virtual Reality (VR) and Augmented Reality (AR), Introduction to Graphics in AI and Machine Learning.

Transactional Mode:
Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Reading:

- *Hughes, John F., van Dam, Andries, McGuire, Morgan, Sklar, David F. (2013). Computer Graphics: Principles and Practice. 3rd Edition, Addison-Wesley, US.*
- *Shirley, Peter, Ashikhmin, Michael, Marschner, Steve. (2021). Fundamentals of Computer Graphics. 5th Edition, CRC Press, US.*
- *Akenine-Möller, Tomas, Haines, Eric, Hoffman, Naty. (2018). Real-Time Rendering. 4th Edition, CRC Press, US.*
- *Pharr, Matt, Jakob, Wenzel, Humphreys, Greg. (2016). Physically Based Rendering: From Theory to Implementation. 3rd Edition, Morgan Kaufmann, US.*
- *Glassner, Andrew S. (1989). An Introduction to Ray Tracing. Academic Press, US.*

Course Title: ADVANCED COMPUTER GRAPHICS LAB	L	T	P	Cr.
Course Code: MCA2456	0	0	2	1

Total Marks:30

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. To Introduce various Graphics Applications in real world scenario
2. To be familiar with image fundamentals and animations
3. To be learn more about 2D, 3D and Curve applications
4. Applying efficient graphics technique to solve engineering problems

Course Content

Sr. No	Name of Experiment
1	To perform open GL program for Bezier Curve.
2	To perform Bezier curve with c^0 and c^1 continuity.
3	To Draw cube with or without back face culling.
4	To perform Hermite Curve.
5	To perform program for diffuse illumination.
6	To perform program for sphere with Back face culling.
7	To perform program for Ambient and diffuse light source.
8	To perform program for ambient & specular & diffuse light source.
9	To perform program for Diffuse only light source.
10	To perform Z buffer visible surface Algorithm
11	To perform open GL program for Bezier Curve.
12	Create a basic particle system to simulate fire, smoke, or water.
13	Implement particle physics for motion and collision.
14	Implement particle physics for motion and collision.
15	Implement basic lighting models (Phong, Gouraud, Flat Shading).
16	Implement a basic PBR pipeline using shaders.
17	Create non-photorealistic effects like cartoon rendering or sketch effects.
18	Implement stereoscopic rendering for VR.

Course Title: ADVANCED WEB TECHNOLOGIES	L	T	P	Cr.
Course Code: MCA2457	0	0	4	2

Total Hours :60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Students are able to develop a dynamic webpage by the use of java script and DHTML.
2. Students will be able to write a well formed / valid XML document.
3. Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
4. Students will be able to write a server-side java application called Servlet to catch form data sent from client, process it and store it on database.
5. Students will be able to write a server-side java application called JSP to catch form data sent from client and store it on database.

Course Content

1. Create a simple webpage using HTML.
2. Use frames to Include Images and Videos.
3. Add a Cascading Style sheet for designing the web page.
4. Design a dynamic web page with validation using JavaScript.
5. Design an HTML having a text box and four buttons viz Factorial,
6. Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate javascript function should be called to display
7. Factorial of that number
8. Fibonacci series up to that number
9. Prime numbers up to that number
10. Is it palindrome or not
11. Write java script programs to demonstrate
12. Math Object with at least five methods.
13. String Object with at least five methods.
14. Array Object with at least five methods.
15. Date Object with at least five methods.
16. Write JavaScript programs on Event Handling
17. Validation of registration form
18. Open a Window from the current window
19. Change color of background at each click of button or refresh of a page
20. Display calendar for the month and year selected from combo box
21. On Mouse over event
22. Write a java program to connect to a database server using JDBC and insert 10 student's information of user choice in to student table.
23. Write a java program to display all records in the student table.
24. Develop a simple Servlet to display Welcome to Servlet.

25. Develop a Servlet to validate user name and password with the data stored in Servlet configuration file. Display authorized user if she/he is authorized else display unauthorized user.
26. Develop a Servlet to validate user name and password stored in database. Display authorized user is she/he is authorized else display unauthorized user.
27. Write a Servlet program to store student details sent from registration form in to database table.
28. Write JSP Program to store student information sent from registration page into database table.
29. Develop a program to validate username and password that are stored in Database table using JSP.
30. Write appropriate JSP pages to insert, update and delete data in student table in a single application with proper linking of JSP pages and session management.

Suggested Reading:

- *Chris Bates, "Web Programming-Building Internet Applications", 2 ed., Wiley Publishers, 2006.*
- *Diet eland Nieto, "Internet and World Wide Web-How to Program", 4*
- *Pearson Education Asia,2007.ed., PHI/*
- *Hans Bergsten, "Java Server Pages", 1sJennifer Niederst, Robbins, "Learning Web Design", 3rd ed., SPD O'REILLY Publications, 10.*
- *Firuza Aibara, "HTML for Beginners", 2nd ed., SPD O'REILLY Publications, 2010.*
- *Marty Hall, "Core Servlets and Java Server Pages", 1 st ed., Prentice Hall PTR, 2000.*
- *The complete Reference Java 2 Fifth Edition by Patrick Naughton and Herbert Schildt. TMH ed., O'REILLYPublications,2000.*

Course Title: IoT & Its Applications	L	T	P	Cr.
Course Code: MCA2458	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Identify the different types of sensors and devices used in IoT.
2. Understand the security and privacy challenges associated with IoT.
3. Compare and contrast different IoT platforms and architectures
4. Develop IoT prototypes using hardware and software components.

Course Content

UNIT I **15 hours**

FUNDAMENTALS OF IoT- Evolution of Internet of Things, Enabling Technologies, M2M Communication, IoT World Forum (IoTWF) standardized architecture, Simplified IoT Architecture, Core IoT Functional Stack, Fog, Edge and Cloud in IoT, Functional blocks of an IoT ecosystem, Sensors, Actuators, Smart Objects and Connecting Smart Objects

UNIT II **15 hours**

IoT PROTOCOLS- IoT Access Technologies: Physical and MAC layers, topology and Security of IEEE 802.15.4, 802.11ah and Lora WAN, Network Layer: IP versions, Constrained Nodes and Constrained Networks,6LoWPAN, Application Transport Methods: SCADA, Application Layer Protocols: CoAP and MQTT

UNIT III **15 hours**

DESIGN AND DEVELOPMENT- Design Methodology, Embedded computing logic, Microcontroller, System on Chips, IoT system building blocks IoT Platform overview: Overview of IoT supported Hardware platforms such as: Raspberry pi, Arduino Board details

UNIT IV **15 hours**

Data Analytics: Introduction, Structured Versus Unstructured Data, Data in Motion versus Data at Rest, IoT Data Analytics Challenges, Data Acquiring, Organizing in IoT/M2M

Supporting Services: Computing Using a Cloud Platform for IoT/M2M Applications/Services, Everything as a service and Cloud Service Models.

Applications of IoT: Smart Home and Smart Cities, Industrial IoT (IIoT) and Smart Manufacturing, IoT in Healthcare and Agriculture, Intelligent Transportation and Smart Grid, Wearable Devices and Environmental Monitoring, Future Trends in IoT (AI + IoT, Blockchain in IoT, etc.) CASE STUDIES/INDUSTRIAL APPLICATIONS: IoT applications in home, infrastructures, buildings, security, Industries, Home appliances, other IoT

electronic
equipment's, Industry 4.0 concepts.

Transactional Mode:
Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, Cisco(2017) ,IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, Press.*
- *Arshdeep Bahga, Vijay Madisetti (2015), Internet of Things – A hands-on approach, Universities Press.*
- *Raj Kamal, Internet of Things: Architecture, Design Principles and Applications, McGraw Hill Higher Education.*

Course Title: Digital Image Processing	L	T	P	Cr.
Course Code: MCA2459	4	0	0	4

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Understand the fundamentals of digital image and its processing
2. Apply image enhancement techniques in spatial and frequency domain.
3. Elucidate the mathematical modeling of image restoration and compression
4. Describe object detection and recognition techniques.

Course Content

UNIT I **15 Hours**

Background: Introduction to electronic systems for image transmission and storage, computer processing and recognition of pictorial data, overview of practical applications.

Fundamentals: Mathematical and perceptual preliminaries, human visual system model, image signal representation, imaging system specification building image quality, role of computers, image date formats.

UNIT II **15 Hours**

Image Processing Techniques: Image enhancement, image restoration, image feature extraction, image data compression and statistical pattern recognition.

Hardware Architecture for image processing: Distributed processing of image data, role of array processing, standard image processor chips (as example).

UNIT III **15 Hours**

Techniques of color image processing: Color image signal representation, color system transformations, extension of processing techniques to color domain.

UNIT IV **15 Hours**

Applications of Image Processing: Picture data archival, machine vision, medical image processing. Image Segmentation: Edge detection: Sobel, Prewitt, Canny, Thresholding techniques: Global and adaptive, Region-based segmentation, Morphological operations: Dilation, erosion, opening, closing, Object recognition and classification, Introduction to deep learning in image analysis (CNN basics)

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Petrou, M. M., &Petrou, C. (2010). Image processing: the fundamentals. John Wiley &*

Sons.

- *Gonzalez, R. C., & Woods, R. E. (1992). Digital image processing addison- wesley. Reading, Ma.B. Channda & D.Dutta.Digital Image Processing and Analysis.*
- *Jain, A. K. (1989). Fundamentals of digital image processing. Prentice- Hall, Inc.*
- *Pitas, I. (2000). Digital image processing algorithms and applications. John Wiley & Sons.*
- *Cristóbal, G., Schelkens, P., & Thienpont, H. (Eds.). (2013). Optical and digital image processing: fundamentals and applications. John Wiley & Sons.*

Course Title: Data Warehousing and Data Mining Techniques	L	T	P	Cr.
Course Code: MCA2460	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Identify the scope and necessity of Data Mining & Warehousing for the society
2. Describe the designing of Data Warehousing so that it can be able to solve the root problems.
3. Remove redundancy and incomplete data from the dataset using data preprocessing methods.
4. Develop a data mining application for data analysis using various tools.

Course Content

UNIT I **15 Hours**

Introduction: Data Warehousing: Definition, Characteristics of a Data Warehouse, Data warehouse Usage, DBMS vs. Data warehouse

Developing Data Warehouse: Data warehousing components, Steps and Crucial decisions for the design and construction of Data Warehouses, Three-tier Data warehouse architecture, Data Warehouse Implementation, Design, performance and technological considerations, Metadata.

UNIT II **15 Hours**

Developing Data Mart based Data Warehouse Types of data marts, Metadata for a data mart, Data model for a data mart, Maintenance of a data mart, Software components for a data mart, Performance issues, Security in data mart.

OLAP Systems Types of OLAP, Relational vs. Multidimensional OLAP, Data modeling: Star schema, Snowflake schema, OLAP tools

UNIT III **15 Hours**

Data Mining: Introduction to data mining, Data mining process, Major issues and Application of Data mining, Data preprocessing: Data cleaning, Data integration and transformation and Data reduction; Tools for data mining.

Data Mining Techniques: Association rules: Introduction, Market basket analysis, Frequent Pattern Mining algorithms: Apriori algorithm, Partition algorithm.

UNIT IV **15 Hours**

Classification and Prediction: Definition, Issues regarding Classification and Prediction, Classification by Decision Tree Induction, Support Vector Machines, k- Nearest-Neighbor, Prediction: Linear and Non-Linear Regression Clustering:

Definition, Types of data in cluster analysis, clustering paradigms: K-Means and K-Medoids, Mining Sequence patterns: Generalized Sequential Patterns (GSP) mining algorithm, Hidden Markov Model, Social Network Analysis. Advanced Topics and Applications Web mining: Web content, structure, and usage mining, Text mining and opinion mining basics, Temporal and spatial data mining, Big data analytics and Hadoop ecosystem overview, Applications of data mining in business, healthcare, e-commerce, finance, Ethical issues and privacy in data mining

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Dunham Margaret H, Sridhar S. (2008). Data mining: Introductory and Advanced Topics, Pearson Education.*
- *HumphiresH.D.(2009). Data Warehousing: Architecture and Implementation Pearson Education.*
- *Anahory.(2008). Data Warehousing in the Real World. Pearson Education.*

Semester – III

Course Title: Artificial Intelligence	L	T	P	Cr.
Course Code: MCA3500	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Differentiate the various searching techniques, constraint satisfaction problem.
2. Classify the role of agents and the way of evaluating it.
3. Analyze and design a real-world problem for implementation and understand the dynamic behavior of a system.
4. Compare different machine learning techniques to design AI machines and enveloping applications for real world problems.

Course Content

UNIT I **15 Hours**

Introduction to Artificial Intelligence (AI) and Problem Space: Introduction AI technique, Turing test, History and developments in AI, applications of AI, State space representation, production systems, systematic control strategies: Breadth first search and Depth first search, problem characteristics, product system characteristics, issues in the design of search programs.

Heuristic Search Technologies: Introduction to heuristic search, Generate and test, Hill Climbing, Best First search, A*, Problem reduction, AO*.

UNIT II **15 Hours**

Knowledge representation methods - Propositional logic and first order predicate logic, Resolution principle, Semantic networks, partitioned semantic nets, Frames, Scripts and conceptual dependencies.

Game playing: Minimax search procedure, reducing alternatives using Alpha-Beta pruning method examples.

UNIT III **15 Hours**

Approaches to AI: Turing Test and Rational Agent Approaches; State Space Representation of Problems, Heuristic Search Techniques, Game Playing, Min-Max Search, Alpha Beta Cutoff Procedures.

Natural Language Processing: Grammar and Language; Parsing Techniques, Semantic Analysis and Pragmatics.

UNIT IV **15 Hours**

Tools and Technologies for AI: Introduction to AI language, Natural Language Processing (NLP), Ethical and Social Implications of AI, AI Trends and Future Directions,

LISP: Symbolic expression, creating, appending and modifying lists, defining functions, Predicates, Conditionals, Recursion, Iteration, Printing and reading, Lambda expressions and higher order function, List storage.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Khemani , D. (2013). A first course in artificial intelligence. McGraw-Hill Education.*
- *Fu, L. M. (2003). Neural networks in computer intelligence. Tata McGraw- Hill Education.*
- *Kamruzzaman, A. M. Artificial Intelligence & Applications.*
- *Russell, S. J. (2010). Artificial intelligence is a modern approach. Pearson Education, Inc.*

Course Title: Research Methodology	L	T	P	Cr.
Course Code: MCA3501	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Understand key research methodology concepts and issues
2. Identify the role and importance of research in the Computer Applications
3. Identify the concepts and procedures of sampling, data collection, analysis and reporting.
4. Analyze appropriate research problem and parameters
5. Implement the basic concepts of research and its methodologies

Course Content

UNIT I **15 Hours**

Research: its concept, nature, scope, need and Objectives of Research, Research types, Research methodology, Research process – Flow chart, description of various steps, Selection of research problem.

UNIT II **15 Hours**

Research Design: Meaning, Objectives and Strategies of research, different research designs, important experimental designs,

Methods of Data Collection and Presentation: Types of data collection and classification, Observation method, Interview Method, Collection of data through Questionnaires, Schedules, data analysis and interpretation, editing, coding, content analysis and tabulation

UNIT III **15 Hours**

Sampling Methods: Different methods of Sampling: Probability Sampling methods, Random Sampling, Systematic Sampling, Stratified Sampling, Cluster Sampling and Multistage Sampling. Non probability Sampling methods, Sample size.

UNIT IV **15 Hours**

Report writing and Presentation: , Interpretation and Report Writing, Types of reports, Report Format/ Thesis – Cover page, Introductory page, Text, Bibliography, Appendices, Typing instructions, Oral Presentation, Thesis, Referencing Styles: APA, MLA, Chicago, Bibliography and Citation Tools, Research Presentation: Oral and Poster Presentations, Publishing Research Papers: Journals, Conferences, Indexing

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Panneerselvam, R, Research Methodology, PHI, New Delhi.*
- *Cooper, D.R.,Schindler,P.S., Business Research Methods, Tata McGraw Hill*
- *Gupta S P,Statistical Methods, Sultan Chand & Sons, Delhi*
- *Ronald E Walpole, Probability and Statistics for Engineers and Scientists (International Edition), Pearson Education.*
- *Geode, Millian J. & Paul K. Hatl, Methods in Research, McGraw Hills, New Delhi*
- *Kothari C.R., Research Methodology, New Age Publisher*
- *Sekran, Uma, Business Research Method, Miley Education, Singapore*

Course Title: Network Administration	L	T	P	Cr.
Course Code: MCA350	0	0	4	2

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Configure, manage, and troubleshoot network devices, including routers, switches, and firewalls.
2. Implement network security measures to protect against vulnerabilities and threats.
3. Monitor and optimize network performance using various tools and techniques.
4. Establish and maintain user access policies and authentication mechanisms.
5. Design and manage scalable networks to meet organizational requirements.

Course Content

1. Server/Client Installation over VMware Environment
2. Packet Analysis by using TCPDUMP and WIRESHARK 149
3. Network Practice with Packet Tracer
4. System Administration: User/Group management, File System Management ….
5. Network Configuration: Start/Stop network Service, network interface configuration
6. Firewall Configuration
7. DNS and DHCP Configuration and Troubleshooting
8. Web and Proxy Server Configuration and Troubleshooting
9. Basic Mail Server Configuration and Troubleshooting
10. SAMBA, NFS, CUPS and FTP configuration and Troubleshooting
11. SDN controller installation and client network implementation (OpenDaylight)
12. Network topology programming with Mininet and visualization
13. Backup and Disaster Recovery Planning
14. Advanced Network Administration and Troubleshooting
15. Network Performance Optimization
16. Server Administration Basics (e.g., Windows Server, Linux Server)

Course Title: Dissertation (Phase 1)	L	T	P	Cr.
Course Code: MCA3503	0	0	0	12

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Apply knowledge of recent computing technologies, skills and current tools of computer science and engineering.
2. Design and conduct experiments, as well as to analyze and interpret data.
3. Understand the contemporary research issues in the different areas of computer science & engineering.
4. Explore research gaps, analyze and carry out research in the specialized/ emerging areas.

Course Content

Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations with implementation tools with suitable platform.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Semester IV

Course Title: Java Programming	L	T	P	Cr.
Course Code: MCA4550	3	0	0	3

Total Hours : 45

Learning Outcomes: After completion of this course, the learner will be able to:

1. Describe the concepts of data types, variables, arrays of object-oriented programming.
2. Analyze interfaces, class hierarchies and exception in programs.
3. Construct appropriate diagrams and textual descriptions to communicate the AWT and Applet for web applications.
4. Implement the concept of SQL package, multithreading and JDBC in java.
5. Solve specified problems by using the Java SDK environment to create, debug and run simple Java programs.

Course Content

UNIT I **11 Hours**

Introduction to Java: Introduction to java, Java History, Java Features; How Java Differs from C and C++; Comments in java, Java Program Structure, Implementing a Java Program, Java Virtual Machine, Command Line Arguments, Programming Style, Java and Internet, Java and World Wide Web, Web Browsers; Hardware and Software Requirements; Java Support Systems, Java Environment, Java Tokens; Java Statements.

Constants, Variables and Data Types: Introduction; Constants, Variables, Data Types, Introduction to Operators, Expressions, Operator Precedence. Decision Making,

Branching and Looping: Decision making and branching Statements, Looping Statements, labeled loops, Jumping Statements.

UNIT II **11 Hours**

Classes, Objects and Methods: Introduction, defining a Class, Data member, member function, Creating Objects, Accessing Class Members, Constructors, Methods Overloading, Static Members, Nesting of Methods. Arrays, Strings, Vectors: Arrays, Zagged Arrays, And Strings, String functions: Vectors, Wrapper Classes.

Inheritance: Extending a Class, Overriding Methods, Final Variables and Methods, Final Classes, Finalizer Methods, Abstract Methods and Classes, Visibility Control.

UNIT III **11 Hours**

Interfaces: Introduction, Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variables, Implementing Multiple Inheritance using Interfaces.

Packages: Introduction; System Packages, Using System Packages, Naming Conventions, Creating Packages, accessing a Package, Using a Package, Adding a Class to a Package, Hiding Classes.
Managing Errors and Exceptions: Introduction; Types of Errors; Exceptions, Exception Handling using Try, Catch and Finally block: Throwing Our Own Exceptions, Using Exceptions for Debugging.

UNIT-IV **12 Hours**

Multithreading: Creating Threads, Thread Life Cycle, Synchronization, Thread Communication, Thread Priorities.
Applet Programming: How Applets Differ from Applications, Applet Life Cycle; Creating an Executable Applet.
JAVA Database Connectivity (JDBC): Merging data from Multiple tablets, joining, Manipulation: Database with JDBC, Prepared Statements, Transaction processing. Advanced Java Concepts: String handling and StringBuffer
Multithreading: Thread class, Runnable interface

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings:

- *Balaguruswamy, E. (2014). Programming with Java-A Primer. McGraw- Hill Professionals.*
- *Chhillar, R. S. (2011). Comparison of C++ and JAVA in the Context of Object-Oriented Metrics. IUP Journal of Computer Sciences, 5(2).*
- *Sedgewick, R., & Wayne, K. (2017). Introduction to programming in Java: an interdisciplinary approach. Addison-Wesley Professional.*
- *Lewis, G., Barber, S., & Siegel, E. (1998). Programming with Java IDL. John Wiley & Sons, Inc.*

Course Title: Java Programming Lab	L	T	P	Cr.
Course Code: MCA4551	0	0	2	1

Total Hours : 30

Learning Outcomes: After completion of this course, the learner will be able to:

1. Classify the structure and model of the Java programming language.
2. Implement the given problems in Java programming language.
3. Develop software in the Java programming language.
4. Evaluate user requirements for software functionality required to decide whether the Java Programming Language can meet user requirements.
5. Connect Java programs to database using JDBC.

Course Content

1. Write a java program to find the Fibonacci series using recursive and non recursive functions.
2. Write a java program to multiply two given matrices.
3. Write a java program that checks whether a given string is palindrome or not.
4. Write a java program for Method overloading and Constructor overloading.
5. Write a java program to represent Inheritance.
6. Write a java program to display the employee details using Scanner class.
7. Write a java program to represent Abstract class with example.
8. Write a java program to implement interface using extends keyword
9. Write a java program to create user defined package
10. Write a java program for creating multiple catch blocks.
11. Write an applet program that displays a simple message.
12. Write a Java program compute factorial value using Applet.
13. Write a java program for handling Mouse events and Key events.
14. Write a java program that connects to a database using JDBC.
15. Write a java program that String manipulation exercises
16. Write a java program to Implementing multithreading using Thread and Runnable.
17. Write a java program that file handling: reading from and writing to files.
18. Write a java program that Building simple GUI applications using AWT/Swing components (buttons, text fields).

Course Title: Mobile Application Development	L	T	P	Cr.
Course Code: MCA4552	0	0	4	2

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Identify the various concepts of mobile programming that make it unique from programming for other platforms.
2. Utilize rapid prototyping techniques to design and develop sophisticated mobile interfaces.
3. Program mobile applications for the Android operating system that use basic and advanced phone features.
4. Deploy applications to the Android marketplace for distribution.

Course Content

UNIT I **15 Hours**

Mobile Application Development - Mobile Applications and Device Platforms - Alternatives for Building Mobile Apps -Comparing Native vs. Hybrid Applications -The Mobile Application Development Lifecycle-The Mobile Application Front-End-The Mobile Application Back-End Key Mobile Application Services-What is Android-Android version History-Obtaining the Required Tools-Launching Your First Android Application-Exploring the IDE-Debugging Your Application-Publishing Your Application

UNIT II **15 Hours**

Understanding Activities-Linking Activities Using Intents-Fragments- Displaying Notifications Understanding the Components of a Screen- Adapting to Display Orientation-Managing Changes to Screen Orientation- Utilizing the Action Bar-Creating the User Interface Programmatically Listening for UI Notifications

UNIT III **15 Hours**

Using Basic Views-Using Picker Views -Using List Views to Display Long Lists-Understanding Specialized Fragments - Using Image Views to Display Pictures - Using Menus with Views Using Web View- Saving and Loading User Preferences-Persisting Data to Files-Creating and Using Databases.

UNIT IV **15 Hours**

Sharing Data In Android-Creating Your Own Content Providers -Using The Content Provider SMS Messaging -Sending Email-Displaying Maps- Getting Location Data- Monitoring A Location.

Consuming Web Services Using HTTP-Consuming JSON Services- Creating Your Own Services - Binding Activities To Services -Understanding Threading. Advanced Mobile Development Concepts: Multimedia Integration: Audio, Video,

Camera, Location-Based Services And Google Maps API, Notifications And Alarms, Publishing Apps On Google Play Store / App Store Basics, Introduction To Cross-Platform Frameworks (Flutter, React Native).

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Steele, J., To, N., & Conder, S. (2011). The Android Developer's Collection (Collection). Addison-Wesley Professional.*
- *Meier, R. (2012). Professional Android 4 application development. John Wiley & Sons.*
- *Burd, B. (2015). Android application development all-in-one for dummies. John Wiley & Sons.*
- *Charland, A., &Leroux, B. (2011). Mobile application development: web vs. native. Communications of the ACM, 54(5), 49-53.*

Course Title: Dissertation (Phase 2)	L	T	P	Cr.
Course Code: MCA4553	0	0	0	12

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Apply knowledge of recent computing technologies, skills and current tools of computer science and engineering.
2. Design and conduct experiments, as well as to analyze and interpret data.
3. Understand the contemporary research issues in the different areas of computer science & engineering.
4. Explore research gaps, analyze and carry out research in the specialized/ emerging areas.

Course Content

Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations with implementation tools with suitable platform.

Transactional Mode:

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Course Title: Theory of Computation	L	T	P	Cr.
Course Code: MCA4554	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Recognize and comprehend formal reasoning languages.
2. Usethe basic concepts of formal languages of finite automata techniques.
3. Design different types of Finite Automata and Machines as Acceptor, Verifier and Translator.
4. Design different types of Push down Automata as Simple Parser.
5. Analyze Context Free languages, Expression and Grammars.

Course Content

UNIT I **13 Hours**

Introduction: Basic Terminology: Alphabet, Formal Language and operations on formal languages, Examples of formal languages.

Finite automata: Concept of Basic Machines, Properties and Limitations of Finite State Machines, Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Equivalence of DFA and NDFA, Non-Deterministic Finite automata with Λ- Transitions.

UNIT II **14 Hours**

Regular expression: Regular Languages and Regular Expressions, Kleen's Theorem. Arden's Method.

Properties of Regular sets: The Pumping Lemma for Regular sets, Application of the Pumping Lemma, Closure Properties of Regular Sets, Myhill- Nerode Theorem and Minimization of Finite Automata, Minimization Algorithm.

Finite Automata with output: Moore and Mealy Machines. Equivalence of Moore and Mealy Machines.

UNIT III **17 Hours**

Context Free Grammars: Examples and Definitions, Derivation trees and ambiguity, An Unambiguous CFG for Algebraic Expressions. Regular Grammar, Simplified forms and Normal forms: Removal of useless symbols and UNIT production, Removal of Λ-moves, Chomsky Normal Form (CNF), Greenback Normal Form (GNF).

Pushdown Automata: Introduction and Definition of Push-Down Automaton, Applications of Push down Automata.

UNIT I **16 Hours**

Turing Machines: Definitions and Examples, Deterministic and Non-

Deterministic Turing Machines, Unsolvable Problems: A Non-recursive Language and an Unsolvable Problem, PCP Problem and MPCP Problem.
More General Languages and Grammars: Recursively Enumerable and Recursive Languages, Unrestricted grammars, Context sensitive Language and grammar. Relation between languages of classes, Chomsky hierarchies of grammars.
Recursive And Recursively Enumerable Languages
Undecidability: Halting Problem, PCP (Post Correspondence Problem)

Transactional Mode:
Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Sipser, M. (1996). Introduction to the Theory of Computation. ACM Sigact News, 27(1), 27-29.*
- *Dai, H. Y. (2015). McCarthy's LISP and Basis for Theory of Computation. In Third International Conference on the History and Philosophy of Computing (p. 39).*
- *Kozen, D. C. (2006). Theory of computation (Vol. 121). Heidelberg: Springer.*
- *Core, A. (1998). Theory of Computation.*

Course Title: Principles and Practices of Management	L	T	P	Cr.
Course Code: MCA4555	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Understand the fundamental principles and functions of management in organizational contexts.
2. Apply management practices to plan, organize, lead, and control effectively.
3. Analyze real-world business scenarios to develop problem-solving and decision-making skills.
4. Demonstrate the ability to manage resources and foster teamwork for achieving organizational goals.

Course Content

Unit -I **Hours 15**

Forms of Business Organizations and Ownership: Sole Proprietorship, Partnership, Joint Stock Company, Public & Private undertakings, Government Companies. Management: Meaning & Definition of Management, Nature, Scope and its various functions. Evolution of management thoughts: classical and new classical systems, contingency approaches, Scientific management.

Unit – II **Hours15**

Planning: nature, purpose and functions, types of plan, Management by Objective (MBO), steps in planning. Decision Making: Meaning, Steps in Decision Making, Techniques of Decision Making. Strategic planning – concepts, process, importance and limitations; Growth strategies- Internal and external.

Unit – III **Hours 15**

Organizing: Concept, formal and informal organizations, task force, bases of departmentation, different forms of organizational structures, avoiding organizational inflexibility. Teamwork – meaning, types and stages of team building.

Concept of staffing– Recruitment and Selection.

Motivation – concept, importance and theories.

Unit- IV **Hours 15**

Authority: definition, types, responsibility and accountability, delegation, decentralization v/s centralization, determinants of effective decentralization. Line and staff authority.

Control: function, process and types of control, nature, process, significance and span of control. Direct control v/s preventive control.Trends and challenges of

management in global scenario, emerging issues in management: Introduction to Total Quality Management (TQM), Just in Time (JIT).

Transactional Mode:
Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Reading:

- *Principles and practices of management: L. M. PRASAD (S. Chand publishers)*
- *Essentials of Management: Koontz H. & Weihrich H. (Tata Mc Graw Hill Publishers)*
- *Management: Stephen Robbins (Pearson publishers)*

Course Title: Software quality and testing assurance	L	T	P	Cr.
Course Code: MCA4556	4	0	0	4

Total Hours : 60

Course Learning Outcomes: After completion of this course, the learner will be able to:

1. Explain and Apply Knowledge of Key Concepts of Software Testing, Quality and Testing Tools.
2. Draw the DD Graph and Identify the Various Test Cases from Paths of Flow Graph of Software Testing Problem and Determine the Complexity of Software.
3. Design Test Cases and Develop Test Suite, Write Test Scripts, Set Environmental Variables for Carrying Out the Various Levels of Testing Manually and Automatically.
4. Manage Software Defects, and Risks Within a Software Project.
5. Work Effectively in Profile of Software Tester, Quality Assurance and Control officer, Project Manager and Leaders

Course Content

UNIT-I **15 Hours**

Software Testing Process, Objectives, Testing Techniques, Software Testing Life Cycle, Concept of Testing, Types of Errors, Stubs and Drivers Verification and Validation, Different Types of Verification & Validations Mechanisms, Concepts of Software Reviews, Code Inspection and Code Walkthrough, Testing of Component Based Software System, Energy Efficient Testing, Mobile Application Testing.

UNIT-II **15 Hours**

Software Testing Methods, Testing Fundamentals, Test Case Design, White Box Testing and its Types, Black Box Testing and its Types, Software Testing Strategies, Strategic Approach to Software Testing, UNIT Testing, Integration Testing, Validation Testing, System Testing, Test Planning, Budgeting and Scheduling.

UNIT-III **15 Hours**

Software Testing Metrics, Concept and Developing Testing Metrics, Different Types of Metrics, Complexity Metrics, Defect Management, Definition of Defects, Defect Management Process, Defect Reporting, Metrics Related to Defects, Using Defects for Process Improvement.

UNIT-IV **15 Hours**

Software Quality, Factors Affecting Software Quality, Quality Models, Software Quality Estimation, Quality Metrics, Quality Assurance, SQA Activities, Software Reviews, Formal Technical Reviews, Quality Control Quality Management, and,

SQA Plan. Quality Improvement, Pareto Diagrams, Cause- Effect Diagrams, Scatter Diagrams, Run Charts, Total Quality Management, Statistical Quality Assurance, Software Reliability, the ISO 9001 Quality Standard, Six Sigma, Informal Reviews.
Quality Costs, Quality Cost Measurement, Utilizing Quality Costs for Decision-Making. Manual Vs Automatic Testing, Basics of Automated Testing, Drawback of Manual Testing, Advantages of Automation of Testing, Factors for Automation Testing, Types Automation of Testing Tools, Introduction to QTP, QTPIDE, Basic Components in QTP, QTP Framework, Write Scripts, Introduction to Winrunner, and Rational Robot.

Transactional Mode:
Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Reading:

- *K.K. Aggarwal& Yogesh Singh, "Software Engineering", 2nd Ed., New Age International Publishers, New Delhi, 2005.*
- *KshirsagarNaik, PriyadarshiTripathy, Software Testing and Quality Assurance Theory and Practice. Wiley-Spektrum; I Edition, August 18, 2008.*
- *Donna C. S. Summers, Quality Management, Pearson; 2 Edition, April 26, 2008*
- *Yogesh Singh, Software Testing, Cambridge University Press, 2012 William Perry, "Effective Methods for Software Testing", John Wiley & Sons, New York, 1995 7. Louise Tamres, "Software Testing", Pearson Education Asia, 2002*
- *CemKaner, Jack Falk, Nguyen Quoc, "Testing Computer Software", Second Edition, Van Nostrand Reinhold, New York, 1993.*
- *Boris Beizer, "Black-Box Testing - Techniques for Functional Testing of Software and Systems", John Wiley & Sons Inc., New York, 1995.*