

The Post Graduate Board of Studies
(PG BOS) held on 9th July 2026 in the
Department of ICT

SCHEME AND SYLLABI

of

TWO YEAR POSTGRADUATE PROGRAMME

**Master in Computer
Applications (MCA)**



WITH EFFECT FROM THE

SESSION 2026-27

Chaudhary BansiLal University,

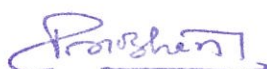
Prem Nagar, Bhiwani (127031)

Haryana, India

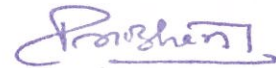
Preface

The overall well-being of a nation depends on the eminence of its human resource. Providing quality education plays a vital role in transforming people into valuable human resource. Well-educated students of today will become innovators and leaders of tomorrow who are going to ensure a constructively competitive but sustainable and peaceful world for everyone. Keeping in the view the demand of the skills based on Computer Applications, the university has introduced a Master in Computer Applications in the Department of Computer Science & Engineering. The curriculum is designed around the framework of Outcome-Based Education (OBE) in which students are at the centre of teaching learning process. The salient features of the curriculum design are as follows:

1. To start with, four Programme Educational Outcomes are defined.
2. The twelve Programme Outcomes (POs) are taken from the Self Appraisal Report format of National Board of Accreditation (NBA) for postgraduate programmes and two Programme Specific Outcomes (PSOs) are outlined to capture the MCA programme.
3. An induction programme of three weeks has been introduced to make the admitted students comfortable in their new environment. The induction programme continues in the form of participation in Sports club or Green club or Cultural, Literature and Film Club etc. for the remaining period of the programme. It is mandatory for every student to join in one of these clubs.
4. In addition to the core and elective courses, there is a provision for many courses from Value Added Courses (VAC), Ability Enhancement Courses (AEC), Skill Enhancement Courses (SEC), Multidisciplinary Courses (MDC). The non-credit mandatory courses are included to make students aware about constitution of India, issues related to environmental and sustainable development, and Indian traditional wisdom.
5. For every course, 4 to 6 Course Outcomes (COs) are defined which are concrete and measurable.
6. Guidelines for preparing sessional examination question papers and assignments have been framed for measuring the attainment levels of COs.
7. The internal and external evaluation criteria for various courses have been succinctly described.
8. The Course Outcomes (COs) are mapped to Programme Outcomes (POs) by



9. defining a CO:PO articulation matrix for every course.
10. The methodology for computing the attainment levels for the Course Outcomes and Programme Outcomes is laid out.
11. The new curriculum has a focus on the problem solving and learning capabilities of the students. There are many laboratory courses which give students a hands-on experience in problem solving. Further, provisions for industry internship/training and project works make students ready to accept challenges and do research to solve difficult engineering problems.
12. Overall, the curriculum is made keeping in the view the continuous cycle of improvement in teaching learning process.



Vision

To be known as University having capacity to transform the society through continual innovation in education, research, creativity and entrepreneurship in the domain of higher education .

Mission

The overall focus of the University will be to create transformative experience for the students so as to enable them to realize their full potential. Towards this end following shall be the focal areas:

- To create an environment in which new ideas, research and scholarship flourish and from which the leaders and innovators of tomorrow emerge.
- To use local resources and environment as testing laboratory by faculty and students for generating new ideas and solution of local problems.
- To provide a challenging and rigorous curriculum which is global in outlook and simultaneously provides an understanding of the needs of society and industry.
- To collaborate with leading academic and research institutes around the world to strengthen the education and research ecosystem.
- To integrate ICT into teaching-learning processes and administrative systems for enhancing their effectiveness.
- To create forums to encourage students to participate in curricular and extra-curricular activities to showcase and hone their talent and creativity.



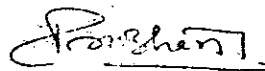
DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

VISION

To advance in the evolving domains of Computer Science by imparting knowledge, fostering relevant practices, and instilling human values to empower students as capable contributors in innovative applications of advanced computing.

MISSION

- Foster strong foundational and technical skills in Computer Science through effective teaching methodologies.
- Cultivate ethical values, creativity, and entrepreneurial spirit among students to inspire them to establish startups.
- Encourage students to prioritize sustainable solutions that enhance quality of life and societal welfare.
- Strengthen research in computing by fostering collaborations between academia and industry.
- Promote lifelong learning of emerging technologies to encourage students to pursue higher studies and continuous professional development.



ABOUT THE PROGRAMME

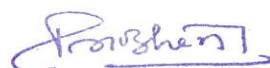
The Department of Information and Communication Technology was established in 2019 to prepare next-gen IT professionals who are competent enough to take up industry challenges effectively. The department intends to provide cutting-edge education ranging from expertise in traditional software development to modern computing technologies. The department is equipped with state-of-the-art laboratories, industry-aligned curriculum and association with top IT companies to provide students education which meets the need of the industry.

From the academic session 2019-20, M.Sc. (Computer Science) programme has been launched with an intake of 30 students. The programme was started with three specializations of Data Analytics, Cyber Security and Networking keeping in mind the current requirements and scope of the Information Technology and to acquaint students with the latest concept of Artificial Intelligence and Internet of Things.

An exit scheme for students of M.Sc. Computer Science has been implemented w.e.f. 2020-2021 in which a student of M.Sc. Computer Science (2-Year) of the University Teaching Department and Affiliated Colleges of the University may exit the programme after completion of the first year in all respect and after that, he/she may be granted a One-year diploma certificate i.e. Post Graduate Diploma in Computer Applications (PGDCA). The programme of such candidate may be considered as Post Graduate Diploma in Computer Applications (PGDCA).

NOTE:

The scheme will be applicable from Academic Session 2026-27 onwards.



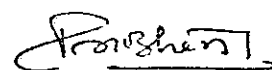
MCA PROGRAMME

PROGRAMME EDUCATION OBJECTIVES

PEO1	To develop the ability to plan, analyze, design, code, test, implement and maintain the software product for real time systems.
PEO2	To excel in problem solving and programming skills in computing fields of IT industries.
PEO3	To encourage students capability to set up their own enterprise in various sectors of Computer Applications.
PEO4	To prepare the students to pursue higher studies in computing and related fields and to work in the fields of teaching and research.

PROGRAMME OUTCOMES

PO1	Computational Knowledge: Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
PO2	Problem Analysis: Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
PO3	Design /Development of Solutions: Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
PO4	Conduct Investigations of Complex Computing Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	Modern Tool Usage: Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
PO6	Professional Ethics: Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.
PO7	Life-long Learning: Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.
PO8	Project management and finance: Demonstrate knowledge and understanding of the



	computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO9	Communication Efficacy: Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.
PO10	Societal and Environmental Concern: Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.
PO11	Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.
PO12	Innovation and Entrepreneurship: Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

PROGRAMME SPECIFIC OUTCOMES

PSO1	Understand and apply the computing techniques with mathematics and industrial concepts for solving the real time industrial problems
PSO2	Analyze, design, develop, test and maintain the software applications with latest computing tools and technologies

Prashant

**Semester wise Structure
and Curriculum for
PG Course in
Master in Computer
Applications (MCA)**

Scheme and Syllabus of Two Year Post Graduate Programme M.C.A. w.e.f. 2026-27

Courses	Nomenclature of Course	Course Code	Credits Distribution			Total Credits	Total Contact Hours			Marks				Total Marks
							L	T	P	Theory		Practical		
										Internal	External	Internal	External	
SEMESTER-I														
CC-1@4credits	Artificial Intelligent and its Applications	26MCA101	4	0	0	4	4	0	0	30	70	-	-	100
CC-2@4credits	Computer Fundamentals and Programming in C	26MCA102	3	0	1	4	3	0	2	20	50	10	20	100
CC-3@4credits	Web Designing Fundamentals	26MCA103	3	0	1	4	3	0	2	20	50	10	20	100
CC-4@4credits	Database management system	26MCA104	3	0	1	4	3	0	2	20	50	10	20	100
PC-1@3credits	Advance Office Management Tools	26MCA105	0	0	3	3	0	0	6	-	-	20	50	70
Anyone of the following														
DEC-1@4 credits	Computer Architecture	26MCA106	4	0	0	4	4	0	0	30	70	-	-	100
	Design and Analysis of Algorithms	26MCA107	4	0	0	4	4	0	0	30	70	-	-	100
SEMESTER-II														
CC-5@4credits	Discrete Mathematics	26MCA201	4	0	0	4	4	0	0	30	70	-	-	100
CC-6@4 credits	Data Structures and Algorithms	26MCA202	3	0	1	4	3	0	2	20	50	10	20	100
CC-7@4 credits	Object oriented programming using Java	26MCA203	3	0	1	4	3	0	2	20	50	10	20	100
CC-8@4 credits	System Software and Operating Systems	26MCA204	3	0	1	4	3	0	2	20	50	10	20	100
PC-2@3 credits	Programming in MATLAB	26MCA205	0	0	3	3	0	0	6	-	-	20	50	70
IKS/Computer C@2credits	One from the pool	26MCA206	2	0	0	2	2	0	0	15	35	-	-	50
Any one of the following														
DEC-2@4 credits	Internet of Things	26MCA207	4	0	0	4	4	0	0	30	70	-	-	100
	Theory of Computation	26MCA208	4	0	0	4	4	0	0	30	70	-	-	100

Prashant

Master of Computer Applications (MCA)

Bridge Course Course

Course No./ Course Name	Course Type	Workload Hours/ Week	Credits	Internal Marks	External Marks	Total Marks
MCABC	Theory	3	3	20	50	70

Note:

1. This course is qualifying in nature. However, no student will be awarded MCA degree without qualifying this.
2. For students taking admission to MCA course and who have graduated without MATHEMATICS.
3. This bridge courses shall be completed by the student(s) as prescribed by the Department/University. MCA degree shall not be awarded unless the students successfully complete the bridge course. Bridge course examination will be conducted by the University. The student has to secure 40% marks in examination in order to pass the bridge course.



MCA Bridge Course	Mathematics	Course Code	MCABC
Maximum Marks	Theory:20(Internal)+50(External)=70	Time of Examination	3 Hours
Note: Examiner will set seven questions and the candidates will be required to attempt four questions in all. Question number one will be compulsory containing five short answer type questions from all Sections. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section.			
Course Learning Outcomes(CLO): The students are expected to be able to understand fundamentals of computer and basic Mathematics.			
Section-I Set Introduction, Objectives, Representation of Sets (Roster Method, Set Builder Method), Types of Sets (Null Set, Singleton Set, Finite Set, Infinite Set, Equal Set, Equivalent Set, Disjoint Set, Subset, Proper Subset, Power Set, Universal Set) and Operation with Sets (Union of Set, Intersection of Set, Difference of Set, Symmetric Difference of Set) Universal Sets, Complement of a Set.			
Section-II Logic Statement, Connectives, Basic Logic Operations (Conjunction, Disjunction, Negation) Logical Equivalence/Equivalent Statements, Tautologies and Contradictions. Progressions: Introduction, Arithmetic Progression, Sum of Finite number of quantities in A.P, Arithmetic Means, Geometric Progression, Geometric Mean.			
Section-III Matrices Introduction, Types of Matrix (Row Matrix, Column Matrix, Rectangular Matrix, Square Matrix, Diagonal Matrix, Scalar Matrix, Unit Matrix, Null Matrix, Comparable Matrix, Equal Matrix), Scalar Multiplication, Negative of Matrix, Addition of Matrix, Difference of two Matrix, Multiplication of Matrices, Transpose of a Matrix.			
Reference Books: 1. Elementary Mathematics, Dr. RD Sharma 2. Comprehensive Mathematics, Parmanand Gupta 3. Elements of Mathematics, ML Bhargava			

Parshant

Semester-I

Name of the Course(CC1) 4 credits	Artificial Intelligent and its Applications	Course Code	26MCA101
Maximum Marks	Theory:30(Internal)+70(External)=100	Time of Examination	3 Hours
Note: Examiner will set nine questions and the candidates will be required to attempt five questions in all. Question number one will be compulsory containing seven short answer type questions from all Sections of 2 marks each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section. All questions will carry equal marks.			
Course Learning Outcomes(CLO): CO1 - Understand the Basics about Artificial Intelligence and Expert Systems. CO2 - Understand the Programming Logics in Artificial Intelligence. CO3 - Understand Various search methods in Artificial Intelligence. CO4 - Understand the Image processing and analysis. CO5 - Understand the latest developments in Knowledge systems and Tools.			
Section-I Introduction to Artificial Intelligence: Artificial Intelligence, How Does AI Work?, Advantages and Disadvantages of Artificial Intelligence, History of Artificial Intelligence, Goals of AI, Turing Test, Types of Artificial Intelligence, Weak AI, Strong AI, Machine Learning and Deep Learning. Expert System: Definition, role of knowledge, architecture and life cycle of Expert System.			
Section-II Knowledge & Its Representation: Types of knowledge, Knowledge acquisition and its techniques, Knowledge engineering, Cognitive behavior; Knowledge representation: Level of representation; Knowledge representation schemes: Formal logic, Inference Engine, Semantic net, Frame, Scripts. Search Strategies: Introduction, Brute Force or Blind Search, Breadth-First Search, Depth-First Search, Hill Climbing, Best-First Search. Perception: Sensing, Speech recognition, Vision, Action.			
Section-III Computational Intelligence: Introduction to Computational Intelligence, Biological and Artificial Neural Network (ANN), artificial neural network models; learning in artificial neural networks; neural network and its applications. Evolutionary Computation: Fundamentals of evolutionary computation, Design and Analysis of Genetic Algorithms, Evolutionary Strategies, comparison of GA and traditional search methods. Genetic Operators and Parameters, Genetic Algorithms in Problem Solving; Optimization: Particle Swarm Optimization, Ant Colony Optimization, Artificial Immune Systems; Other Algorithms: Harmony Search, Honey-Bee Optimization, Memetic Algorithms, Co-Evolution, Multi-Objective Optimization, Tabu Search, Constraint Handling.			
Section-IV Trends in AI: AI and Ethical Concerns, AI as a Service (AIaaS), Recent trends in AI, Expert System, Internet of Things, Artificial Intelligence of Things (AIoT). Industrial Applications of AI: Application of AI in Healthcare, Application of AI in Finance, Application of AI in Retail, Application of AI in Agriculture, Application of AI in Education, Application of AI in Transportation, AI in Experimentation and Multi-disciplinary research.			
References: 1. Artificial Intelligence: A Modern Approach by Stuart Russell and Peter Norvig 2. Computer Vision: Algorithms and Applications by Richard Szeliski 3. Reinforcement Learning: An Introduction by Richard S. Sutton and Andrew G. Bart. 4. Artificial Intelligence in Practice by Bernard Marr 5. Reema Thareja, Artificial Intelligence: Beyond Classical AI, Pearson Education, 2023.			

Prashant

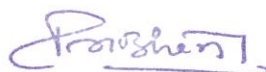
Semester-I

Name of the Course(CC2) 3 credits	Computer Fundamentals and Programming in C	Course Code	26MCA102
Maximum Marks	Theory:20(Internal)+50(External)=70	Time of Examination	3 Hours
Note: Examiner will set seven questions and the candidates will be required to attempt four questions in all. Question number one will be compulsory containing eight short answer type questions from all Sections of one mark each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section.			
Course Learning Outcomes(CLO): CO1: Learn the functional units and classify types of computers, their applications and effects. CO2: Understand system softwares and their working. CO3: Understand the logic building used in programming. CO4: Design and develop algorithms for solving various real-life problems. CO5: Design and develop programs using C.			
Section-I Computer Fundamentals: Concept of data and information. Components of Computer. Input and Output Device, Components of CPU, Memory and Storage Devices, Classification of Computers, Advantages and Limitations of Computer, Applications of Computer, Social concerns of Computer Technology: Positive and Negative Impacts, Computer Crimes, Viruses and their remedial solutions. Computer Software: System and Application Software, Overview of Operating System Programming Languages Machine. Assembly. High Level Language, 4GL. Language Translator, Linker and Loader.			
Section-II Problem Solving: Problem Identification. Analysis, Algorithms, Flowcharts. Pseudo codes. Decision Tables. Program Coding. Program Testing and Execution. C Programming Fundamentals: Keywords, Variables and Constants, Structure of a C program. Operators & Expressions: Arithmetic, Unary, Logical. Bit-wise, Assignment& Conditional Operators. Decision Making: Decision making using if...else. Else If Ladder; Switch, break. Continue and Goto statements.			
Section-III Loops: Looping using while, do...while, for statements. Nested loops. Functions: Defining & Accessing User defined functions. Library Functions, Function Prototype, Passing Arguments, Passing array argument. Recursion, Use of Library Functions. Macro vs. Functions, Pointers in C.			
References: 1. Gill Nasib Singh:Computing Fundamentals and Programming in C, Khanna Books Publishing Co., New Delhi. 2. E. Balaguruswamy: Programming in C.Tata McGraw Hill. 3. Rajender Singh Chhillar: Application of IT to Business, Ramesh Publishers, Jaipur.			

Rajender

Semester-I

Name of the Course (CC4) 3 credits	Web Designing Fundamentals	Course Code	26MCA103
Maximum Marks	Theory:20(Internal)+50(External)=70	Time of Examination	3 Hours
Note: Examiner will set seven questions and the candidates will be required to attempt four questions in all. Question number one will be compulsory containing eight short answer type questions from all Sections of one mark each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section.			
Course Learning Outcomes(CLO): After completion of course, students would be able to understand: • The client side and server side scripts using programming • The basic concept of designing websites • Database connectivity with the webpages			
Section-I Introduction to Web programming – Overview of HTML-basic formatting tags -heading, paragraph, underline break, bold, italic, underline, superscript, subscript, font and image. Attributes - align, color, bgcolor, font face, border, size. Navigation Links using anchor tag-internal, external, mail and image links. Lists ordered, unordered and definition, Table tag, HTML Form controls-form, text, password, text area, button, checkbox, radio button, select box, hidden controls.			
Section-II Cascading Style Sheets: Introduction, Inline, Internal, External CSS, Linking CSS to Web Page. Client-Side Programming: Introduction to JavaScript, Basic Syntax, Variables and Data types, Statements, Operators, Literals, Functions, Objects, Arrays. XML: Relation between XML and HTML, Goals of XML, Structure and Syntax of XML, Well Formed XML, DTD and its Structure, tree structures in data organization, Searching with XPath.			
Section-III Apache Web Server: Introduction, architecture, features, Apache's Role in the Internet, LAMP, WAMP, XAMPP. Serversideprogramming –PHP–DesigningdynamicwebpagesusingPHP–DefiningPHP variables –variable types –operators –control flow constructs in PHP –passing form data between pages - Establishing connection with MySQL database – managing database			
References: 1. DickOliver: Tech Yourself HTML4 in 24Hours,Techmedia. 2. Satish Jain:"O"-Level Information Technology. 3. Achyut Godbole, "Web Technologies", Tata McGraw Hill, India. 4. CraigZacker: 10minutes Guide to HTML Style Sheets, PHI. 5. V.K.Jain:"O"-Level Information Technology, BPB Publications.			



Semester-I

Name of the Course (CC4) 3 credits	Database management and management	Course Code	26MCA104
Maximum Marks	Theory:20(Internal)+50(External)=70	Time of Examination	3 Hours
Note: Examiner will set seven questions and the candidates will be required to attempt four questions in all. Question number one will be compulsory containing eight short answer type questions from all Sections of one mark each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section.			
Course Learning Outcomes(CLO): • Express the basic concepts of DBMS and RDBMS. • Apply normalization theory to the normalization of a database • Apply the concept of Transaction Management & Recovery techniques in RDBMS. • Analyse various concurrency control and recovery techniques.			
Section-I Database System Concepts and Architecture: Traditional File Processing System vs DBMS, Characteristics & Advantages of DBMS, Three-Schema Architecture and Data Independence; Data Models, Schemas, and Instances; Database Languages and Interfaces; Classification of DBMS. Data Modeling: Overview of Entity-Relationship Diagram, Relational Model -Constraints, Relational Database Schemas, Relational Algebra and Relational Calculus; CoddRules.			
Section-II Normalization for Relational Databases: Functional Dependencies and Normalization; SQL: SQL as 4GL, SQL Components: DDL, DML, DQL, DCL, TCL; Data Definition and Data Types; Constraints, Queries, Insert, Delete, and Update Statements; Views, Stored Procedures and Functions; Database Triggers, SQL Injection. Query Processing and Optimization: Translating SQL queries into Relational Algebra, Basic Algorithm for Executing Query Operations, Using Heuristic in Query Optimization, Using Selectivity and Cost Estimation in Query Optimization, Semantic Query Optimization.			
Section-III Transaction Processing: Introduction, Desirable properties of Transactions, Schedules & Recoverability, Serialization of Schedulers, Transaction Support in SQL. Concurrency Control Techniques: Locking techniques for Concurrency Control, Concurrency Control based on Timestamp ordering, Multiversion Concurrency Control Techniques, Validation Currency Control Techniques, Granularity of data items and multiple granularities locking, Using locks for Concurrency Control in Indexes. Database Recovery Techniques: Basic Concepts, Recovery Technique based on Deferred Update, Recovery Technique based on Immediate Update, Shadow Paging, The ARIES recovery algorithm, Database backup and recovery from catastrophic failure.			
References: 1. Elmasri & Navathe: Fundamentals of Database Systems, Sth edition, Pearson Education. 2. C.J.Date: An Introduction to Database Systems, 8th edition, Addison Wesley N. Delhi. 3. Thomas Connolly, Carolyn Begg: Database Systems, Pearson Education. 4. Any other book(s) covering the contents of the paper in more depth.			

Prashant

Semester-I

Name of the Course (PC1) 3 credits	Advance Office Management Tools	Course Code	26MCA105
Maximum Marks	Practical:20(Internal)+50(External)=70		
Course Learning Outcomes(CLO): • Understand concept of Word Processor and use its features. • To use the advanced features of Ms-Word to make day to day usage easier. • To work comfortably with Ms-Excel Environment. To create work sheets and user advanced feature of Excel. • To create make presentations and inserting multimedia in them.			
Section: I Introduction to MS Office & MS Word: MS-Word: Features of MS-Word, MS-Word Window components, working with formatted text, Shortcut keys, formatting documents: Selecting text, Copying & moving data, formatting characters, changing cases, Paragraph formatting, Indents, Drop Caps, using format painter, Page formatting, Bullets & numbering, Tabs, Forming tables. Finding & replacing text. Overview MS Word Advanced features: Header & footer, Macro, Tables, Mail merging, Printing documents, Tables: Insert tables, Charts			
Section:2 MS-Excel: Excel Features, Spreadsheets, cell, cell referencing, workbooks, creating, saving & edit workbook, Renaming sheet, spell check, Adding and deleting rows and columns, Sorting, Filtering, Advance Filter, Conditional Formatting, Charts, Pivot Table and Chart, conditional formatting, tables			
Section:3 MS-PowerPoint: Features of Power Point, Uses, components of power point window, designing slides, slide layouts, customizing slides through slide master, changing colour scheme, changing background and shading, adding header and footer, Inserting picture, shapes, charts, audio, video into slide. Adding transition and animations, Working in slide sorter view (deleting, duplicating, rearranging slides),slide show			
References: 1. Fundamentals of Computer–V. Rajaraman, Prentice Hall of India. 2. Introduction to Computers–Peter Norton McGraw-Hill.			



Semester-I

Name of the Course(DEC-1) 4 credits	Computer Architecture	Course Code	26MCA106
Maximum Marks	Theory: 30(Internal)+70(External)=100	Time of Examination	3 Hours

Note:

Examiner will set nine questions and the candidates will be required to attempt five questions in all. Question number one will be compulsory containing seven short answer type questions from all Sections of 2 marks each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section. All questions will carry equal marks..

Course Learning Outcomes(CLO):

- Study of the basic structure and operation of a digital computer system.
- Analysis of the design of arithmetic & logic unit and understanding of the fixed point and floating point arithmetic operations.
- Implementation of control unit techniques and the concept of Pipelining
- Understanding the hierarchical memory system, cache memories and virtual memory

Section: I

Number System: Number System: Binary, Octal, Hexadecimal, and Decimal; 1's and 2's Complements; Inter conversion of Numbers.

Codes: Weighted and Non-weighted Codes, BCD Codes, Gray Codes, Self-complementing Codes, Error-Detecting/Correcting Codes, Alphanumeric Codes, Hamming Codes.

Floating Point Numbers: Binary Arithmetic, Binary Addition and Subtraction, 2's Complement Arithmetic, Booth Coding, Binary Multiplication.

Logic Design: Logic Gates, Truth Tables, Boolean Algebra, Boolean Expressions- Variables and Literals, Equivalent and Complement of Boolean Expressions, Theorems of Boolean Algebra, Simplification Techniques, SOPs & POSs of Boolean Expressions.

Section:2

Combinational Circuits: Combinational Logic, Arithmetic Circuits - Adder and Subtractor, BCD Adder, Code Converters, Magnitude Comparator, Parity Generators/Checkers, Multiplexers, Demultiplexers, Decoders, Encoders.

Sequential Circuits: Latches, RS Flip-Flop, Level-Triggered and Edge-Triggered Flip-Flops, JK Flip-Flop, Master-Slave Flip-Flops, T Flip-Flop, D Flip-Flops.

Section:3

Basic Computer Design: Computer Instructions and Types, Instruction Set, Instruction Cycle, Instruction Formats, Addressing Modes, Computer Registers, Bus System, Register Transfer Language Terminology.

CPU Design: CPU Registers, Micro-operations and Types, Design of ALU, Control Unit Design - Microprograms, Control Unit of a Basic Computer, Timing and Control: Hardwired and Micro-programd Controlled Unit, Architectures - RISC, CISC, Scalar, Superscalar, and Pipelined Architectures.

Prashant

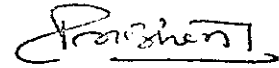
Section:4

Input/Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Mode of Transfer, Priority Interrupt, Direct Memory Access, Input-Output Processor, Serial Communication.

Advanced Architecture: Introduction to Parallel Processing - Pipelining, Parallel Computer Structures, Architectural Classification, Pipelining & Vector Processing; Instruction and Arithmetic Pipelines, Principles of Designing Pipelined Processors, Structures for Array Processors - SIMD Array Processor, SIMD Interconnection Networks, Parallel Processing Applications.

References:

1. Mano, M. M. Digital Logic and Computer Design. Prentice-Hall of India.
 2. Gill, Nasib Singh and Dixit, J. B. Digital Design and Computer Organization. University Science Press (Laxmi Publications), New Delhi.
 3. Stallings, William. Computer Organization & Architecture.
 4. Anand Kumar. Fundamentals of Digital Circuits. PHI.
 5. Hwang, Kai. Advanced Computer Architecture. McGraw-Hill International.
 6. Mano, M. M. Computer System Architecture. Prentice-Hall of India.
- Tokheim. Digital Electronics. TMH.



Semester-I

Name of the Course(DEC-2)4 credits	DESIGN AND ANALYSIS OF ALGORITHMS	Course Code	26MCA107
Maximum Marks	Theory:30(Internal)+70(External)=100	Time of Examination	3 Hours
Note: Examiner will set nine questions and the candidates will be required to attempt five questions in all. Question number one will be compulsory containing seven short answer type questions from all Sections of 2 marks each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section. All questions will carry equal marks.			
Course Learning Outcomes(CLO): - To learn the basic concept of the asymptotic performance of algorithms. - To understand the writerigorous correctness proofs for algorithms. - Demonstrate familiarity with major algorithms and data structures and apply them to solve problems.			
Section: I Introduction to analysis of algorithms: Analysis of algorithms, asymptotic notation-Big- O, Omega and Theta notations, recurrence relations, solving recurrences, Abstract data types, Linear Data Structures and their sequential storage representation: stacks, queues, priority queues, and their applications.			
Section: 2 Divide and Conquer: General method, Binary Search, Exponentiation problem, MergeSort, Quick Sort, Selection Sort, Strassen's Matrix Multiplication algorithms and analysis of algorithms for these problems. Greedy Method: General method, Knapsack Problem, Job sequencing with deadlines, Minimum Spanning trees, Single source shortest path and analysis of these algorithms			
Section:3 Dynamic Programming: General method, 0/1 Knapsack problem, Optimal BST, All Pairs shortest path, Traveling Salesman Problem, longest common subsequence (LCS). Back Tracking:Generalmethod,8queen'sproblem,graphcoloring,Hamiltoniancyclesand Analysis of these problems.			
Section:4 NP-HardandNP-CompleteProblems:P,NP,NP-Hard&NPCompleteClasses,Reductions: Vertex cover, Simple Max-Cut, Hamiltonian Circuit, Traveling salesman problem, kernel, 3- dimensional matching, and other NP-Complete Problems, Satisfiability and variations, Cook's theorem, examples of NP-Hard problems.			
References: - Cormen, T.H., Leiserson, C.E., Rivest, R.L.andStein, C., - Introduction to Algorithms, MIT Press, 2010. - Goodrich, M.T., Tamassia, R. and Mount, D.M., Data Structures and Algorithms in C++, John Wiley & Sons, 2016. - Langsam, Y., Augenstein, M. and Tenenbaum, A.M., DataStructuresusingCandC++, Prentice Hall, 2015. - Lipschutz, S., Schaum's Outline of Theory and Problems of Data Structures, McGraw-Hill, 2014			

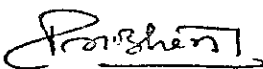
Forshant

Semester-I

Name of the Course(1credit)	PRACTICAL(BASED ON 26MCA102)	Course Code	26MCA102P
Maximum Marks	Practical:10(Internal)+20(External)=30		

List of Practical:

1. Write a C program to compute roots of quadratic equation $ax^2+bx+c=0$, where a, b, and c are three coefficients of a quadratic equation are inputs.
2. Design and develop an algorithm to find the reverse of an integer number.
3. Design and develop an algorithm to check whether given number is PALINDROME or NOT. Implement a C program for the developed algorithm that takes an integer number as input and output the reverse of the same with suitable messages. Ex: Num: 2019, Reverse: 9102, Not a Palindrome.
4. Design and develop a c program to implement simple calculator using switch case statement.
5. Develop, implement and execute a C program to search a Number in a list using linear searching Technique.
6. Develop an algorithm, implement and execute a C program that reads N integer numbers and arrange them in ascending order using Bubble Sort.
7. Design and develop a C program to read and print a matrix and check whether a given Matrix is a sparse Matrix or not.
8. Write a C program to implements the following string manipulation functions till the use wishes to continue (infinite loop): (i) strcpy() (ii) strlen() (iii) strrev () (iv) strcmp() (v) strcat().
9. Design and develop a C function RightRotate (x, n) that takes two integers x and n as input and returns value of the integer x rotated to the right by n positions. Assume the integers are unsigned.
10. a. Write a C program to maintain a record of n student details using an array of structures with four fields (Roll number, Name, Marks, and Grade). Assume appropriate data type for each field. Input & Print the members of the structure
b. Write a C program to take 2 structures HH:MM: SS as T1 & T2 & display the time difference as structure as T3.
11. Write a C program using pointers to compute the sum, mean and standard deviation of all elements stored in an array of n real numbers.

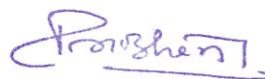


Semester-I

Name of the Course (1 credit)	PRACTICAL(BASED ON 26MCA103)	Course Code	26MCA103P
Maximum Marks	Practical:10(Internal)+20(External)=30		

List of Practical:

1. Create a simple HTML page with title, heading, and paragraph, formatting tags, hyperlinks, list items and image elements.
2. Create a simple HTML page having image elements with the use of map.
3. Create a simple HTML page having a complex table.
4. Create a simple HTML page having multiple Frames.
5. Embed Video in a HTML page.
6. Create a simple HTML page that uses Inline CSS
7. Create a simple HTML page that uses Document level CSS
8. Create a simple HTML page that uses External level CSS
9. Create a HTML page that uses all CSS properties Contents Tasks List Description
10. Write a simple JavaScript to print text on to the HTML document.
11. Write a simple script in head portion of HTML document.
12. Write a simple script in body portion of the HTML document.
13. Write an external script and link it to the HTML document.
14. Write a PHP script to get the PHP version and configuration information
15. Write a PHP script to display any multi-line string
16. Write a PHP script to place a variable to a title and as hyperlink of the Web page
17. Create a simple HTML form and accept the user name and display the name through PHP echo statement.

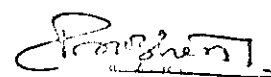


Semester-I

Name of the Course (1credit)	PRACTICAL(BASED ON 26MCA104)	Course Code	26MCA104P
Maximum Marks	Practical:10(Internal)+20(External)=30		

List of Practical:

1. Creation of a database and writing SQL queries to retrieve information from the database.
2. Performing Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions.
3. Creation of Views, synonyms, Sequences, indexes, and save points.
4. Creating an employee database to set various constraints.
5. Mini project (Any 2 Application Development using MySQL)
 - a. Inventory Control System
 - b. Material Requirement Processing.
 - c. Hospital Management System.
 - d. Railway Reservation System.
 - e. Personal Information System.
 - f. Web-Based User Identification System.
 - g. Time Table Management System.
 - h. Hotel Management



Semester-2

Name of the Course (CC5) (4 credits)	Discrete Mathematics	Course Code	26MCA201
Maximum Marks	Theory: 30(Internal)+70(External)=100	Time of Examination	3 Hours

Note:

Examiner will set nine questions and the candidates will be required to attempt five questions in all. Question number one will be compulsory containing seven short answer type questions from all Sections of 2 marks each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section. All questions will carry equal marks.

Course Outcomes

CO1: Solve mathematical problems based on concepts of set theory, relations, functions and lattices.

CO2: Express logical sentences in terms of quantifiers and logical connectives.

CO3: Apply basic counting techniques to solve permutation and combination problems.

CO4: Solve recurrence relations.

CO5: Develop the given problem as graph networks and solve it with techniques of graph theory.

Section-I

Set Theory: Introduction to set theory, Venn diagrams, Set operations, Algebra of sets, Duality, Finite and infinite sets, Counting principles, Power sets, Partitions, and Multi sets.

Relations: Cartesian product, Representation of relations, Types of relation, Binary relation, Equivalence relations, Partitions, Partial ordering relations, POSET, Hasse diagram, Lattices and its types.

Functions: Definition, Types of functions, Bijective functions, Composition of functions, Inverse functions, recursively defined functions, Finite and infinite sets, Countable and uncountable sets, Cantor's diagonal argument and The Power Set theorem, Schroeder-Bernstein theorem.

Section-II

Propositional Logic: Proposition logic, basic logic, Logical Connectives, truth tables, tautologies, contradiction, Logical implication, Logical equivalence, Normal forms, Theory of Inference and deduction.

Predicate Calculus: Predicates and quantifiers. Mathematical Induction.

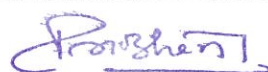
Matrices: Definition, Types of Matrices, Addition, Subtraction, Scalar Multiplication and Multiplication of Matrices, Adjoint and Inverse of a matrix. Determinants: Definition, Minors, Cofactors, Properties of Determinants, Applications of determinants in finding area of triangle, solving a system of linear equations. .

Section-III

Graphs Theory: Introduction to graphs and their properties, Degree, Connectivity, Path, Cycle, Directed and undirected graphs, Subgraph, Bipartite Graphs, Regular Graphs, Connected Graphs, Multigraph and Weighted graph, Homomorphic and Isomorphic graphs, cut points and bridges, Paths and circuits, shortest path algorithm for weighted graphs, Eulerian paths and circuits, Hamiltonian path and circuits, Planar Graphs, Euler's formulae, Graph Colouring.

Section-IV

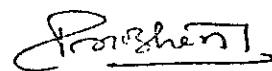
Introduction to Probability Theory: Sample space and events, random variables, probabilities of events and combinations of events, conditional probability, expectation, mean and variance of random variables. **Probability distributions:** Binomial and Poisson distributions, properties and occurrence; normal distribution, properties, examples, relation to Poisson approximation.



Semester-2

References:

1. Kenneth H. Rosen, *Discrete Mathematics and its Applications*, 6th Edition, Tata McGraw Hill, 2011.
2. Satinder Bal Gupta: *A Text Book of Discrete Mathematics and Structures*, University Science Press, Delhi.
3. C. L. Liu and D. P. Mohapatra, *Elements of Discrete Mathematics A Computer Oriented Approach*, Tata McGraw Hill, 3rd Edition, 2008.
4. J.P. Trembley and R. Manohar, *Discrete Mathematical Structures with Applications to Computer Science*, Tata McGraw Hill – 13th reprint, 2012.
5. Richard Johnsonbaugh, *Discrete Mathematics*, 6th Edition, Pearson Education Asia, 2011.
6. S. Lipschutz and M. Lipson, *Discrete Mathematics*, Tata McGraw Hill, 3rd Edition, 2010.



Semester-2

Name of the Course (CC6) (3 credits)	Data Structures and Algorithms	Course Code	26MCA202
Maximum Marks	Theory:20(Internal)+50(External)=70	Time of Examination	3 Hours

Note:

Examiner will set seven questions and the candidates will be required to attempt four questions in all. Question number one will be compulsory containing eight short answer type questions from all Sections of one mark each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section.

Course Learning Outcomes(CLO):

CO1: Choose appropriate data structures and algorithms and use it to design solution for a specific problem.

CO2: Execute the operations of hashing to retrieve data from data structure.

CO3: Comprehend and select algorithm design approaches in a problem specific manner.

CO4: Design and analyze programming problem statements.

CO5: Come up with analysis of efficiency and proofs of correctness.

Section-I

Introduction: Definition and its types, Data Types: Primitive, Composite and Abstract Data Types, Data Structures: Concept, Classification, and Importance; Data Structures v/s Data Types, Linear v/s Non- linear Data Structures.

Arrays: Single and Multidimensional arrays; Address Calculation using Column and Row major ordering; Various operations on Arrays; Vectors; Sparse Matrix; Application of Arrays; Implementation of Arrays in C.

Section-II

Stacks and Queues: Representation of stacks and queues using arrays and linked-list. Circular queues. Priority Queue and D-Queue. Applications of stacks: Conversion from infix to postfix and prefix expressions. Evaluation of postfix expression using stacks; Implementation in C.

Linked list: Singly Linked List; Operations on Linked Lists. Linked Stacks and Queues. Polynomial Representation and Manipulation using Linked Lists. Circular Linked Lists. Doubly linked lists; Implementation in C.

Searching and Sorting: Linear search, Binary search, Insertion sort, selection sort, Bubble sort, Merge sort, Quick Sort, Heap Sort; Hashing, Hash table, Hash functions,

Section-III

Trees: Concept, Representation and Applications of Trees, Forest, Binary Tree, Threaded Binary Tree; Binary tree representation of a general tree; Conversion of forest into tree; Binary search tree: Height balanced (AVL) tree, B-trees, B+Tree, B* Tree.

Binary tree traversal methods: Pre-order. In-order. Post-ordered traversal. Recursive Algorithms.

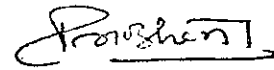
Graphs: Representation: Adjacency matrix, Adjacency lists; Type of Graphs; Paths: Euler Graphs, Hamiltonian Paths & Circuits; Cut-sets, Connectivity and Separability, Planar Graphs, Isomorphism, Graph Coloring, Covering and Partitioning, Breadth-First Search, Depth-First Search; Minimum Spanning Trees: Prim's and Kruskal's algorithms; Shortest-path Algorithms: Dijkstra's and Floyd's algorithm;

Pragathi

Semester-2

References:

1. Fundamentals of Data Structures: Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.
2. Schaum's outline series: Data Structure, TMH.
3. Tenenbaum, Langsam, Augenstein: Data Structures using C, Pearson Education.
4. E. Horowitz and S. Sahani: Fundamentals of Data Structures, Galgotia Books Source Pvt. Ltd.
5. Bala Guruswamy: Data Structures Using C, TMH.



Semester-2

Name of the Course (CC7)(3 credits)	Object oriented programming using Java	Course Code	26MCA203
Maximum Marks	Theory: Theory:20(Internal)+50(External)=70	Time of Examination	3 Hours
Note: Examiner will set seven questions and the candidates will be required to attempt four questions in all. Question number one will be compulsory containing eight short answer type questions from all Sections of one mark each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section.			
Course Learning Outcomes(CLO): CO1: able to apply object-oriented programming features and concepts for solving given problem. CO2: able to use java standard API library to write complex programs. CO3: able to implement object oriented programming concepts using java CO4: able to develop interactive programs using applets and swings.			
Section-I			
Object Oriented Methodology -Paradigms of Programming Languages, Evolution of OO Methodology, Basic Concepts of OO Approach, Comparison of Object Oriented and Procedure Oriented Approaches, Benefits of OOPs, Introduction to Common OO Language, Applications of OOPs. Object Oriented Methodology: Classes and Objects, Abstraction and Encapsulation, Inheritance, Method Overriding and Polymorphism. Java Language Basics: Introduction To Java, Basic Features, Java Virtual Machine Concepts, Primitive Data Type And Variables, Java Operators, Expressions, Statements and Arrays.			
Section-II			
Object Oriented Concepts: Class and Objects--Class Fundamentals, Creating objects ,Assigning object reference variables; Introducing Methods, Static methods, Constructors ,Overloading constructors; This Keyword; Using Objects as Parameters, Argument passing, Returning objects ,Method overloading, Garbage Collection, The Finalize ()Method. Inheritance and Polymorphism: Inheritance Basics, Access Control, Multilevel Inheritance, Method Overriding, Abstract Classes, Polymorphism, Final Keyword. Packages: Defining Package, CLASSPATH, Package naming, Accessibility of Packages, using Package Members, Interfaces: Implementing Interfaces, Interface and Abstract Classes, Extends and Implements together.			
Section-III			
Exceptions Handling: Exception, Handling of Exception, Using try-catch, Catching Multiple Exceptions, Using finally clause, Types of Exceptions, Throwing Exceptions, and Writing Exception Subclasses. Multithreading: Introduction, The Main Thread, Java Thread Model, Thread Priorities, Synchronization in Java, Interthread Communication. Strings and Characters: Fundamentals of Characters and Strings, The String Class, String Operations, Data Conversion using Value of () Methods, String Buffer Class and Methods.			
References: 1. Programming in Java, E Balagurusamy. 2. The Complete Reference JAVA, TMH Publication. 3. Beginning JAVA, Ivor Horton, WROX Public.			

Prashant

Semester-2

Name of the Course (CC 8) (3 credits)	System Software and Operating Systems	Course Code	26MCA204
Maximum Marks	Theory:20(Internal)+50(External)=70	Time of Examination	03 Hours
Note: Examiner will set seven questions and the candidates will be required to attempt four questions in all. Question number one will be compulsory containing eight short answer type questions from all Sections of one mark each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section.			
Course Learning Outcomes(CLO): CO1: Explain the basic concepts of system software and operating system. CO2: Understanding the process management policies and scheduling algorithms. CO3: Design the various memory management techniques. CO4: Understand file system concepts. CO5: Evaluate deadlock detection and prevention mechanism.			
Section-I			
Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services. Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS, SJF, SRTF, RR Scheduling.			
Section-II			
Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Semaphores, Classical IPC Problems: Reader's & Writer Problem, Dining Philosopher Problem, Producer Consumer Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, and Deadlock Avoidance: Banker's algorithm. Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK			
Section-III			
Memory Management: Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Swapping, Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging, Segmentation, Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Optimal Page Replacement and Least Recently used (LRU). File Management: Concept of File, Access methods, File types, File operation, Directory, structure, File System structure, Allocation methods (contiguous, linked, indexed), efficiency and performance.			
References: 1. Operating System Concepts Essentials, 9th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition. 2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India. 3. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing 4. Operating Systems: A Modern Perspective, 2 nd Edition by Gary J. Nutt, Addison-Wesley			

Ranishant

Semester-2

Name of the Course (PC2) (3 credits)	PROGRAMMING IN MATLAB	Course Code	26MCA205
Maximum Marks	Practical:20(Internal)+50(External)=70		
Course Learning Outcomes(CLO): • Ability to find importance of this software for Lab Experimentation. • In-depth knowledge of providing virtual instruments on MATLAB Environment. • Articulate importance of software's in research by simulation work.			
Section-I Introduction to MATLAB programming, Basic of MATLAB programming, Variables and assignments, datatypes, operators, working with numbers, mathematical operations, functions, good programming style, commands, M-files, Introduction to Vectors in MATLAB: Vector types, referencing the elements of vector, Matrix generations, Array operations and Linear Equations			
Section-II Introduction to programming in Matlab: M-File scripts, M-File functions, Colon notations Introduction to Matrices in Matlab: defining Matrix, Matrix functions, Vector operations Matrix operations; Looping and Decision Making: for loops, while loop, branching and nesting, if statement, if-else statement, else-if statement.			
Section-III Subroutine, built in function and user defined functions, handling functions in m-files; Data Files: Data import and data output, read/write, Plotting and Graphics in MATLAB: Polar plot, 2D and 3D plots, mesh, contour, Algebra, Optimization, Numerical Integration, Numerical Differentiation, solving polynomial equations, Introduction to SIMULINK.			
References: 1. Amos Gilat, MATLAB: An Introduction With Applications 5ed, Wiley, ISBN 13: 978-1118629864 2. Krister Ahlertsen, An Introduction to Matlab, Bookboon.com, ISBN: 978-87-403-0283-7 3. Stephen J. Chapman, MATLAB Programming for Engineers, Cengage Learning, 2015			

Prashant

Semester-2

Name of the Course (DEC-2@4 credits)	Internet of Things	Course Code	26MCA207
Maximum Marks	Theory: 30(Internal)+70(External)=100	Time of Examination	3 Hours
Note: Examiner will set nine questions and the candidates will be required to attempt five questions in all. Question number one will be compulsory containing seven short answer type questions from all Sections of 2 marks each. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section. All questions will carry equal marks.			
Course Learning Outcomes(CLO): CO1: Understand the concepts of Internet of Things CO2: Analyze basic protocols network CO3: Understand the concepts of Web of Things CO4: Basic Understanding of Cloud Computing. CO5: Design IoT applications in different domain and be able to analyze their performance			
Section-I			
INTRODUCTION TO IoT: Introduction to IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models & APIs, IoT & M2M Machine to Machine, Difference between IoT and M2M, Software define Network, Challenges in IoT (Design ,Development, Security), Role of Sensors and Actuators in IoT			
Section-II			
RESOURCE MANAGEMENT IN IoT: Domain specific applications of IoT, Home automation, Industry applications, Surveillance applications, Water quality, Warehouse, Retail, Driver Assistance, Collision impact, Other IoT applications Arduino and Raspberry Pi: Arduino: Architecture, Programming and Application Raspberry Pi: Architecture, Programming and Application			
Section-III			
Challenges in IoT: Design challenges, Development challenges, Security challenges, Other challenges Domain specific applications of IoT: Home automation, Industry applications, Surveillance applications, Other IoT applications Cloud for IoT: IoT and cloud, Cloud services — SaaS, PaaS, IaaS, Security in cloud			
Section-IV			
Data Analytics: Introduction, Structured Versus Unstructured Data, Data in Motion versus Data at Rest, IoT Data Analytics Challenges, Data Acquiring, Organizing in IoT/M2M, Messaging protocols (HTTP, CoAP, MQTT, AMQP, and XMPP), Transport protocols (Li-Fi, BLE), IPv4, IPv6, 6LoWPAN: forwarding, addressing, header compression, neighbor discovery, URI			
References: 1. . Vijay Madisetti, ArshdeepBahga, "Internet of Things: A Hands-On Approach" 2. WaltenegusDargie,ChristianPoellabauer, "Fundamentals of Wireless Sensor Networks: Theory and Practice 3. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, StamatisKarnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence", 1 st Edition, Academic Press, 2014. 4. Peter Waher, "Learning Internet of Things", PACKT publishing, BIRMINGHAM – MUMBAI 5. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer			

Prashant

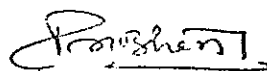
Name of the Course (DEC-2@4 credits)	Theory of Computation	Course Code	26MCA208
Maximum Marks	Theory: 30(Internal)+70(External)=100	Time of Examination	3 Hours
Note: Examiner will set nine questions and the candidates will be required to attempt five questions in all. Question number one will be compulsory containing seven short answer type questions from all Sections. Further, examiner will set two questions from each section and the candidates will be required to attempt one question from each Section. All questions will carry equal marks.			
Course Learning Outcomes(CLO): CO1: Analyze and design finite automata, formal languages, and grammars. CO2: Understand the basic concepts of DFA and NDFA. CO3: Construct context free grammar for various languages. CO4: Understand Turing Machine and recursive language. CO5: Gain understanding about tractable and non-tractable problems.			
Section-I Review of Mathematical Terms and Theory: Basic Mathematical Notations And Set Theory, Logic Functions And Relations, Language Definitions, Mathematical Inductions and Recursive Definitions. Finite Automata: Introduction, Alphabets, Strings and Languages, Kleen-closure; Deterministic Finite Automata (DFA) and Nondeterministic Finite Automata (NFA) -Formal definition, simpler notations (state transition diagram, transition table), Regular and NonRegular Languages, Equivalence of NDFA & DFA, NDFA to DFA conversion, DFA minimization using Myhill- Nerode Theorem, Applications of Finite Automata, Finite automata with output (Moore and Mealy machines) and inter-conversion			
Section-II Context Free Grammar: Introduction to CFG, CFG and Known Languages, Unions Concatenations and *'S Notations and CFL, Derivations of Trees and Ambiguity, Unambiguous CFG and Algebraic Expressions, Normal Forms and Simplified Forms. Formal Grammar: Definition, Chomsky hierarchy of grammars, Construction of Context free, derivation, parse tree, ambiguity in grammars, Removal of null and unit production, Normal forms- CNF & GNF. Pushdown Automata: Introduction to PDA, Types of PDA, Designing of PDA, CFG Corresponding to PDA, Introduction to CFL, Intersections and Complements of CFL, Decisions Problems and CFL, Equivalence of Pushdown Automata and CFL, Pumping Lemma for CFL, Applications.			
Section-III Turing Machines: Model of Computation and Church Turning Thesis, Definition of Turing Machine, Tm and Language Acceptors, Variations of Tm, Non- Deterministic Tm, Universal Tm, Tm & computers. Recursive Language: Introduction, Enumerable and Language, Recursive and Non Recursive Enumerable, their properties. PCP: Introduction to undecidability, undecidable problems about TMs, Post correspondence problem (PCP), Modified PCP.			
Section-IV Computation Functions, Measuring, Classifications and Complexity: Primitive Recursive Functions, Halting Problem, Recursive Predicates and Some Bounded Operations, Unbounded Minimizations and μ -Recursive Functions, Godel Numbering, Computable Functions and μ -Recursive, Numerical Functions. Tractable and Intractable Problems: Growth Rate and Functions, Time and Speed Complexity, Complexity Classes, Tractable and Possibly Intractable Problems, P And NP Completeness, Reduction Of Time, Cook's Theorem, NP-Complete Problems.			
References: 1. John C. Martin: Introduction to Language and theory of Computation, Mcgraw Hill. 2. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman: Introduction to Automata Theory Languages and Computation, Pearson Education 3. K. L. P Mishra, N. Chandrashekar: Theory of Computer Science-Automata Languages and Computation, Prentice Hall of India, India. 4. K.Krithivasan and R.Rama: Introduction to Formal Languages, Automata Theory and Computation; Pearson Education.			

Krishna

Name of the Course (1credit)	PRACTICAL(BASEDON 26MCA202)	Course Code	26MCA202P
Maximum Marks	Practical:10(Internal)+20(External)=30		

List of Practical:

1. Write a program to search an element in a two-dimensional array using linear search.
2. Using iteration & recursion concepts write programs for finding the element in the array using Binary Search Method
3. Write a program to perform following operations on tables using functions only a) Addition b) Subtraction c) Multiplication d) Transpose
4. Using iteration & recursion concepts write the programs for Quick Sort Technique
5. Write a program to implement the various operations on string such as length of string concatenation, reverse of a string & copy of a string to another.
6. Write a program for swapping of two numbers using 'call by value' and 'call by reference' strategies.
7. Write a program to implement binary search tree. (Insertion and Deletion in Binary search Tree)
8. Write a program to create a linked list & perform operations such as insert, delete, update, reverse in the link list
9. Write the program for implementation of a file and performing operations such as insert, delete, update a record in the file.
10. Create a linked list and perform the following operations on it a) add a node b) Delete a node
11. Write a program to simulate the various searching & sorting algorithms and compare their timings for a list of 1000 elements.
12. Write a program to simulate the various graph traversing algorithms.
13. Write a program which simulates the various tree traversal algorithms.



Name of the Course (1credit)	PRACTICAL(BASED ON 26MCA203)	Course Code	26MCA203P
Maximum Marks	Practical:10(Internal)+20(External)=30		

List of Practical:

- 1 A) Write a java program to find the Fibonacci series using recursive and non-recursive functions
 - B) Write a java program to multiply two given matrices.
 - C) Write a java program for Method overloading and Constructor overloading
- 2 A) Write a program to demonstrate execution of static blocks, static variables & static methods.
 - B) Write a program to display the employee details using Scanner class
 - C) Write a program for sorting a given list of names in ascending order
- 3 A) Write a program to implement single and Multi-level inheritance
 - B) Write a program to implement Hierarchical Inheritance.
 - C) Write a program to implement method overriding.
- 4 A) Write a program to create an abstract class named Shape that contains two integers and an empty method named printArea (). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method printArea () that prints the area of the given shape.
 - B) Write a program to implement Interface.
 - C) Write a program to implement multiple and Hybrid Inheritance
- 5 A) Write a program to create inner classes
 - B) Write a program to create user defined package and demonstrate various access modifiers.
 - C) Write a program to demonstrate the use of super and final keywords.
- 6 A) Write a program if number is less than 10 and greater than 50 it generate the exception out of range. else it displays the square of number.
 - B) Write a program with multiple catch Statements.
 - C) Write a program to implement nested try
- 7 A) Write a Program to implement simple Thread by extending Thread class and implementing runnable interface.
 - B) Write a program that implements a multi-thread application that has three threads
 - C) Write a program to set and print thread priorities
- 8 Write a program to implement following collections
 - a) array List
 - b) Vector

Prasanth

c) Hash table

d) Stack

9 A) Write a program to demonstrate lambda expressions.

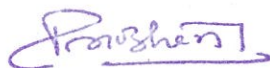
B) Write a program for producer and consumer problem using Threads

10 A) Write a program to list all the files in a directory including the files present in all its subdirectories.

B) Write a Program to Read the Content of a File Line by Line

Prashant

Name of the Course (1credit)	PRACTICAL(BASED ON 26MCA204)	Course Code	26MCA204P
Maximum Marks	Practical:10(Internal)+20(External)=30		
<u>List of Practical:</u> 1.Basics of UNIX 2.UNIX vs. Windows operating system 3.UNIX commands 4.Shell in Unix 5.Basic Unix Commands 6.File Manipulation Commands 7. Unix Editors			



Programme Learning Outcomes (PLOs) for MCA Course

1. Knowledge and Understanding: Evaluate core and advanced concepts in computing such as algorithms, system design, artificial intelligence, and data-driven systems, with the ability to interpret emerging trends and technologies.
2. Technical Skills: Design, implement, and evaluate software systems using modern computing tools, programming languages, and best practices in system development and deployment.
3. Application of Knowledge and Skills: Integrate computational knowledge with domain-specific problems to build and deploy practical solutions across sectors such as healthcare, environment, education, and business.
4. Communication Skills: Demonstrate the ability to effectively articulate technical ideas, research findings, and project outcomes in written, oral, and visual formats for diverse stakeholders.
5. Critical Thinking: Analyze complex technical problems, synthesize information from multiple sources, and apply logical reasoning to develop, test, and refine effective solutions.
6. Ethics: Apply ethical principles, legal guidelines, and professional standards in the design, implementation, and evaluation of computing technologies.
7. Life-long Learning: Pursue continuous professional growth and knowledge enhancement through research, certifications, conferences, and scholarly engagement.
8. Creativity: Design innovative computing models and user-centric solutions by applying creativity in algorithm development, interface design, and system architecture.
9. Research Aptitude: Formulate research problems, conduct literature reviews, design experimental setups, and evaluate results using appropriate scientific methods and tools.
10. Problem Solving: Identify real-world challenges, define computational problems clearly, and develop optimized, scalable solutions using algorithmic thinking and systematic evaluation.

Course Learning Outcomes (CLO) for MCA

1. Programming Proficiency

- Develop skills in programming languages like C, C++, Java, and Python.
- Write, debug, and optimize code for real-world applications.
- Apply logical thinking to solve computational problems.
- Understand object-oriented and functional programming concepts.

2. Data Structures and Algorithms

- Implement data structures like arrays, linked lists, stacks, queues, trees, and graphs.
- Analyse and optimize algorithms for better efficiency.
- Solve computational problems using sorting, searching, and recursion.
- Understand time and space complexity concepts.

3. Database Management Systems (DBMS)

- Learn SQL and NoSQL database management systems.
- Design and implement relational database schemas.
- Perform CRUD (Create, Read, Update, Delete) operations efficiently.
- Understand database normalization and indexing techniques.

4. Web Development

- Develop responsive web pages using HTML, CSS, and JavaScript.
- Implement front-end frameworks like React, Angular, or Vue.js.
- Work with back-end technologies such as Node.js, PHP, and Django.
- Create and manage dynamic web applications.

5. Software Engineering Principles

- Understand Software Development Life Cycle (SDLC) models.
- Apply Agile and DevOps methodologies in software projects.
- Implement software testing and debugging techniques.

- Maintain software documentation and version control systems like Git.

6. Networking and Security

- Understand OSI and TCP/IP network models.
- Learn about networking protocols like HTTP, FTP, and DNS.
- Apply cybersecurity principles to protect data and systems.
- Implement firewalls, encryption, and authentication techniques.

7. Artificial Intelligence and Machine Learning

- Understand AI and ML concepts, including supervised and unsupervised learning.
- Work with libraries like TensorFlow, Scikit-learn, and Keras.
- Implement machine learning models for data analysis and predictions.
- Explore natural language processing and computer vision applications.

8. Cloud Computing and Virtualization

- Learn about cloud platforms like AWS, Azure, and Google Cloud.
- Deploy, manage, and scale applications using cloud services.
- Understand virtualization, containers, and Kubernetes.
- Implement cloud security and compliance measures.

9. Internet of Things (IoT)

- Learn about IoT architecture, sensors, and actuators.
- Develop IoT-based applications using microcontrollers like Arduino and Raspberry Pi.
- Implement IoT communication protocols like MQTT and CoAP.
- Analyse and manage IoT data for real-time processing.

10. Professional Ethics and Communication

- Develop effective verbal and written communication skills.
- Learn teamwork, leadership, and problem-solving abilities.
- Understand ethical responsibilities in software development.
- Present technical ideas through reports and presentations.

11. Research and Innovation

- Explore emerging technologies in computer science.
- Conduct research in AI, blockchain, cybersecurity, or other domains.
- Develop innovative solutions to real-world problems.
- Participate in hackathons, coding competitions, and academic conferences.