MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE

MADANAPALLE

(UGC-AUTONOMOUS)

www.mits.ac.in

Estd 1998

DEPARTMENT OF COMPUTER APPLICATIONS

Course Structure

And

Detailed Syllabi (R22)

For the students admitted to

Master of Computer Applications from the academic year 2022-23 onwards

Vision and Mission of the Institution

Vision

To become a globally recognized research and academic institution and thereby contribute to technological and socio -economic development of the nation.

Mission

To foster a culture of excellence in research, innovation, entrepreneurship, rational thinking and civility by providing necessary resources for generation, dissemination and utilization of knowledge and in the process create an ambience for practice-based learning to the youth for success in their careers.

Vision and Mission of the Department

Vision

To be the source of producing competent computer application professionals in academic and research activities to serve the industry and society.

Mission

M1: To empower students with knowledge of computer applications through state-of-art infrastructure and curriculum.

M2: To groom students to become competent professionals in emerging technologies with industry specific programs.

M3: To inculcate ethical values, leadership and managerial skills in the students.

PROGRAMME OUTCOMES

The Programme Outcomes (POs) outline the key skills, knowledge, and competencies students must acquire upon completing the Master of Computer Applications (MCA). These outcomes aim to prepare students for successful careers in the rapidly evolving field of computer science and information technology.

PO1 (Foundation Knowledge):

Apply knowledge of mathematics, programming logic and coding fundamentals for solution architecture and problem solving.

PO2 (Problem Analysis):

Identify, review, formulate and analyse problems for primarily focussing on customer requirements using critical thinking frameworks.

PO3 (Development of Solutions):

Design, develop and investigate problems with as an innovative approach for solutions Incorporating ESG/SDG goals.

PO4 (Modern Tool Usage):

Select, adapt and apply modern computational tools such as development of algorithms with an understanding of the limitations including human biases.

PO5 (Individual and Teamwork):

Function and communicate effectively as an individual or a team leader in diverse and multidisciplinary groups. Use methodologies such as agile.

PO6 (Project Management and Finance):

Use the principles of project management such as scheduling, work breakdown structure and be conversant with the principles of Finance for profitable project management.

PO7 (Ethics):

Commit to professional ethics in managing software projects with financial aspects. Learn to use new technologies for cyber security and insulate customers from malware

PO8 (Life-long learning):

Change management skills and the ability to learn, keep up with contemporary technologies and ways of working.

PROGRAM EDUCATIONAL OBJECTIVES

The MCA graduates will be able to:

PEO1: Excel in the software industry with the application of comprehensive knowledge and skills.

PEO2: Contribute by building innovative and sustainable solutions to the problems in the IT industry.

PEO3: Become successful professionals, exhibiting social responsibility leading to lifelong learning.

MCA I Year I Semester

S.No	Course Code	Course Title	L	T	P	C
1	22MATP101	Mathematical Foundations for Computer Applications	3	1	0	4
2	22MCAP101	Python Programming	3	1	0	4
3	22MCAP102	Database Management Systems	3	1	0	4
4	22MCAP103	Computer Organization and Architecture	3	0	0	3
5	22MCAP104	Operating Systems	4	0	0	4
6	22MCAP105	Computer Networks	4	0	0	4
7	22ENGP201	Corporate Communication Laboratory	0	0	4	2
8	22MCAP201	Python Programming Laboratory	0	0	3	1.5
9	22MCAP202	Database Management Systems Laboratory	0	0	3	1.5
10		Skill Oriented Course – I	1	0	2	2
Total			**21**	**3**	**12**	**30**

MCA I Year II Semester

S.No	Course Code	Course Title	L	T	P	C
1	22MCAP106	Full Stack Web Development	4	0	0	4
2	22MCAP107	Data Structures and Algorithms	3	1	0	4
3	22MCAP108	Software Engineering	4	0	0	4
4	22MCAP109	Cryptography and Network Security	3	1	0	4
5	22MCAP110	Artificial Intelligence	4	0	0	4
6		Discipline Elective-I	3	0	0	3
7	22MCAP203	Full Stack Web Development Laboratory	0	0	3	1.5
8	22MCAP204	Data Structures and Algorithms Laboratory	0	0	3	1.5
9	22MCAP205	Software Engineering Laboratory	0	0	3	1.5
10		Skill Oriented Course – II	1	0	2	2
Total			**22**	**2**	**11**	**29.5**

MCA II Year I Semester

S.No	Course Code	Course Title	L	T	P	C
1	22MCAP111	Mobile Application Development	4	0	0	4
2	22MCAP112	Fundamentals of Data Science	3	1	0	4
3	22MCAP113	Java Programming	3	1	0	4
4		Discipline Elective-II	3	0	0	3
5		Discipline Elective-III	3	0	0	3
6		Open Elective	3	0	0	3
7	22MCAP206	Mobile Application Development Laboratory	0	0	3	1.5
8	22MCAP207	Data Science using Python Laboratory	0	0	3	1.5
9	22MCAP208	Java Programming Laboratory	0	0	3	1.5
10		Audit Course	2	0	0	0
11	22MCAP701	Mini Project	0	0	4	2
Total			**21**	**2**	**13**	**27.5**

MCA II Year II Semester

S.No	Course Code	Course Title	L	T	P	C
1	22MCAP702	Comprehensive Viva	0	0	4	2
2	22MCAP703	Internship/Major Project	0	0	26	13
Total			**0**	**0**	**30**	**15**

DISCIPLINE LECTIVES

Discipline Elective – I

S.No	Course Code	Course Title	L	T	P	C
1	22MCAP401	Machine Learning - Algorithms and Applications	3	0	0	3
2	22MCAP402	IoT Technology and Applications	3	0	0	3
3	22MCAP403	Agile Software Development Process	3	0	0	3
4	22MCAP404	Computer Graphics and Multimedia	3	0	0	3
5	22MCAP405	Image Processing	3	0	0	3
6	22MCAP406	Blockchain Technology	3	0	0	3
7	22MCAP407	XML and Web Services	3	0	0	3

Discipline Elective – II

S.No	Course Code	Course Title	L	T	P	C
1	22MCAP408	Deep Learning	3	0	0	3
2	22MCAP409	Wireless Sensor Networks	3	0	0	3
3	22MCAP410	Software Quality Assurance and Testing	3	0	0	3
4	22MCAP411	Virtual Reality	3	0	0	3
5	22MCAP412	Video Analytics	3	0	0	3
6	22MCAP413	Software Development using Block chain	3	0	0	3
7	22MCAP414	Advanced Web Development	3	0	0	3

Discipline Elective – III

S.No	Course Code	Course Title	L	T	P	C
1	22MCAP415	Reinforcement Learning	3	0	0	3
2	22MCAP416	Privacy and Security in IoT	3	0	0	3
3	22MCAP417	Software Project Management	3	0	0	3
4	22MCAP418	Generative AI	3	0	0	3
5	22MCAP419	Computer Vision	3	0	0	3
6	22MCAP420	Cyber Security using Blockchain	3	0	0	3
7	22MCAP421	.Net Framework and C#	3	0	0	3

Open Elective

S.No	Course Code	Course Title	L	T	P	C
1	22MBAP301	Management and Organizational Behavior	3	0	0	3
2	22MBAP302	Design Thinking	3	0	0	3
3	22MBAP303	Management Information System	3	0	0	3
4	22MBAP304	E-Commerce and Digital Markets	3	0	0	3
5	22MBAP305	Entrepreneurship Development and Project Management	3	0	0	3

Skill – Oriented Course - I

S.No.	Course Code	Course Title	L	T	P	C
1	22MCAP601	Programming with C++	1	0	2	2
Any Skill-Oriented courses can be appended in future.						

Skill – Oriented Course - II

S.No.	Course Code	Course Title	L	T	P	C
1	22MCAP602	Programming with Matlab	1	0	2	2
Any Skill-Oriented courses can be appended in future.						

Audit Course

S.No.	Course Code	Course Title
1	22ENGP903	Soft Skills
Swayam NPTEL MOOCs		
2	22ENGP9M01	Soft Skills
3	22ENGP9M02	Developing Soft Skills and Personality
4	22ENGP9M03	Soft Skill Development
5	22ENGP9M04	Enhancing Soft Skills and Personality
Any Soft Skill related courses can be appended in future.		

MCA I Year I Semester

MCA I Year I Semester

22MATP101 MATHEMATICAL FOUNDATIONS FOR COMPUTER APPLICATIONS

L	T	P	C
3	1	0	4

Course Prerequisite: Basic Mathematics

Course Description:

This course introduces the concepts of set theory, relations, functions, logic and rules of inference. It illustrates the structured approach to graph theory models. It provides the valuable basic information on mathematical statistics. It explains about key concepts of probability and random variables and probability distributions.

Course Objectives:

Students will be able to:

1. Understand the mathematical reasoning to construct mathematical and logical arguments.
2. Differentiate various types of relations, Functions and posets.
3. Illustrate the different terminologies of graph theory and related concepts.
4. Identify the fundamentals of mathematical statistics.
5. Examine key concepts of probability, random variables and Probability distributions.

UNIT-I PROPOSITIONAL LOGIC **12 hours**

Propositions, Connectives, Conditionals and Biconditionals, Well-formed formulae, Tautologies, Equivalence of formulas, Normal Forms, Rules of inference.

UNIT-II RELATONS, FUNCTIONS AND POSETS **12 hours**

Set Theory, Functions and Relations, Properties of a Relation, Binary relations, Matrix and Digraph representation of a relation, Equivalence relation, Partially ordered Relations ,POSETS , Hasse Diagram, Lattices, Properties of Lattices.

UNIT-III GRAPH THEORY **12 hours**

Basic terminology for undirected and directed graphs, multigraphs and weighted graphs, paths and circuits, Eulerian paths and circuits, Hamiltonian paths and circuits, Planar Graphs, Graph Coloring, Cut sets. Trees: Introduction to Trees, Tree terminology, Prefix codes.

UNIT-IV DESCRIPTIVE STATISTICS **12 hours**

Measures of central tendencies and Dispersion, Coefficient of variation, Skewness, Kurtosis, Data visualization, Grouped data, Histograms, Ogives, Percentiles, Box-Plot, Correlation, Scatter diagram, Rank correlation and Linear Regression.

UNIT-V PROBABILITY **12 hours**

Introduction to Probability, , Axioms of probability, Conditional Probability, , Bayes theorem, Random Variable, discrete and continuous probability densities, cumulative distribution, Expectation, variance and standard deviation, Binomial distribution, Poisson Distribution and Normal Distribution.

Course Outcomes:

After completion of the course the student will be able to

1. Form truth tables, prove results by truth tables, find normal forms and learn proof techniques and concepts of inference
2. Construct the matrix, digraphs of relation, Identify different types of relations, functions and draw the Hasse diagram for poset.
3. Apply graph theory models to solve different communication and network problems.
4. Analyze the data and characterize the nature of the distribution of data.
5. Solve the real time problems through the probability and its distributions.

Text Books:

1. Kenneth H Rosen, Discrete Mathematics and Its Applications, 7th Edition, 2017, McGraw Hill Education.
2. Aczel Sounder Pandian, Complete Business Statistics, 7th Edition, McGraw Hill Education.

Reference Books:

1. J.S. Milton and J.C. Arnold, Introduction to Probability and Statistics, 4th edition, 2003 Tata McGraw-Hill Publications.

2. C.L. Liu & Mohapatra, Elements of Discrete Mathematics, 4th Edition, 2017, McGraw Hill Education.

3. Thomas Koshy, Discrete Mathematics with Applications, 2012, Elsevier Academic Press.

4. S. C Gupta and V.K Kapoor, Fundamentals of Mathematical Statistics, 11th edition 2010 Sultan Chand & Sons, New Delhi.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA I Year I Semester

22MCAP101 PYTHON PROGRAMMING

L	T	P	C
3	1	0	4

Course Prerequisite: None

Course Description:

This course provides the students how to write programs in python language to perform different tasks in major concepts such as functions, collections, comprehensions, files, exception handling and OOPS.

Course Objectives:

Student will be able to

1. Understand the basics of python programming such as variables, data types, operators, etc.
2. Apply the concept of Lambda and user defined functions
3. Understand the use of collections such as string, list, set, tuple, dictionary
4. Implement Python comprehensions and manage file operations for efficient code and data processing.
5. Handle exceptions and apply OOPS concepts to create robust, modular, and reusable code.

UNIT-I FUNDAMENTALS 12 hours

History, Python-Interpreter, Keywords, Variables, Datatypes, Operators, Expressions, Input and Output functions, Type conversion functions, Indentations, Conditionals - if constructs, Loop Structures - For loop, While loop, break and continue statements, Local and Global variables, Modules, Packages and Libraries

UNIT-II LAMBDA AND USER DEFINED FUNCTIONS 12 hours

Lambda functions, User Defined Function definition and function call, parameter passing, Pass and Return statements, Types of functions -Default arguments, Arbitrary arguments using *, Keyword arguments, Arbitrary Keyword arguments using **, Recursive functions.

UNIT-III COLLECTIONS 12 hours

String operations and functions, Slicing and Indexing operations, format function, multiple inputs using split function, List operations and functions, Tuple operations and functions, Set operations and functions, Dictionary operations and functions, Array and its operations

UNIT-IV COMPREHENSIONS AND FILE HANDLING 12 hours

List comprehensions - for, for with if , for with if and else, nested comprehensions with for, Dictionary comprehensions, Dictionary comprehension with if, Set comprehensions, map, zip, filter and other related functions, File path, opening and closing files, reading and writing files, File position.

UNIT-V EXCEPTION HANDLING AND OOPS 12 hours

Errors and Exceptions, Exception Handling, Multiple Exceptions, Raise statement, User Defined Exceptions, OOPS-Class, Object, Constructor, Methods, Inheritance and its types, super function, method overriding, Encapsulation, Polymorphism.

Course Outcomes:

After completion of the course the student will be able to:

1. Understand the fundamentals of Python programming, including variables, data types, and operators.
2. Apply the concepts of lambda functions and user-defined functions.
3. Utilize collections such as strings, lists, sets, tuples, and dictionaries.
4. Analyze the use of comprehensions and file operations.
5. Analyze and apply the concepts of exceptions and OOP in real-world applications.

Text Books:

1. Python Programming – An Introduction to computer science, John Zelle, JimLeisy
2. Programming and Problem Solving with Python by Ashok Namdev Kamthane and Amit Ashok Kamthane, McGraw Hill Education; First edition (1 November 2017)

Reference Books:

1. Programming Python, Mark Lutz, O'Reilly, 3rd Edition, 2006
2. Core Python Programming, Wesley J Chun, PH, 2nd Edition Python Programming: A Compatible Guide for Beginners to Master and Become an Expert in python programming Language, Brain Draper, CreateSpace Independent Publishing Platform, 2016

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA I Year I Semester

22MCAP102 DATABASE MANAGEMENT SYSTEMS

L	T	P	C
3	1	0	4

Course Prerequisite: None

Course Description:

This course is designed to investigate how database management system techniques are used to design, develop, implement and maintain database applications in organizations.

Course Objectives:

Students will be able to:

1. Explain data base concepts and its applications, data base system structure and ER data modeling.
2. Illustrate the relational model, DDL, DML and Relational Algebra.
3. Demonstrate Schema refinement and Normal Forms.
4. Explain transactions and its management, concurrency control schemes and recovery.
5. Interpret the concepts of PL/SQL and No SQL.

UNIT I: INTRODUCTION 12 hours

Introduction to data bases, Database system Vs File system, Data abstraction, Instances and Schemas, Database users, Database system structure, Entities, Attributes, Entity sets, Relationships and Relationship sets, Database design and ER diagrams, Specialization and Generalization and participation features of ER model.

UNIT II: THE RELATIONAL MODEL, SQL & RELATIONAL ALGEBRA 12 hours

The Relational model: Introduction, Various types of Integrity constraints.

SQL: Structured Query Language, data types, DDL, DML, TCL, views, NULL Values, Set operations, aggregate functions, character functions, Date functions, String functions, Nested queries, correlated nested queries.

Relation Algebra: Fundamental operations, Additional operations.

UNIT III: NORMALIZATION 12 hours

Problems Caused by redundancy, FIRST, SECOND,THIRD Normal forms, BCNF, Lossy decomposition, Lossless-join Decomposition, Dependency preserving, Multi valued Dependencies , Fourth Normal Form and Fifth Normal form.

UNIT IV: RANSACTION MANAGEMENT AND CONCURRENCY CONTROL

12 hours

Transaction Management: ACID properties, Transactions and Schedules, Concurrent Execution of transactions, Serializability, Conflict serializability, View serializability, Testing for serializability.
Concurrency Control: Two Phase Locking protocol, Time stamp ordering protocol

Database recovery
UNIT V: PL/SQL & NO SQL 12 hours

PL/SQL: Functions, procedures, triggers, cursors and exceptional handling, Package in PL/SQL.

No SQL : Key value store Database, Column Store Database, Document Database, Graph Database

Course Outcomes:

After completion of the course the student will be able to

1. Demonstrate the basic concepts and advantages of DBMS and draw ER diagram for given problem.
2. Implement DDL, DML and various types of SQL queries.
3. Design Normal Forms for the given application.
4. Illustrate the concepts of transaction processing, concurrency control and recovery.
5. Execute a relational database system using PL/SQL and provide introduction of No SQL

Text Books:

1. Silberschatz, Korth, Sudarshan Data Base System Concepts, 6/e,TMH,2019
2. Raghurama Krishnan, Johannes Gehrke, Data Base Management Systems, TMH
3. Sadalage Pramod J, Fowler Martin, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot persistence , 1st Edition, Addison-Wesley, 2013

Reference Books:

1. Elmasri Navathe, Data Base Management System, Pearson publications
2. C J Date, An Introduction to Database Systems, Pearson

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA I Year I Semester

22MCAP103 COMPUTER ORGANIZATION AND ARCHITECTURE

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:
This course introduces internal digital circuits & their operations and basic building block of various computers. Addressing modes and pipelining detailed along with Memory Management. It also covers the concepts of I/O organization and Multiprocessor

Course Objectives:
Students will be able to

1. Understand the basics of digital circuits
2. Analyze the working mechanism of processing unit.
3. Classify the Pipelining techniques.
4. Explore the concept of Memory Hierarchy.
5. Understand the process of I/O organization and Multiprocessor

UNIT I DIGITAL BUILDING BLOCKS 9 hours

Basic structure of a digital computer, Components of computer, Logic gates, combinational circuits, flip flop, Sequential Circuits, adders, multiplexers, encoders, decoders, Registers, Decoder, Multiplexers, Number System Conversion, fixed Point and floating point representation.

UNIT II PROCESSING 9 hours

Fundamental concepts, Parts of a mother board along with various CPU's, Instruction Set, Instruction formats, Addressing modes, RISC, CISC, Registers, Hardwired control, Micro programmed control, Nano programming

UNIT III PIPELINING 9 hours

Basic concepts: Data hazards, Instruction hazards, Control Hazards Influence on instruction sets, Data path and control considerations, Performance considerations Exception handling, Case Study: Intel Core i7 Pipelines

UNIT IV MEMORY SYSTEM 9 hours

Basic concepts, Memory Hierarchy, Characteristics of primary and secondary memories, Semiconductor, Magnetic, Optical memories, Semiconductor Memories (RAM,ROM), Cache memories, RAID Levels, Virtual memory, Secondary storage devices.

UNIT V I/O ORGANIZATION 9 hours

Input Output Interface, Connect different Input /Out Put devices, Asynchronous Data Transfer, Modes of Transfer, Priority interrupts, Direct Memory Access.

Multiprocessor: Characteristics of Multiprocessor, Inter Connection Structure, Inter Process Arbitration, Inter Process Communication and synchronization, cache Coherence.

Course Outcomes:

After successful completion of the course, students will be able to

1. Demonstrate the basic components of Digital Computer.
2. Explain the internal working concept of processing unit along with instruction sets.
3. Identify the different Pipelining flow characteristics.
4. Illustrate various memory organization structures.
5. Analyze the I/O Organization Interface structure along with multiprocessor

Text Book(s)

1. Mano, M. Morris. Computer system Architecture. Prentice-Hall of India, 4th Edition, 2003.

2. Stallings, W. Computer Organization and Architecture: designing for performance. Pearson Education India, 2003.

Reference Books

1. Computer Organization, Carl Hamacher ,Zaky

2. Computer Architecture and Organization, Miles Murudocca, Vincent Heuring ,Wiley
3. Computer System Organization and Architecture, M.Usha, T.S.Srikanth

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA I Year I Semester

22MCAP104 OPERATING SYSTEMS

L	T	P	C
4	0	0	4

Pre-requisite: None

Course Description:

The purpose of this course is to provide an overview of computer operating systems. Topics covered include: Operating system structures, processes, process synchronization, deadlocks, CPU scheduling, memory management, file systems, Threads and disk management.

Course Objectives:

Students will be able to:

1. Explain the fundamental components of the operating system and process concepts
2. Understand the principles of Process management and Deadlocks.
3. Analyze various CPU scheduling algorithms
4. Study the importance of memory management
5. Compare I/O managements and File systems

UNIT I INTRODUCTION 12 hours

Introduction to operating systems -Types of operating systems-Operating System Structures-Operating System Operations- Operating Systems Services-System Calls- System Programs -Process Concept-Process Scheduling–Operations on Processes – Cooperating Processes – Inter-process Communication.

UNIT II PROCESS MANAGEMENT 12 hours

System Model – Deadlock Characterization – Methods for handling Deadlocks -Deadlock Prevention – Deadlock avoidance – Deadlock detection – Recovery from Deadlocks - Storage Management – Swapping – Contiguous Memory allocation – Paging – Segmentation – Segmentation with Paging

UNIT III CPU SCHEDULING 12 hours

Threads – Overview – Threading issues - CPU Scheduling – Basic Concepts – Scheduling Criteria – Scheduling Algorithms – Multiple-Processor Scheduling – Real Time Scheduling - The Critical-Section Problem – Synchronization Hardware – Semaphores – Classic problems of Synchronization – Critical regions – Monitors.

UNIT IV MEMORY MANAGEMENT 12 hours

Virtual Memory – Demand Paging – Process creation – Page Replacement – Allocation of frames – Thrashing - File Concept – Access Methods – Directory Structure – File System Mounting – File Sharing – Protection

UNIT V FILE SYSTEM 12 hours

File System Structure – File System Implementation – Directory Implementation – Allocation Methods – Free-space Management. Kernel I/O Subsystems - Disk Structure – Disk Scheduling – Disk Management – Swap-Space Management - Case Study: The Linux System & Windows

Course Outcomes:

After completion of the course the student will be able to:

1. Explain the Operating Systems basics and process concepts.
2. Analyze various process synchronization approaches
3. Elaborate various CPU scheduling algorithms
4. Classify the various memory management Concepts
5. Identify the concept of file systems.

Text Book(s)

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc., 2012.
2. Harvey M. Deitel, “Operating Systems”, 2 nd edition, Pearson Education, 2002.

Reference Books

1. William Stallings, “Operating Systems – Internals and Design Principles”, 7th Edition, Prentice Hall, 2011.
2. Andrew S. Tanenbaum, “Modern Operating Systems”, Second Edition, Addison Wesley, 2001.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA I Year I Semester

22MCAP105 COMPUTER NETWORKS

L	T	P	C
4	0	0	4

Pre-requisite None

Course Description:
This course provides an introduction to computer networks, with a special focus on the Internet architecture and protocols. Topics include layered network architectures, addressing, naming, forwarding, routing, communication reliability, the client-server model, web and email protocols.

Course Objectives:
Students will be able to

1. Understand networking concepts and basic communication model.
2. Classify the network architectures and components required for data communication.
3. Analyse the function and design strategy of physical, data link, network layer and transport layer.
4. Identify various application protocol for internet security issues and services
5. Investigate the principles of various application protocols and security issues and services available

UNIT I NETWORK FUNDAMENTALS 12 hours

Uses of Networks – Categories of Networks -Communication model –Data transmission concepts and terminology – Protocol architecture – Protocols – OSI – TCP/IP – LAN Topology - Transmission media .

UNIT II DATA LINK LAYER 12 hours

Data link control - Flow Control – Error Detection and Error Correction - MAC – Ethernet, Token ring, Wireless LAN MAC – Blue Tooth – WiFi-WiMaX.

UNIT III NETWORK LAYER 12 hours

Network layer – Switching concepts – Circuit switching – Packet switching –IP — Datagrams — IP addresses- IPV4 and IPV6 – ICMP – Routing Protocols – Distance Vector – Link State-Flooding.

UNIT IV TRANSPORT LAYER 12 hours

Transport layer –service –Connection establishment – Flow control – Transmission control protocol – Congestion control and avoidance – User datagram protocol. -Transport for Real Time Applications (RTP).

UNIT V APPLICATIONS AND SECURITY 12 hours

Applications - DNS- SMTP – WWW –SNMP- Security –threats and services - DES- RSA.

Course Outcomes:

After successful completion of the course, students will be able to

1. Understand the flow of information from one node to another node in the network
2. Identify the components required to build different types of networks
3. Understand the functionalities needed for data communication into layers
4. Choose the required functionality at each layer for given application
5. Investigate the working principles of various application protocols and security issues.

Text Book(s)

1. Andrew S.Tannenbaum David J. Wetherall, “Computer Networks” Fifth Edition , Pearson Education 2011
2. William Stallings, ―Data and Computer Communications‖, Tenth Edition, Pearson Education, 2013.

Reference Books

1. Douglas E. Comer, ―Internetworking with TCP/IP (Volume I) Principles, Protocols and Architecture, Sixth Edition, Pearson Education, 2013.
2. Forouzan, “ Data Communication and Networking”, Fifth Edition , TMH 2012.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA I Year I Semester

22ENGP201 CORPORATE COMMUNICATION LABORATORY

L	T	P	C
0	0	4	2

Course Prerequisite: None

Course Description: English is practical and it is a must for any institution to provide students with opportunities to indulge in actively applying their language skills. Thus the Communication Skills Lab facilitates students with adequate opportunities to put their communication skills in use. It also accommodates peer learning by engaging students in various interactive sessions. This lab will be accompanied by a practical lab component.

Course Objectives: This course enables the students to –

1. Focus on their interactive skills
2. Develop their communicative competency
3. Fortify their employability skills
4. Empower their confidence and overcome their shyness
5. Become effective in their overall performance in the industry

UNIT – I LISTENING AND SPEAKING SKILLS 9 hours

Listening/watching interviews, conversations, documentaries, etc.; Listening to lectures, discussions from TV/Radio/Podcast. Conversational skills (Formal and Informal); Group Discussion; Making effective Power Point presentations;

UNIT – II READING AND WRITING SKILLS 8 hours

Reading different genres of texts including newspapers Magazines: creative writing; Writing job applications and resume; Emails; Letters; Memorandum; Reports; Writing abstracts and summaries; Interpreting visual texts.

UNIT – III ACCLIMATIZING STUDENTS TO OTHER EXAMS 5 hours

Test of English as a Foreign Language (TOEFL); International English language Testing System (IELTS): Civil Service Examinations; Verbal--ability.

UNIT – IV INTERVIEW SKILLS 8 hours

Different types of interviews: Answering questions and offering information; Mock interviews; Body Language; Articulation of sounds; Word Stress, Sentence Stress and Intonation.

Course Outcomes: At the end of the course, learners will be able to:

1. Read articles from magazines and newspapers
2. Participate effectively in informal conversations
3. Introduce themselves and their friends and express opinions in English
4. Comprehend conversations and short talks delivered in English
5. Write short essays of a general kind, draft Reports and personal letters and emails in English.

Suggested Reading/Textbook:

1. Sanjay Kumar and PushpLata; *Communication Skills*; Oxford University Press, 2012.
2. Sabina Pillai and Agna Fernandez; *Soft Skills and Employability Skills*;Cambridge University Press, 2018.
3. S.P. Dhanavel; *English and Communication Skills for Students of Science and Engineering*; Orient Blackswan, 2009.
4. M. Ashraf Rizvi; *Effective Technical Communication*; Tata Mc Graw Hill Co. ltd, 2005.

Reference:

1. Dr.M.Adithan; *Study Skills for Professional Students in Higher Education*; S.Chand& Co. Pvt., 2014.
2. Guy Brook Hart & Vanessa Jakeman; *Complete IELTS*: Cambridge University Press, 2014.
3. Vanessa Jakeman& Clare Mcdowell; *Action Plan for IELTS:* Cambridge University Press, 2006.
4. Guy Brook Hart; *Instant IELTS*;Cambridge University Press, 2004.
5. S.P.Bakshi&Richa Sharma; *Descriptive General English;*Arihant Publications, 2012.
6. Charles Browne, Brent Culligan 7 Joseph Phillips; *In Focus (level 2);* Cambridge University Press.
7. Steven Gershon; *Present Yourself 2* (second edition); Cambridge University Press.
8. Leo Jones; *Let's Talk 3*(second edition); Cambridge University Press.
9. Nutall J. C.; *Reading Comprehension*; Orient Blackswan.
10. *www.cambridgeenglish.org/in/*
11. https://learnenglish.britishcouncil.org/en/english-grammar
12. https://www.rong-chang.com/

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA I Year I Semester

22MCAP201 PYTHON PROGRAMMING LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: Any Programming Language

Course Description:

This course provides the students how to write programs in python language to perform different tasks in major concepts such as functions, collections, comprehensions, files, exception handling and OOPS.

Course Objectives:

Student will be able to

1. Write algorithms and draw flowcharts using raptor tool for different computational problems
2. Design Python programs using looping, decision making and user defined functions.
3. Develop Python programs using collections, comprehensions, files, exceptions and OOPS

LIST OF EXPERIMENTS

SIMPLE PROGRAMS USING IF CONSTRUCTS AND CONTROL STRUCTURES

1.Draw flow charts for the following problems using Raptor Tool

a. To exchange/swap the values of two variables

b. To find factorial of a given number

c. To find the sum of set of numbers

d. To generate Fibonacci sequence

e. To reverse the digits of a number

2.Develop Python scripts for the problems given in Ex.1

3.Develop Python scripts for the following

a. To count and display the total number of prime numbers in given two ranges

b. To calculate the sum of digits of given N digit number

c. To find the largest of given three numbers

4.Develop Python scripts for the following

a. To generate Armstrong numbers

b. To check the given number for perfect or not

c. To find the factors of the given number

5.Develop Python scripts for the following

a. To convert the given Centigrade/Celsius temperature into Fahrenheit

b. To generate multiplication table for the given number with terms

c. To convert the given decimal number into binary

USING LAMBDA AND UDF

6.Develop Python scripts for the following using Lambda function

a. To create a lambda function that multiplies argument x with argument y

b. To filter, odd and even numbers

c. To square and cube every number in a given list

d. To extract year, month, date and time

e. To find palindromes in a given list of strings

7.Develop Python scripts for the following using UDF

a. To check the given number for Prime or not

b. To check the given number for odd or even

c. To calculate the length of given string

d. To count vowels in given string

e. To calculate the sum of all items in the given list

USING COLLECTIONS

8.Develop Python scripts for the following without using String built-in functions

a. To calculate the length of the given string

b. To reverse the given string

c. To count and display the Vowels in the given string

d. To remove spaces from the given string

e. To convert the given string from lowercase to uppercase

9.Develop Python scripts to implement slicing and indexing operations on the following

a. String

b. List

c. Tuple

d. Set

e. Dictionary

10.Develop Python scripts for the following without using List built-in functions

a. To sum all the items in given list

b. To count the total number of items in given list

c. To check the given list is empty or not

d. To get the largest number from a given list

e. To shuffle and print the given list

11.Develop Python scripts to implement the following Set operations

a. Union function

b. Difference function

c. Intersection function

d. All and any functions

e. Enumerate function

12.Develop Python scripts to do the following in Tuple

a. To demonstrate packing and unpacking

b. To check for membership with in and not in operators

c. Slicing operations

d. To concatenate more than one tuple into a single tuple

e. Indexing operations

13.Develop Python scripts for the following in Dictionary

a. To create a dictionary and add items

b. To modify existing values using keys

c. Update function

d. To print values only

e. To print keys only

14.Develop Python scripts for the following in Array

a. To sort the given array elements

b. To add given two matrix

c. To transpose the given matrix

USING COMPREHENSIONS AND FILES

15.Develop Python scripts for the following in Comprehensions

a. To print a list with if

b. To print a list with if-else

c. To print a list with nested comprehensions

d. To print Dictionary items with a condition using zip function

e. To print Set items

16.Develop Python scripts for the following in Files

a. To create a file with given data

b. To print an existing file data

c..To print student marks with percentage

USING EXCEPTION HANDLING AND OOPS

17.Develop Python scripts for the following in Exceptions

a. To check zero division error

b. To check invalid index reference in str/list/…

c. To handle multiple exceptions

d. To raise an exception

e. To implement user defined exception

18.Develop Python scripts for the following in OOPS

a. To implement a class with attributes and methods

b. To implement a class with constructor

c. To implement single inheritance

d. To implement method overriding

e. To implement encapsulation with all access controls

Course Outcomes:

After completion of the course the student will be able to:

1. Draw flow charts for different tasks using raptor tool
2. Write python basic programs using conditional, looping structures and functions
3. Implement python programs for collections, comprehensions, files, exceptions and OOPS

Text Books:

1) Python Programming – An Introduction to computer science, John Zelle, JimLeisy

2) Programming and Problem Solving with Python by Ashok Namdev Kamthane and Amit Ashok Kamthane, McGraw Hill Education; First edition (1 November 2017)

Reference Books:

1) Programming Python, Mark Lutz, O'Reilly, 3rd Edition, 2006

2) Core Python Programming, Wesley J Chun, PH, 2nd Edition

3) Python Programming: A Compatible Guide for Beginners to Master and Become an Expert in python programming Language, Brain Draper, CreateSpace Independent Publishing Platform, 2016

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA I Year I Semester

22MCAP202 DATABASE MANAGEMENT SYSTEMS LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: None

Course Description:

This course provides an executive understanding and approach to the technical subject of database management system.

Course Objectives:
Students will be able to:

1. Explain DDL, DML,TCL and DCL Commands
2. Explain date functions Character functions, String functions and Aggregate functions
3. Explain PL/SQL functions, Procedure functions, Triggers and exceptions.

List of Experiments:

1. Implement DDL Commands. (use constraints while creating tables).
2. Implement DML commands by using various examples.
3. Implement Unique, NULL, NOT NULL, Primary key, Foreign key, Check constraints.
4. Implement DCL commands, NVL(),Coalesce() functions.
5. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSET.
6. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING.
7. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr)
8. Queries using date functions (Sysdate, next_day, add_months, last_day, months_between).
9. Implement all types of joins.
10. Calculate Employee Net salary and Gross salary for the following data using cursors. If

    Basic salary <= 8000 then DA is 8.18% of basic,

    HRA is 20% of basic PF is 20.18% of basic PT = 60Rs
    If Basic salary > 8000 and <= 14000 then DA is 32% of basic,
    HRA is 7% of basic PF = 600Rs
    PT = 80Rs
    If Basic salary >14000 and <= 16000 then
    DA is 30% of basic
    HRA is 6% of basic
    PF = 600
    PT =160

11. Implement functions, procedures and triggers in PL/SQL
12. Implement User defined and System defined exceptions.

Mini Project:

Each student has to implement any one of the following system.

I. Employee management system
II. Student Result information system
III. Bank Management system
IV. Library information system
V. Hotel management system
VI. Railway reservation system

Note: Each system contains minimum of 6 tables. Each table contains minimum 10 rows.

Course Outcomes:

At the end of the course the student will be able to:

1. Execute DDL, DML and DCL commands.
2. Implement PL/SQL procedures, functions, cursors, triggersand exceptions
3. Design Relational model for the given system.

Text Books:

1. Benjamin Rosenzweig, Elena Silvestrova, ORACLE PL/SQL by example, Pearson Education
2. Dr.P.S. Deshpande, SQL & PL/SQL for Oracle 10g, Black Book.

Reference Books:

1. Rick F.Vander Lans, Introduction to SQL, Pearson Education.
2. Steven Feuerstein, Oracle PL/SQL Programming, SPD.
3. N.Gehani, The Database Book, Universities Press.
4. Shah, Database Systems using Oracle: A Simplified Guide to SQL and PL/SQL, PHI.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

SKILL-ORIENTED COURSE

MCA I Year I Semester

22MCAP601 PROGRAMMING WITH C++

L	T	P	C
1	0	2	2

Pre-requisite: Basic Programming Knowledge

Course Description:

This course contains the basics of object oriented programming fundamentals through C++. It contains expressions, control flow, functions, arrays, pointers, managing I/O and object oriented programming features.

Course Objectives:

Students will be able to:

1. Understand the basic concepts of object oriented programming.
2. Utilize the Functions in C++ programming
3. Explain Class, Object, Constructor &Destructor and pointers.
4. Demonstrate inheritance, operator overloading and dynamic memory allocation concepts and Describe polymorphism and generic constructs
5. Explain streams, File I/O and exception handling in C++.

UNIT I **6 hours**

Introduction to OOP: Object oriented programming, Basicconcepts of OOP, Benefits, and applications of OOPS, User defined types

C++ Basics: Structure of a C++ program, Token, Data types, Declaration of variables, Expressions,Operators

1) Develop a C++ program to find all roots of a quadratic equation ax2+bx+c=0.
2) Develop a C++ program find the largest among three different numbers entered by user.
3) Develop a C++ program to find the Fibonacci series till the limit entered by the user.
4) Write a C++ program to print the following by reading number of rows to be printed from user

```
*
* *
* * *
* * * *
* * * * *
```

5) Develop a C++ program to find the sum of digits of a number

UNIT II **6 hours**

C++ Functions: Simple functions, ‘Call and Return’ by reference, Inline functions, Function overloading, Function Over riding, Default arguments, pointers and functions

1) Write a C++ program to find the value of a number raised to its power that demonstrates a function using call by value.
2) Write a C++ program to implement recursion in finding the factorial.
3) Write a C++ program write a program to find the multiplication values and the cubic values using inline function.
4) Write a C++ program to calculate the volume of cube, cylinder and rectangular box using function overloading.

UNIT III **6 hours**

Classes and Objects: Class, Member functions, Access Specifiers, Static data members and member functions, Arrays of objects, Returning objects,Friend class and friend functions.

Constructors and Destructors: Constructors, Default Constructors, Copy Constructor, Constructor overloading, Destructors, Use of “this” pointer, dynamic memory allocation

Pointers in C++: Introduction-Pointers and arrays, Pointers to objects, this pointer, Pointers to functions, Call by pointer.

1) Write a C++ program to display student’s information using Class Declarations, Definition, and Accessing Class Members by objects.
2) Define a class to represent a Bank account. Include the following members.

   Data members: Name of the depositor; Account number, Type of account, Balance amount in the account, Rate of interest (static data).
   Provide a default constructor; a parameterized constructor and a copy constructor to this class Also provide **Member Functions:** - 1. To deposit amount. 2. To withdraw amount after checking for minimum balance. 3. To display all the details of an account holder. 4. Display rate of interest (a static function) Illustrate all the constructors as well as all the methods by defining objects and destroy the object using destructor.
3) Develop a C++ program to find the area of a rectangle by converting the member of a class square which is a friend class of rectangle. Declare Rectangle as a friend of Square so that Rectangle member functions could have access to the private member of square.

4) Write a C++ program to implement flight class with data member as flight number, source, destination and fare. Write a member function to display the flight information using this pointer.

a) Write C++ Program to store GPA (Grade Point Average) of n number of students anddisplay it using new and delete operator

UNIT IV 6 hours

Inheritance: Single inheritance, Multi-level inheritance, Multiple inheritance, Hierarchical inheritance, Hybrid inheritance. Operator Overloading**.**

Virtual Functions and Run Time Polymorphism: Overriding, Static and Dynamic binding, Virtual functions, Pure virtual functions, Abstract classes.

1) Write a C++ program to accept details of n instructors and display them. Create base Class: Person (id, name) and Derive Two Sub Classes: Teaching (Subject, Name), Nonteaching(dept.) from class Person and derive new class Instructor from two sub classes Teaching and Nonteaching.
2) Write a C++ program to explain virtual function (polymorphism) by creating a base class c_polygon which has virtual function area (). Two classes c_rectangle and c_traingle derived from c_polygon and they have area () to calculate and return the area of rectangle and triangle respectively
3) Write a C++ program to count the number of persons inside a bank, by increasing count whenever a person enters a bank, using an increment (++) operator overloading function, and decrease the count whenever a person leaves the bank using a decrement (--) operator overloading function inside a class.

UNIT V 6 hours

Templates and Streams in C++: Generic Functions - Generic Classes - Console stream classes, Formatted and unformatted console I/Ooperations, Manipulators; File stream classes, File I/O.

Exception Handling: Principles of Exception handling, Exception handling mechanism, Multiplecatch, Nested try.

1) Write a C++ program create a template T for a class named pair having two data members of type T which are inputted by a constructor and a member function get-max () return the greatest of two numbers to main.

2) Write a C++ program to perform the deletion of white spaces such as horizontal tab, vertical tab, space, line feed, new line and carriage return from a text file and store the contents of the file without the white spaces on another file.
3) Write a C++ program to read the class object of student info such as name, age, sex, height and weight from the keyboard and to store them on a specified file using read () and write () functions. Again the same file is opened for reading and displaying the contents of the file on the screen

Course Outcomes:

After completion of the course the student will be able to:

1. Explain the concepts of object-oriented programming and C++ basics.
2. Interpret the concepts of functions in C++
3. Implement class and object Constructor &Destructor and pointers .
4. Demonstrate inheritance, operator overloading and dynamic Apply inheritance and operator overloading concepts and Develop the programs using polymorphism, function and class templates.
5. Implement streams, File I/O and exception handling in C++.

Text Book(s)

1. E. Balagurusamy, "Object Oriented Programming with C++", Tata McGraw-Hill, 7th Edition, 2018.
2. Schildt Herbert, The Complete Reference C++, Tata McGraw Hill, 4th Edition, 2009.

Reference Books

1. Deitel and Deitel, C++ How to program, Pearson Education Asia, 9th Edition, 2014
2. Yeshavant Kanetkar, Let Us C++, Second Edition, BPB Publications,2003.
3. Lafore, Robert, Object Oriented Programming in Turbo C++, Galgotia Publications Pvt. Ltd, 2000.
4. Gaddis Tony, Starting Out with C++, dreamtech Press, 3rd Edition, 2002.
5. Sotter A Nicholas and Kleper J Scott, Professional C++, Wiley Publishing Inc.
6. Walter Savitch, Problem solving with C++, Pearson Education, 6th Edition 2007.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

MCA I Year II Semester

22MCAP106 FULL STACK WEB DEVELOPMENT

L	T	P	C
4	0	0	4

Pre-requisite Computer Networks Basics

Course Description:

This course introduces web designing and development methodologies using the front end development languages such as HTML, CSS, Javascript and Angular JS. Also, this course gives the idea of web development process using NODE JS.

Course Objectives:

Student will be able to:

1. Understand web design concepts using HTML and CSS
2. Utilize the concepts on Bootstrap framework and JavaScript
3. Identify the working methodologies on Angular JS concepts
4. Apply Node JS techniques in web design
5. Establish the database connectivity for web application development

UNIT I HTML & CSS 12 hours

HTML Basics HTML Elements --HTML Attributes - HTML Formatting - HTML Links and Images HTML Lists, HTML tables – Frames and Framesets - Blocks, Classes - HTML Layout and Forms. CSS: Introduction CSS3 -□CSS3Colours - Backgrounds, Borders, Padding, Height/Width - CSS3 Gradients, Shadows - CSS3 Text, Fonts - CSS3 2D &3D Transforms - CSS Links – CSS Lists & Tables - CSS Box Model, Outline, Display, Max-width, Position - CSS Float, Inline-block – CSS Align – CSS Pseudo-class, Pseudo-element - CSS Navigation, Dropdowns, Tooltips, Images – CSS Selectors - CSS Forms, Buttons - CSS3 Multiple Columns - CSS3 User Interface: Box Sizing, Filters – Menu creation -Responsive CSS

UNIT II BOOTSTRAP AND JAVASCRIPT 12 hours

Introduction to Bootstrap - Bootstrap Basics - Bootstrap Grids - Bootstrap Themes – Bootstrap with CSS Introduction to JavaScript - Java Script Language Basics -Scope - JavaScript Events - Strings JavaScript Math - Arrays – Boolean – Comparisons - JavaScript Loops & Decisions - JavaScript objects and Method - JavaScript Errors - Debugging - JavaScript Functions - JavaScript Forms - JavaScript DOM

UNIT III ANGULAR JS 12 hours

Introduction – Features of Angular JS – Angular Expressions – Directives – Controllers – Modules – Forms – Dependency Injection and Services – Angular JS Animations

UNIT IV NODE JS 12 hours

NodeJS - Server-Side JavaScript – NPM - JavaScript Build Processes - Event Loop and Emitters- File System Interaction – Modules - Intro to Security and Authentication - Basic and Digest Authentication – Session - Based Authentication Document Databases

UNIT V DATABASE CONNECTIVITY WITH NODE JS 12 hours

Overview of Database Queries – Connecting String – Configuring Node JS – Working with select Command – Updating Records – Deleting Records –MERN STACK - Project Development Using Node JS

Course Outcomes:

After completion of the course, students will be able to

1. Understand the HTML and CSS scripts to design web pages.
2. Implement JavaScript methodologies while designing web pages
3. Design web pages using Angular JS techniques
4. Implement Node JS concepts in web development
5. Develop web sites using Node JS with database connectivity.

Text Book(s)

1. HTML & CSS: The Complete Reference, Fifth Edition, Thomas Powell, McGrawHill
2. Complete Bootstrap: Responsive Web Development with Bootstrap 4, Matt Lambert,Bass Johnson, David Cochran, Ian Whitley, PACKT publishing
3. Javascript for Dummies, Emily Vander Veer, Wiley Publishing, 4th Edition
4. Learning Angular JS, Ken Williamson, O.Reilly, 2015
5. Beginning Node JS, Basarath Ali Syed, Apress, 2014

Reference Books

1. Complete Bootstrap: Responsive Web Development with Bootstrap 4, Matt Lambert,Bass Johnson, David Cochran, Ian Whitley, PACKT publishing
2. Mastering Javascript, VedAntani, PACKT publishing, 2016
3. Node JS Web Development, David Herron, PACKT publishing, 2016

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP107 DATA STRUCTURES AND ALGORITHMS

L	T	P	C
3	1	0	4

Pre-requisite None

Course Description:

The purpose of this course is to impart knowledge on various data structures and design and analyze the algorithm concepts.

Course Objectives:

1. Understand the algorithms and basic concepts of data structures.
2. Implement the various types of linked list and its applications.
3. Explore and implement non- linear data structures with trees and graphs.
4. Explain different types of hashing and sorting techniques.
5. Analyze different types of Problem-Solving Methods.

UNIT I INTRODUCTION 12 hours

Introduction of Algorithms - Analyzing Algorithms - Arrays: Sparse Matrices Representation of Arrays - Stacks and Queues: Fundamentals - Evaluation of Expression - Infix to Postfix Conversion - Multiple Stacks and Queues - Perform Analyze the Algorithms.

UNIT II LINEAR DATA STRUCTURES 12 hours

Linked List: Singly Linked List – Doubly Linked List – Circular Linked List - Linked Stacks and Queues - Polynomial Addition - Sparse Matrices - Dynamic Storage Management - Garbage Collection and Compaction.

UNIT III NON - LINEAR DATA STRUCTURES 12 hours

Trees: Basic Terminology - Binary Trees - Binary Tree Representations - Threaded Binary Trees - Binary Tree - Representation of Trees.

Graphs: Terminology and Representations – Traversals - Connected Components and Spanning Trees.

UNIT IV HASHING AND SORTING TECHNIQUES 12 hours

Introduction to Hashing, Deleting from Hash Table, Efficiency of Rehash Methods, Hash Table Reordering, Resolving collision by Open Addressing.

Sorting Techniques: Bubble sort – Insertion sort - Quick sort – merge sort – heap sort - Divide and Conquer method.

UNIT V PROBLEM-SOLVING METHODS 12 hours

Dynamic Programming Methods: All pairs shortest paths, Travelling salesman problem; Backtracking- N-queen problem, Graph coloring; Branch and Bound-15-puzzle.

Course Outcomes:

After successful completion of the course, students will be able to

1. Understand the algorithms and fundamental concepts of data structures.
2. Analyze and implement the linear data structures using linked lists.
3. Applying Non-linear data structures concepts and algorithms.
4. Compare various hashing and sorting techniques.
5. Implement various types of problem-solving approaches.

Text Book(s)

1. Fundamentals of Data structures, E. Horowitz, S. Sahni and Susan Anderson-Freed, Universities Press.
2. EllisHorowitz, SartajShani "Data and File Structures', GalgotiaPublication, Second Edition,2013.

Reference Books

1. Benjamin Baka , Python Data Structures and Algorithms, Packt Publishing 2016.

2. Rance D. Necaise "Data Structures and Algorithms Using Python", John Wiley & Sons, Inc.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP108 SOFTWARE ENGINEERING

L	T	P	C
4	0	0	4

Pre-requisite None

Course Description:

Software engineering is the branch of computer science that creates practical, cost-effective solutions to computing and information processing problems, preferentially by applying scientific knowledge, developing software systems in the service of mankind. This course covers the fundamentals of software engineering, including understanding system requirements, finding appropriate engineering compromises, effective methods of design, coding, and testing, team software development.

Course Objectives:

The course is designed to meet the objectives of:

1. Understand software engineering principles and process models.
2. Analyze software requirements in the engineering process.
3. Explain UML diagrams for software design.
4. Demonstrate software testing strategies.
5. Identify tools for software reliability and quality management.

UNIT I INTRODUCTION TO SOFTWARE ENGINEERING AND PROCESS MODELS 12 hours

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths. A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI), process patterns, process assessment, personal and team process models. Process models: The waterfall model, incremental process models, evolutionary process models, the unified process.

UNIT II SOFTWARE REQUIREMENTS & MODELS 12 hours

Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document. Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management. System models: Context models, behavioural models, data models, object models, structured methods.

UNIT III DESIGN ENGINEERING 12 hours

Design Engineering: Design process and design quality, design concepts, the design model. Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modelling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

UNIT IV TESTING STRATEGIES 12 hours

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging. Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

UNIT V SOFTWARE RELIABILITY AND QUALITY MANAGEMENT 12 hours

Software Reliability And Quality Management: Software Reliability, Statistical Testing, Software Quality, Software Quality Management System, ISO 9000, SEI Capability Maturity Model. Software Maintenance: Characteristics of Software Maintenance. Software Reuse: what can be Reused? Why almost No Reuse So Far?, Basic Issues in Reuse Approach, Reuse at Organisation Level.

Course Outcomes:

After successful completion of the course, students will be able to

1. Comprehend the role of software and various process models.
2. Analyze the software project from requirement gathering to implementation.
3. Apply various UML diagrams to the software project.
4. Assess the testing strategies and product metrics.
5. Understand knowledge about software reliability and quality management.

Text Book(s)

1. Software Engineering: A Practitioner's Approach, Roger S. Pressman, 10th ed, Mc Graw Hill
2. Rajib Mall, “Fundamentals of Software Engineering”, 5th Edition, PHI, 2018.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

Reference Books

1. Software Engineering, 8/e, Sommerville, Pearson
2. Software Engineering principles and practice, W S Jawadekar, TMH
3. Craig Larman, Applying UML and Patterns: An Introduction to object- Oriented Analysis and Design and iterative development,3rd Edition, Pearson,2005.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP109 CRYPTOGRAPHY AND NETWORK SECURITY

L	T	P	C
3	1	0	4

Pre-requisite: Computer Networks

Course Description:

Network Security introduces techniques for protecting information and network components against attacks. It highlights the core cryptographic mechanisms and their implementation procedure to provide security for the data. Investigates various networking security standards and methods for enforcing and enhancing those standards. It also covers the electronic mailing system and supporting protocols.

Course Objectives:

Students will be able to

1. Understand the symmetric key encryption techniques
2. Explain asymmetric key cryptographic algorithms and encryption methods for security.
3. Demonstrate the message authentication codes and hashing algorithms.
4. Apply the method of public, private key distribution and applications related to authentication.
5. Develop about IP security and Transport layer security with supporting protocols

UNIT I SYMMETRIC CIPHERS 12 hours

Introduction to security attacks, services and mechanisms, Classical Encryption Techniques – Substitution Ciphers and Transposition Ciphers, Stream and Block Ciphers. Modern Block Ciphers: Block cipher principles, Shannon's theory of confusion and diffusion, fiestal structure, modular arithmetic, DES, AES.

UNIT II ASYMMETRIC CIPHERS 12 hours

Prime and Relative Prime numbers, Extended Euclidean Algorithm, Fermat's and Euler's theorem, primality testing, Chinese Remainder Theorem, Discrete Logarithmic Problem, Principle of Public key crypto systems, RSA algorithm, security of RSA.

UNIT III MESSAGE AUTHENTICATION 12 hours

Message Authentication Codes: Authentication requirements, authentication functions, message authentication code, hash functions, security of hash functions, Secure hash algorithm (SHA). Digital Signatures: Digital signature standards (DSS).

UNIT IV KEY DISTRIBUTION & AUTHENTICATION APPLICATIONS 12 hours

Key Management and distribution: Symmetric key distribution, Diffie-Hellman Key Exchange, Public key distribution, X.509 Certificates, Public key Infrastructure (PKI). Authentication Applications: Kerberos, Electronic mail security: Secure Multipurpose Internet Mail Extensions (S/MIME).

UNIT V NETWORK & TRANSPORT LAYER SECURITY 12 hours

IP Security: Architecture, Authentication header, Encapsulating security payloads. Introduction to Secure Socket Layer, Secure electronic transaction (SET). System Security: Introductory idea of Intrusion, Intrusion detection systems, Viruses and related threats.

Course Outcomes:

After Completion of the course, the student will be able to

1 Understand the knowledge about block ciphering and symmetric ciphering techniques.
2. Implement RSA algorithm
3. Analyse the Secure Hashing and Digital Signature Algorithms.
4. Implement authentication applications Kerberos and MIME.
5. Apply Knowledge on IP security, Transport layer security Protocols and system security

Text Book(s)

1. William Stallings, “Cryptography and Network security Principles and Practices”, Pearson/PHI, 4th Edition
2. Behrouz A. Frouzan, “Cryptography and Network Security”, Tata McGraw Hill, 2nd Edition

Reference Books

1. Bruce Schiener, “Applied Cryptography”. John Wiley & Sons
2. Bernard Menezes,” Network Security and Cryptography”, Cengage Learning
3. Atul Kahate, “Cryptography and Network Security”, TMH

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP110 ARTIFICIAL INTELLIGENCE

L	T	P	C
4	0	0	4

Course Prerequisites: None

Course Description:

To understand the importance of AI techniques and its wide range of applications. The course is introduced with basics of AI and ML. Then and applications and techniques of AI in various domains

Course Objectives:

Students will be able to:

1. Understand the fundamentals of AI and ML
2. Summarize NLP, Chatbots and Speech Recognition
3. Demonstrate the concepts of Image Processing and Computer Vision
4. Discuss reinforcement learning and its applications
5. Develop various real time applications that are implemented using AI

UNIT I: INTRODUCTION TO AI & ML 12 hours

AI: What is AI, Applications of AI, Advanced search, Constraint satisfaction problems, Knowledge representation and reasoning, Uncertain and probabilistic reasoning - ML: Introduction, Types of Machine Learning, Perceptron, Neural Networks, Applications of Machine Learning

UNIT II: NLP, CHATBOTS & SPEECH RECOGNITION 12 hours

Natural Language Processing: Introduction, Natural Language Understanding, Components of NLP, Enterprise Applications, how to Use NLP, Challenges of NLP - Chatbots: Introduction, how to Build a Chatbot, Architecture of Chatbot, Challenges of building a Chatbot, Industry case studies - Speech Recognition: Speech Fundamentals, Speech Analysis, Speech Modeling, Speech Recognition, Speech Synthesis, Text to Speech,Virtual Assistants: What is a Virtual Assistant?

UNIT III: IMAGE PROCESSING & COMPUTER VISION 12 hours

Image Processing: Introduction, Image Noise Removal of Noise, Removal of Noise from Images, Colour Enhancement, Segmentation, Edge Detection, Optical Character Recognition, Feature Detection & Recognition, Feature Extraction - Computer Vision: Capabilities of Computer Vison for an Enterprise, how to use Computer Vision, Computer Vision on Mobile Devices, Existing Challenges and Implementation - Agriculture [Crop and Soil Monitoring, Predictive Analytics], Retail and Retail Security [Amazon Go]

UNIT IV: REINFORCEMENT LEARNING 12 hours

Reinforcement Learning: Introduction, Game Playing [Deep Blue in Chess, IBM Watson in Jeopardy, Google's DeepMind in AlphaGo], Agents and Environment, Action-Value Function, Applications: Robotics, Gaming, Diagnostic systems, Virtual Assistants

UNIT V: SMART APPLICATIONS 12 hours

Smart Manufacturing, Smart Agriculture, Smart Grids, Smart Transportation and Autonomous Vehicles, Smart Homes, Smart Cities.

Course Outcomes:

After completion of the course the student will be able to:

1. Understand the basic concepts of AI, ML and its applications.
2. Discuss briefly about NLP, Chatbots and Speech Recognition.
3. Illustrate the concepts of Image Processing and Computer Vision.
4. Demonstrate reinforcement learning and its applications.
5. Implement smart applications for various domains.

Textbooks:

1. Tom Markiewicz& Josh Zheng,Getting started with Artificial Intelligence, Published by O'Reilly Media,2017
2. Stuart J. Russell and Peter Norvig,Artificial Intelligence A Modern Approach
3. Joseph Howse, Prateek Joshi, Michael Beyeler - Opencv_ Computer Vision Projects with Python-Packt Publishing (2016)

Reference Books:

1. A classical approach to Artificial Intelligence, Munesh Chandra Trivedi, Khanna Publications
2. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer 2010
3. Artificial Intelligence and Machine Learning, Chandra S.S. & H.S. Anand, PHI Publications
4. Machine Learning, Rajiv Chopra, Khanna Publishing House
5. Build an AI Assistant with Wolfram Alpha and Wikipedia in Python. https://medium.com/@salisuwy/build-an-ai-assistant-with-wolfram-alpha-and-wikipedia-in-python-d9bc8ac838fe

Mode of evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I YEAR II Semester

22MCAP203 FULL STACK WEB DEVELOPMENT LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite Basic knowledge on Java, DBMS

Course Description:

Full Stack Web Development course will make students to become master in front-end technology. It provides basic information and experiments to grow to be a Full-Stack web developer. With fast growing technologies, the students can update their knowledge on technologies. This will help the students to learn the complete set of process like designing, development and deployment.

Course Objectives:

Students will be able to:

1. Acquire knowledge on web designing using front end tools
2. Develop coding using scripting languages
3. Develop applications using AJAX

List of Programs

1. Write a HTML Program to create a Website for your department using all HTML tags.
2. Write a Java Script program to perform Arithmetic Operations
3. Write a Java Script Program
   i. To Sort an Array of Strings
   ii. To print the array values using for loop
4. Develop a Javascript program to print the Even numbers less than the given number using do-while loop
5. Write a Javascript program to
   i. Check whether the person is eligible for vote or not using if statement
   ii. Print the Message “Good Morning”, “Good Afternoon” and “Good Evening” based on system timings. (Use If Else)
6. Write Javascript programs to perform all relational operations and show “True” or “False”
7. Write a Javascript program
   i. To count the number of vowels using functions
   ii. To perform the arithmetic operations using functions
8. Develop a web page using HTML, CSS and Javascript to display a Calculator. If the button is clicked for Normal Calculator it should display normal calculator. If the button is clicked for Scientific Calculator it should display the scientific calculator
9. Develop a Javascript program to create a validation form
10. Write a Javascript program to demonstrate the Javascript DOM Events
11. Create a form that collects the first name, last name, email, user id, password and confirm password from the user. All the inputs are mandatory and email address entered should be in

correct format. Also, the values entered in the password and confirm password textboxes should be the same. After validating using JavaScript, display proper error messages in red color just next to the textbox where there is an error.

12. Write a Java Script Program to compute the average and grade of the student
13. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and email id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document

Course Outcomes:

After completion of the course, students will be able to

1. Implement webpages using HTML and CSS
2. Execute different types of Javascript programs
3. Apply the concepts of XML for web programming

Text Book(s)

1. Java Script for ProgrammersPaul J. Deitel, Deitel & Associates, Inc.Harvey M. Deitel, Deitel& Associates, Inc.
2. XSLT: Working with XML and HTMLKhun Yee Fung, Addison Wesley, 2000

Reference Books

1. Web Coding Bible, An Accelerated Course, Chong Lip Phang, 2015

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA I Year II Semester

22MCAP204 DATA STRUCTURES AND ALGORITHMS LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite None

Course Description:
This course introduces on the practical part of Data Structures and Algorithms using Python language. This course allows students to implement linear and nonlinear data structures. It gives practical exposure for solving sorting and searching problems. It enables to develop programs using various problem solving methods.

Course Objectives:

Students will be able to

1. Design Linear and Non-Linear Data structures.
2. Compare different types of searching and sorting techniques.
3. Demonstrate various problem-solving methods.

List of Programs:

1. Create a Stack and do the following operations.
   a. Push b. Pop c. Peep
2. Create a queue and do the following operations.
   a. Add b. Delete c. Display
3. Write a program to convert from infix to postfix expression.
4. Implement the operations on the following data structures:
   a. singly linked list b. circular linked list c. doubly linked list
5. Implement the operations on Binary Search tree.
6. Write a program on Tree Traversal
   a. Pre order b. In order c. Post order
7. Write a program to find minimum spanning tree using
   a. Prim's method b. Kruskal's method.
8. Write a program to sort elements using the following sorting techniques
   a. Insertion b. Selection c. Bubble.
9. Implement the following Divide and Conquer techniques to sort elements.
   a. Quick sort b. Merge sort
10. Implement single source shortest path problem.
11. Implement all pairs shortest path problem.
12. Implement N-queen's problem using backtracking.

Course Outcomes:

After completion of the course the student will be able to

1. Select a suitable data structure for real time applications.
2. Implement Linear and Non-Linear Data structures.
3. Apply various problem-solving techniques for complex problems.

Text Book(s)

1. Data Structure and Algorithms in Python by Michael T. Goodrich, and Roberto Tamassia O’reilly and wiley.
2. Data Structures and Algorithm using Python by Brad Miller and David Ranum, Luther College.
3. Problem-Solving with Algorithms and Data Structures Using Python by Bradley N. Mille, Franklin Beedle & Associate Incorporated.

Reference Books

1. Data Structures, S. Lipscutz, Schaum’s Outlines, TMH.

2. Data Structures and Algorithms ,Alfred Aho,John E. Hop Croft, Ullman, Addision Wesley.
3. Algorithm Design by Jon Kleinberg, Pearson.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA I Year II Semester

22MCAP205 SOFTWARE ENGINEERING LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite None

Course Description:

This course focuses on providing hands-on experience in designing and developing software systems. Specifically, the course studies software modelling tools, and testing tools.

Course Objectives:

1. To have hands on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.
2. To practice the various design diagrams through the appropriate tool.
3. To learn to implement various software testing strategies

List of Programs:

1. Introduction to UML, To Develop Data Dictionary and Use case Diagram
2. To Develop Activity diagram and Class diagram
3. To Develop Sequence diagrams and Collaboration Diagram
4. To add interface to class diagram
5. To Develop Deployment diagram
6. To Prepare test plan and perform validation testing.
7. Develop test cases for unit testing and integration testing
8. Develop test cases for various white box and black box testing techniques.

Course Outcomes:

After successful completion of the course, students will be able to

1. Design and Implement UML diagrams for real word applications.
2. Implement various testing strategies for software projects.
3. Develop test cases for various testing techniques.

Text Book(s)

1. Software Engineering? A Practitioner‟ s Approach, Roger S. Pressman, 1996, MGH.

Reference Books

1. Software Engineering by Ian Sommerville, Pearson Edu, 5th edition, 1999

2. An Integrated Approach to software engineering by Pankaj Jalote , 1991 Narosa.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

Skill Oriented Course – II
MCA I Year II Semester

22MCAP602 PROGRAMMING WITH MATLAB

L	T	P	C
1	0	2	2

Pre-requisite 20MAT101, 20EEE101

Course Description:

This course introduces students to MATLAB programming, and demonstrate its use for scientific computations. The basis of computational techniques is expounded through various coding examples and problems. The practical ways to use MATLAB will be discussed.

Course Objectives:

This course enables students to

1. Understand basic MATLAB commands and elementary functions
2. Study and implement mathematical operations and matrices manipulation
3. Understand MATLAB functions and expressions
4. Apply flow control and files in MATLAB
5. Understand Plotting and Simulink blocks in MATLAB

UNIT I MATLAB BASICS 6 hours

Introduction, Matlab environment, Matlab as a calculator, Matlab Online, Syntax and Semantics, Help, Data Types-Matrix, string, cell and structure, Variables and Arrays, **Initializing Variables,** Multidimensional Arrays, Sub arrays, Special Values, Displaying Output Data, Data Files, Scalar and Array Operations, Hierarchy of Operations, Built-in MATLAB Functions, Debugging MATLAB Programs

- Swap the values in two variables without using temporary variable. For example, the variable ‘x’ contains the value ‘5’ and the variable ‘y’ contains the value ‘10’. The program should swap the values in the variable’s ‘x’ and ‘y’. After the execution of the program the value in the variable ‘x’ should be ‘10’ and the value in the variable ‘y’ should be ‘5’. This should be accomplished without using the temporary variable.
- Write a function which should return either maximum or minimum value of the element in an array.
- Write a code to find whether the given number is even or not.
- Write a function that should sort the elements in the array either in the ascending order or descending order.
- Write a program which should count the number of occurrences of particular element in the array.

UNIT II MATRICES AND OPERATORS 6 hours

Introduction, Colon Operator, Accessing Parts of a Matrix, Combining and Transforming Matrices, Arithmetic operations

- Write a program to find the maximum and minimum value of the elements of the matrix
- Write a program to compute the sum of diagonal elements of the given matrix
- Write a program to test whether the given matrix is symmetric or not?
- Obtain the rank of the following matrices (i) $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ (ii) $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$ and comment on the result.
- Write a program to check whether the given matrix is invertible or not? {Hint: A matrix is invertible if is not singular. The determinant of the matrix should not be equal to zero}
- Write a program to check the given matrix is orthogonal or not?
- Use the built-in function to compute the eigen value and the eigen vector of the given matrix. From the eigen value is it possible to find whether the given matrix is (i) Positive definite (ii) Positive semidefinite.

- Create a vector ‘x’ that should contain elements from 1 to 10. Write a code to perform the following operation
  (i) Add a constant (say 3) to each element of ‘x’.
  (ii) Make all the even indexed elements to zero.
  (iii) Make all the odd indexed elements to zero.
  (iv) Generate ‘y’ which should contain elements in the reverse order of ‘x’.
  (v) Generate ‘y’ such that it should have first five elements of ‘x’ and the remaining elements to zero
  (vi) Add the constant to odd indexed elements of ‘x’.
  (vii) Add the constant to the even indexed elements of ‘x’
- Write a program to solve the linear algebraic equation
  (i) 5x-3y+2z = 10
  (ii) -3x+8y+4z = 20
  (iii) 2x+4y-9z = 9
- Write a program to determine the eigen vector and eigen values of A = [1 2 3; 4 5 6; 7 8 9]

UNIT III FUNCTIONS AND EXPRESSIONS 6 hours

Introduction, Function I/O, Formal Definition of Functions, Sub functions, Scope, Advantages of Functions, Scripts, and Problem-Solving **File Input-Output, Expressions,** write a function which returns the “median” of the array of elements.

- Write a code to print the prime numbers from one to hundred.

- Write a function which accepts the radius of the circle as input and returns the area and perimeter of the circle.
- Write a code which will compute sum of integers ranging from 1 to 100.
- Write a code to compute the “body mass index”. The input to the code should be (i) Weight and (ii) Height of the person. The output of the program should be “body mass index (bmi)”
- Write a program to convert the temperature in degrees to Celsius.
- Write a program to check whether the given string (word) is palindrome or not?
- Write a program to compute the factorial of the given number.
- Find the roots of the polynomial

$$f(x) = 3x^6 + 15x^5 + 10x^3 + 4x$$

- An R-L-C circuit has R = 180 ohms, C = 1/280 farads, L = 20 Henries and an applied voltage E(t) = 10 sin t. Assuming that no charge is present but an initial current of I ampere is flowing at t = 0 when the voltage is first applied, find q and $i = \frac{dq}{dt}$ at any time t. q is given by the differential equation.

$$L\frac{d^2q}{dt^2} + R\frac{dq}{dt} + \frac{q}{c} = E(t)$$

- The function sin(x) can be written as a Taylor series by:

$\sin x = \sum_{k=0}^{\alpha} \frac{(-1)^k x^{2k+1}}{(2k+1)!}$ Write a user-defined function file that calculates sin(x) by using the Taylor series.

UNIT IV FLOW CONTROL AND FILES 6 hours

For – Loops, While – Loops, Break Statements, Logical Indexing, Pre allocation. Data Types: Introduction, Strings, Structs, Cells. Selection, If – Statements, Relational and Logical Operators, Nested If – Statements, Variables Number of Function, Arguments, Robustness, Persistent Variables. switch and case statement, while statement, break, Continue. **Files- File Input/ Output: File I/O, Excel Files, Text Files, Binary Files.**

- Without using the **max** command, find the maximum value of matrix (a) where a =[11 3 14;8 6 2;10 13 1]

- Let x=[2 6; 1 8], y=[.8 -0.3 ; -0.1 0.2], prove that y is not the inverse matrix of x
- The value of s could be calculated from the equation below:

$$s = \begin{cases} \sqrt{y^2 - 4xz} & if \;\; y \geq 4xz \\ \alpha & if \;\; y > 4xz \end{cases}$$

write a MATLAB program in M-File to do the following steps: -

a) input the value of x, y, z

b) calculate s

c) print the output as shown below

```
x = . . .
y = . . .
z = . . .
s = . . .
```

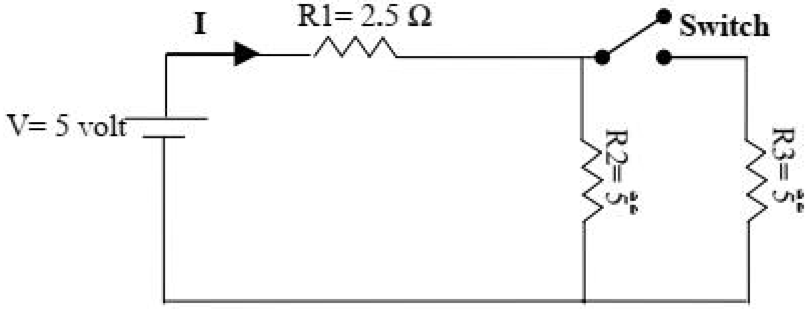
- Use a for-end loop in a script file to calculate the sum of the first n terms of the series:

$$\sum_{k=1}^{n} \frac{(-1)^k k}{2^k}$$ Execute the script file for n = 4 and n = 20.

- Write a program to find the current I in the circuit shown below

a) By using conditional statements.

b) Without using any conditional statements.



UNIT V PLOTTING AND SIMULINK 6 hours

Two – Dimensional Plots - Plot, fplot, Multiple Graphs, Formatting, Logarithmic Axes, Error Bars, Special Graphics, Histograms, Polar Plots, Multiple Plots on The Same Page, Multiple Figure Windows, **Three-Dimensional Plots-** Line Plots, Mesh and Surface Plots, Special Graphics, View Command. **Simulink:** Getting Started, Simulink Library Browser, Basic Elements-Blocks, Lines, building a System-Gathering Blocks, Modifying the Blocks, Connecting the Blocks, Running Simulations, Specification, Toolboxes, Building Systems.

- The expression for sine wave is given by $x(t) = A\sin(2\pi ft + \phi)$ .Write a code which accepts the input as (i) Amplitude (A) (ii) Frequency (f) and (iii) Phase(ϕ) and generates the sine wave. Plot the sine wave.
- Write a program to convert the sine wave to (i) Half wave rectified sine wave and (ii) Full wave rectified sine wave.
- Write a program which converts the sine wave to a square wave [Equivalent to that of “zero-crossing detector” or “comparator” concept in “Linear Integrated Circuits”].
- Write a program to generate three-phase sinusoidal signal. [The student should know what is the phase difference between three phases in a three-phase sinusoidal signal and the importance of three phase power]
- Design a Simulink block for power electronic circuits

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Interpret the MATLAB commands and elementary functions
2. Solve mathematical operations and matrices manipulation
3. Apply MATLAB functions and expressions
4. Execute implementation of flow controls and files in MATLAB
5. Demonstrate Plotting and Simulink blocks in MATLAB.

Text Books:

1. Getting Started with MATLAB, Rudra Pratap Oxford University Press, 1st edition, 2019
2. MATLAB for Beginners: A Gentle Approach, Kattan, Peter Issa, Petra books, 2008

Reference Books:

1. MATLAB for Engineering Applications, William Palm, Mcgraw Hill,4th edition, 2019.
2. MATLAB for Engineers**,** Holly Moore, Pearson Education,5th edition,2018

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

DISCIPLINE ELECTIVE-I

DISCIPLINE ELECTIVE-I

MCA I Year II Semester

22MCAP401 MACHINE LEARNING – ALGORITHMS AND APPLICATIONS

L	T	P	C
3	0	0	3

Pre-requisite Algorithm Design & Programming, Probability & Statistics

Course Description:

This course provides a concise introduction to the fundamental concepts in machine learning and popular machine learning algorithms. This will cover the standard and most popular supervised learning algorithms including linear regression, logistic regression, decision trees, k-nearest neighbour, an introduction to Bayesian learning and the naïve Bayes algorithm, support vector machines and neural networks.

Course Objectives:

Students will be able to:

1. Understand the fundamentals of machine learning concepts.
2. Analyse Regression and Classification methods.
3. Identify the Reinforcement Learning algorithms and its applications
4. Investigate Clustering methods for real world problems
5. Demonstrate Artificial Neural Networks for various inputs and expected outputs

UNIT I INTRODUCTION 9 hours

Well posed learning problems, Designing a Learning system, Perspectives and issues in Machine Learning, Types of Machine Learning, Bayes theorem and Concept learning, Maximum likelihood and least squared Error hypotheses

UNIT II REGRESSION & CLASSIFICATION ALGORITHMS 9 hours

Simple Linear Regression, Multiple Linear Regression, Classification Methods- Logistic regression, Nearest neighbor Classifier- Decision trees- Support Vector Machine, Genetic Algorithm.

UNIT III REINFORCEMENT ALGORITHMS 9 hours

Reinforcement Learning – Components of Reinforcement, Types of Reinforcement, Applications, Reinforcement Learning Algorithms-Value-based, Policy-based, Model-based, Characteristics, Markov Decision Process, Q Learning.

UNIT IV CLUSTERING ALGORITHMS 9 hours

Clustering Methods-Partitioned based Clustering – K means, K medoids; Hierarchical Clustering – Agglomerative, Divisive, Distance measures; Density based Clustering, Gaussian model

UNIT V ARTIFICIAL NEURAL NETWORK ALGORITHMS 9 hours

Neural network representation, Perceptron, Multi Layer Perceptron with Back propagation, An Illustrative example: Face Recognition, Advance topics in Artificial Neural Networks

Course Outcomes:

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

1. Select real-world applications that needs machine learning based solutions.
2. Apply supervised Machine Learning algorithm for Classification and Regression.
3. Identify the Reinforcement Modes for Real Time Predictions
4. Implement the appropriate Clustering real time algorithms for real world problems
5. Analyse the Artificial Neural Network for real time Applications

Text Book(s)

1. Aurelien Geron, *Hands on Machine Learning with Scikit -Learn , Keras and Tensor Flow Concepts, Tools and Techniques to build intelligent Systems*,2nd Edition by O'Reilly Media Publishers, 2019.
2. Tom Michel , "Machine Learning", Mc.Graw Hill, Indian Edition, 1997.

Reference Books

1. Mohri, Rostamizadeh and Talwalkar, "Foundations of Machine Learning", MIT Press, 2012
2. Laura Igual Santi Seguí, *Introduction to Data Science, A Python approach to concepts and Applications*, Springer-2017.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP402 IoT TECHNOLOGY AND APPLICATIONS

L	T	P	C
3	0	0	3

Pre-requisite Computer Network, Artificial Intelligence

Course Description:
Internet of Things is transforming our physical world into a complex and dynamic system of connected devices on an unprecedented scale. This course introduces the IoT technologies, protocols, and designing IoT systems with hardware basis. It provides basic information about Data Analytics and its Supporting Services. It imparts the Applications on industry, healthcare and smart cities.

Course Objectives:
Student will be able to:

1. Understand Smart Objects and IoT Architectures.
2. Identify the various IOT-related protocols
3. Implement IoT applications using Arduino and Raspberry Pi.
4. Analyze the data analytics and cloud systems in the context of IoT
5. Classify the IoT infrastructure for popular applications.

UNIT I FUNDAMENTALS OF IoT 9 hours

Evolution of Internet of Things – Enabling Technologies – IoT Architectures: oneM2M, IoT World Forum (IoTWF) and Alternative IoT models – Simplified IoT Architecture- Core IoT Functional Stack– Smart Objects – Sensors, Actuators, Sensor Networks and Connecting Smart Objects

UNIT II IoT PROTOCOLS 9 hours

IoT Access Technologies: Physical and MAC layers, topology and Security of IEEE 802.15.4, 802.15.4g, 802.15.4e, 1901.2a, 802.11ah and LoRaWAN – Network Layer: IP versions, The Key Advantages and Adoption of the Internet Protocol, Constrained Nodes and Networks -Optimizing IP for IoT: 6LoWPAN, Header Compression, Fragmentation , Mesh Addressing, Routing over Low Power and Lossy Networks – **Application Transport Methods**: SCADA – IoT Application Layer Protocols: CoAP and MQTT.

UNIT III BUILDING IoT WITH RASPBERRY PI AND ARDUINO 9 hours

Design Methodology – Embedded computing logic - Building IOT with RASPBERRY PI- IoT Systems– Logical Design using Python – IoT Physical Devices & Endpoints - IoT Device -Building blocks -Raspberry Pi -Board - Linux on Raspberry Pi - Raspberry Pi Interfaces -Programming Raspberry Pi with Python- IoT Design and Prototyping Using Arduino Boards.

UNIT IV DATA ANALYTICS AND SUPPORTING SERVICES 9 hours

Data Analytics for IoT: Structured Vs Unstructured Data, Data in Motion Vs Data in Rest – Role of Machine Learning – No SQL Databases – Hadoop Ecosystem – Apache Kafka, Apache Spark, lambda – Edge Streaming Analytics and Network.

UNIT V APPLICATIONS AND CASE STUDIES 9 hours

Applications: Smart and Connected Cities-Underlying Technologies for Smart Cities, Driverless Vehicles, Smart Buildings Smart Campuses, Smart Grid- Health Care- Agriculture: Crop Water Stress Index, IoT Irrigation System, Crop Disease and Pest Management

Case Study: Smart City Streetlights Control and Monitoring

Course Outcomes:

After completion of the course, students will be able to

1. Explain the term 'internet of things' in different contexts.
2. Identify the various protocols for IoT.
3. Design an IoT applications using Raspberry Pi/Arduino
4. Incorporate the data analytics frameworks and services
5. Analyze the applications of IoT in real time scenario.

Text Book(s)

1. IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, Cisco Press, 2017.
2. Hassan, Qusay F., ed. Internet of things A to Z: Technologies and Applications. John Wiley & Sons, 2018.

Reference Books

1. Arshdeep Bahga, Vijay Madisetti, “Internet of Things – A hands-on approach”, Universities Press, 2015.
2. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), “Architecting the Internet of Things”, Springer, 2011.
3. Olivier Hersent, David Boswarthick, Omar Elloumi , ―The Internet of Things – Key applications and Protocols, Wiley, 2012
4. Jan Ho¨ ller, Vlasios Tsiatsis , Catherine Mulligan, Stamatis , Karnouskos, Stefan Avesand. David Boyle, “From Machine-to-Machine to the Internet of Things – Introduction to a New Age of Intelligence”, Elsevier, 2014.
5. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), ―Architecting the Internet of Things, Springer, 2011.
6. Michael Margolis, Arduino Cookbook, Recipes to Begin, Expand, and Enhance Your Projects, 2nd Edition, O’Reilly Media, 2011.
7. Kamal, Raj. Internet of Things- Architecture and Design Principles. McGraw-Hill Education, 2017.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP403 AGILE SOFTWARE DEVELOPMENT PROCESS

L	T	P	C
3	0	0	3

Pre-requisite Software Engineering

Course Description:

Object oriented analysis, UML, Agile software development and Frameworks of Agile development. DevOps is fundamental for developers and application oriented personalities. Improved workflows and faster deployment will be helpful for developers in their project implementation.

Course Objectives:

Students will be able to:

1. Explain the object oriented techniques used in the real world software industries.
2. Acquire knowledge about the classes, objects and UML diagrams.
3. Compare agile with traditional software development and Contrast different agile methodologies.
4. Analyze the DevOps relationship to Agile, Lean and ITSM.
5. Improve workflows and speed up deployment.

UNIT I INTRODUCTION TO OBJECT ORIENTED ANALYSIS 9 hours

Introduction: SDLC & its variations, Approaches to system development, SSAD, OOAD, Unified Process and its characteristics, Unified Process life cycle, Basics of Object Orientation- OO development and its themes; Modelling and usefulness of OO development

UNIT II OBJECT ORIENTED MODELING AND UML 9 hours

Object Oriented Modeling-Capturing, modeling, documenting system requirements, and Object oriented basic components; Dynamic Modeling- Use-case Modeling and Various UML diagrams.

UML: Introduction to UML, UML Building Blocks. Conceptual model of UML, Architecture, Classes, Relationships, Common Mechanisms, Class-Object-Sequence-Activity-Use Case Diagrams.

UNIT III INTRODUCTION TO AGILE AND ITS SIGNIFICANCE 9 hours

Software Development Process. Iterative and Evolutionary Methods. Agile software development Vs other traditional methods. Agile Manifesto: Principles, and Benefits of agile development, User Stories, Generating User Stories. Agile Story: Evolutionary delivery, Scrum Demo, Planning game, Sprint back log, adaptive planning. Agile Motivation Problems with the Waterfall - Research Evidence; Agile roles

UNIT IV AGILE FRAMEWORK 9 hours

Method Overview, Life cycle phases and Work product roles and practices- Scrum, Extreme Programming., Unified process, EVO

UNIT V INTRODUCTION OF DEVOPS 9 hours

Continuous Delivery, and the three ways The Principles of Flow Feedback Continual Learning and Experimentation. Selecting which value stream to start with Understanding the Work in our value stream Making it visible Expanding - Design Organization and architecture Outcomes by Integrating Operations.

Course Outcomes:

After completion of the course, students will be able to

1. Analyze the object oriented techniques.
2. Draw UML diagrams for real-world project components.
3. Identify the significance of agile development over the traditional approaches.
4. Examine the different frameworks of agile software development approaches.
5. Analyze DevOps and its relationship with agile development.

Text Book(s)

1. Object Oriented Analysis and Design with Applications by Grady Booch Robert, Third Edition.
2. Agile Software Development Best Practices for Large Software Development Projects by Thomas Stober, and Uwe Hansmann.
3. Shore, James. The Art of Agile Development: Pragmatic guide to agile software development. "O' Reilly Media, Inc.", 2007.
4. The DevOps Handbook by Gene Kim, Jez Humble, Patrick Debois and John Willis.

Reference Books

1. Object - Oriented Modeling and Design with UML 2nd Edition.
2. Agile Software Development, Principles, Patterns, and Practices by Robert C. Martin
3. Effective DevOps, by Jennifer Davis, Ryn Daniels.
4. DevOps for Web Development by MiteshSoni.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP404 COMPUTER GRAPHICS AND MULTIMEDIA

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

Computer Graphics and Multimedia is to teach students theory, technology, procedures, and skills in computer graphics and multimedia. The compulsory courses are intended for improvement of both basic and in-depth knowledge of image synthesis in computer graphics, speech processing and recognition, sound and video sequences algorithms for multimedia, and development of systems for human-computer interaction.

Course Objectives:

1. Understand the fundamental principles of Computer Graphics concepts.
2. Identify the 3D transformation, image synthesis and shape modelling for #D applications
3. Understand the basic concepts related to Multimedia including data standards, algorithms and software.
4. Classify the multimedia communication technologies
5. Examine the Multimedia applications.

UNIT I BASIC CONCEPTS 9 hours

2D Transformations – Clipping – Point Clipping – Line Clipping – Polygon Clipping – Text Clipping – Exterior Clipping – Window to View Port Mapping – Interactive Input Methods – Picture Construction Techniques.

UNIT II 3D GRAPHICS 9 hours

3D Concepts – 3D Transformations – 3D Viewing – Visible Surface Detection Methods – Back Face Detection Method – Depth Buffer Method – Scan Line Method –Virtual Reality Environment.

UNIT III MULTIMEDIA BASICS 9 hours

Introduction to Multimedia – Applications– Hypermedia – Authoring –– File formats –Color Models – Digital Audio– Digital Music Making – MIDI – Digital Video – Video Compression Techniques – Video Performance Measurements –Multimedia Databases–Animation.

UNIT IV MULTIMEDIA COMMUNICATION 9 hours

Multimedia Network Services–Network Protocols–Requirements for Multimedia Communications – Multimedia Conferencing Architectures –QuickTime Movie File Format– MHEG–Multimedia File Sharing –Multimedia & Internet–Real-Time Interchange.

UNIT V MULTIMEDIA APPLICATION DEVELOPMENT 9 hours

Design of a Multimedia System –Content Based Information Retrieval – HDTV, ATV, EDTV, IDTV Standards –Development of User Interface Design – Multimedia Broadcasting –Social Media Sharing – Multimedia Development Issues – Sample Multimedia Project.

Course Outcomes:

After successful completion of the course, students will be able to

1. Understand the various algorithms of 2D Computer graphics.
2. Infer the 3D graphics modelling technology.
3. Understand different forms of Multimedia and gain knowledge about Audio and Video.
4. Classify the Networks used for Multimedia and communication Applications.
5. Investigate the various Multimedia Applications.

Text Book(s)

1. Donald Hearn and M. Pauline Baker, "Computer Graphics C Version", Second Edition, Pearson Education.
2. David Hillman, "Multimedia – Technology and applications",Galgotia Publications, Delhi, 2008.

Reference Books

1. Ralf Steinmetz and Klara "Multimedia Computing, Communications and Applications", Pearson Education,2009
2. Tom McReynolds – David Blythe," Advanced Graphics Programming Using OpenGL", Elsevier, 2005

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP405 IMAGE PROCESSING

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:
This course is designed to provide basic understanding on image processing. The course material further used for developing any digital image processing applications. Course covers image enhancement methods, image restoration techniques, multi-resolution analysis, image segmentation models, feature extraction methods, supervised, unsupervised algorithms and applications.

Course Objectives:
Student will be able to:

1. Understand the basic concepts of digital image processing.
2. Classify the image enhancement techniques.
3. Explain the various image processing techniques and their applications.
4. Analyse the use of current technologies in image processing systems.
5. Investigate real world applications of image processing.

UNIT I FUNDAMENTALS OF IMAGE PROCESSING 9 hours

Introduction – Steps in Digital Image Processing – Components – Elements of Visual Perception – Image Sensing and Acquisition – Image Sampling and Quantization – Relationships between pixels and color models – Image operations.

UNIT II IMAGE ENHANCEMENT 9 hours

Image Transforms: Discrete Fourier Transform – Fast Fourier Transform – Discrete Cosine Transform – Image Enhancement in Spatial and Frequency Domain – Grey Level Transformations – Histogram Processing –Spatial Filtering – Smoothing and Sharpening –Frequency Domain: Filtering in Frequency Domain.

UNIT III IMAGE RESTORATION AND MULTI-RESOLUTION ANALYSIS 9 hours

Multi Resolution Analysis: Image Pyramids – Multi Resolution Expansion – Wavelet Transforms – Image Restoration – Image Degradation Model – Noise Modelling – Blur – Order Statistic Filters – Image Restoration Algorithms.

UNIT IV IMAGE SEGMENTATION AND FEATURE EXTRACTION 9 hours

Image Segmentation – Detection of Discontinuities –Edge Operators –Edge Linking and Boundary Detection – Thresholding – Region based Segmentation – Image Features and Extraction – Image Features – Types of Features – Feature Extraction – SIFT, SURF and Texture – Feature Reduction Algorithms.

UNIT V IMAGE PROCESSING APPLICATIONS 9 hours

Image Classifiers – Supervised Learning – Support Vector Machines, Image Clustering – Unsupervised Learning – Hierarchical and Partition based Clustering Algorithms – EM Algorithm

Course Outcomes:

After successful completion of the course, students will be able to

1. Understand basic image processing operations.
2. Apply new techniques in the areas of image enhancement and restoration.
3. Understand the image segmentation algorithms.
4. Extract features from images by using classifiers and clustering algorithms.
5. Design image processing application that uses different concepts of image processing.

Text Book(s)

1. Rafael Gonzalez, Richard E. Woods, “Digital Image Processing”, Fourth Edition, Pearson Education, 2018.
2. Milan Sonka, Vaclav Hlavac, Roger Boyle, “Image Processing Analysis and Machine Vision”, Fourth Edition, Cengage India, 2017.

Reference Books

1. S. Annadurai, S. Shanmugalakshmi, “Fundametals of Digital Image Processing”, Pearson, 2006.
2. Anil K. Jain, “Fundamentals of Digital Image Processing”, PHI, 2011.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year II Semester

22MCAP406 BLOCKCHAIN TECHNOLOGY

L	T	P	C
3	0	0	3

Pre-requisite: 22MCAP105

Course Description:

This course is designed for the understanding of blockchain technology and cryptocurrencies. Upon completion of the programme students will able to develop secure applications using blockchain concepts, design and develop automated smart contracts, implementation of industry-specific blockchain models and frameworks. Possible career opportunities upon completion of the programme are Blockchain business analyst, Blockchain Consultant, Blockchain Solution Architect, Cryptocurrency Developer, etc.

Course Objectives:

The course is designed to meet the objectives of:

1. Understand the blockchain architecture and applications
2. Identify the problems of centralization
3. Explain the concept of Concept of Double Spending
4. Understand the concept of Bitcoin and its Applications
5. Understand the basic concepts of Ethereum and its regulations

UNIT I INTRODUCTION TO BLOCKCHAIN 9 hours

Introduction to Blockchain, How Blockchain works, Blockchain vs Bitcoin, Practical applications, public and private key basics, pros and cons of Blockchain, Myths about Bitcoin.

UNIT II BLOCKCHAIN 9 hours

Architecture, versions, variants, use cases, Life use cases of blockchain, Blockchain vs shared Database, Introduction to cryptocurrencies, Types, Applications.

UNIT III CONCEPT OF DOUBLE SPENDING 9 hours

Concept of Double Spending, Hashing, Mining, Proof of work. Introduction to Merkel tree, Privacy, payment verification, Resolving Conflicts, Creation of Blocks.

UNIT IV INTRODUCTION TO BITCOIN 9 hours

Introduction to Bitcoin, key concepts of Bitcoin, Merits and De Merits Fork and Segwits, Sending and Receiving bitcoins, choosing bitcoin wallet, Converting Bitcoins to Fiat Currency.

UNIT V INTRODUCTION TO ETHEREUM 9 hours

Introduction to Ethereum, Advantages and Disadvantages, Ethereum vs Bitcoin, Introduction to Smart contracts, usage, application, working principle , Law and Regulations. Case Study.

Course Outcomes:

After successful completion of the course, students will be able to

1. Demonstrate the basics of Block chain concepts using modern tools/technologies.
2. Analyze the application of specific block chain architecture for a given problem.
3. Evaluate the usage of Block chain implementation/features for the given problem.
4. Exemplify the usage of bitcoins and its impact on the economy.
5. Understand the basic concepts of Ethereum and its regulations.

Text Book(s)

1. Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions by ArshdeepBikramaditya Signal, GautamDhameja (PriyansuSekhar Panda., APress.
2. Blockchain Applications: A Hands-On Approach by Bahga, Vijay Madisetti
3. Blockchain by Melanie Swan, OReilly

Reference Books

1. Bitcoin and Cryptocurrency Technologies by Aravind Narayan. Joseph Bonneau, princton
2. Bitcoin and Blockchain Basics: A non-technical introduction for beginners by Arthu.T Books.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA I Year I Semester

22MCAP407 XML AND WEB SERVICES

L	T	P	C
3	0	0	3

Pre-requisite Web designing concepts, HTML concepts

Course Description:

This course provides XML technology family, architecture of web services, the building blocks of Web services, the role of XML in E-business and content management.

Course Objectives:

Student will be able to:

1. Understand the overview of XML family and related technologies
2. Learn the concepts of Web Services Architecture
3. Understand the ideas on Web services building blocks
4. Identify the implementation of XML in E-Business
5. Understand the concepts of content management in XML

UNIT I XML TECHNOLOGY FAMILY 9 hours

XML – benefits – Advantages of XML over HTML – EDL –Databases – XML based standards – DTD –XML Schemas – X- Files – XML processing – DOM –SAX presentation technologies – XSL – XFORMS – XHTML – voice XML – Transformation – XSLT – XLINK – XPATH –XQ

UNIT II ARCHITECTING WEB SERVICES 9 hours

Business motivations for web services – B2B – B2C- Technical motivations – limitations of CORBA and DCOM – Service – oriented Architecture (SOA) – Architecting web services – Implementation view – web services technology stack – logical view – composition of web services – deployment view – from application server to peer to peer – process view – life in the runtime

UNIT III WEB SERVICES BUILDING BLOCK 9 hours

Transport protocols for web services – messaging with web services – protocols – SOAP – describing web services – WSDL – Anatomy of WSDL – manipulating WSDL – web service policy – Discovering web services – UDDI – Anatomy of UDDI- Web service inspection – Ad-Hoc Discovery – Securing web services.

UNIT IV IMPLEMENTING XML IN E-BUSINESS 9 hours

B2B - B2C Applications – Different types of B2B interaction – Components of ebusiness XML systems – ebXML – Rosetta Net Applied XML in vertical industry – Web services for mobile devices

UNIT V XML AND CONTENT MANAGEMENT 9 hours

Semantic Web – Role of Meta data in web content – Resource Description Framework – RDF schema – Architecture of semantic web – content management workflow – XLANG –WSFL.

Course Outcomes:

After completion of the course, students will be able to

1. Illustrate the ideas on XML concepts
2. Explain the architecture of web services.
3. Analyze web designing building blocks
4. Explain the implementation ideas XML in E-Business
5. Illustrate the XML content management system

Text Book(s)

1. Ron schmelzer et al, “XML and Web Services”, Pearson Education, 2002.
2. Sandeep Chatterjee and James Webber, “Developing Enterprise Web Services: An Architect’s Guide”, Prentice Hall, 2004

Reference Books

1. Henry Bequet and Meeraj Kunnumpurath, “Beginning Java Web Services”, Apress, 2004
2. Russ Basiura and Mike Batongbacal, “Professional ASP.NET Web Services”, Apress, 2003.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP111 MOBILE APPLICATION DEVELOPMENT

L	T	P	C
4	0	0	4

Pre-requisite **Object Oriented Programming Concepts**

Course Description:

This course is designed to do the development of applications on Android platform. Android is used as a basis for the development of mobile applications. This course starts with the basic concepts of Java, history of android and architecture. It introduces the major building blocks that are used to develop an android application. It also covers the development of applications using widgets, events, networking.

Course Objectives:

This course enables students to

1. Understand the need and characteristics of mobile applications.
2. Creation of android user interface and its testing mechanisms
3. Build the design and development of mobile applications.
4. Know about the storage mechanism in android using SQLite.
5. Develop mobile applications using sensors.

UNIT I INTRODUCTION ABOUT ANDROID TOOLS 12 hours

Android Overview – History – Android Versions - Android Flavors. Android Stack: Linux, Native Layer and Hardware Abstraction Layer (HAL) – ART Installation and Use of Android Tools: Installing the Android SDK - Anatomy of an Android Project - Drawable Resources Mobile Application Model- Overview of Android Building Blocks –Logging Messages in Android.

UNIT II USER INTERFACE 12 hours

Example. Input Components – Text View – Image View – List View and Alert Dialogues – Menus: Popup, Options and Context Menus – Screen Navigation through App Bar – Recycler View – Material Design –Testing the User Interface: Expresso – Screen Navigation using Intents: Definition – Usage of Intends – Creation of Indents with example program – Lists and Adapters – Types of Adapters – Examples using Adapters

UNIT III APPLICATION DESIGN 12 hours

Threading in Android – AsyncTask – Loaders – AsyncTask Loader – Connecting to Internet: JSON - HTTP API, Apache HTTP Client, HTTP URL Connection - Broadcast Receivers: Custom Broadcasts – Broadcasting Intends and their related API - Boot Receiver - Alarms and system services – Examples on alarms and services – Services: Services Life Cycle – Intent Service – Implementing Intent Service – Notifications: Managing Notifications

UNIT IV ANDROID FILE SYSTEM 12 hours

Android File systems and Files - Action Bar: Preferences and Action Bar - Shared Preferences – App Settings - Databases on Android - SQLite - Status Contract Class, Update Refresh Service – Cursors – Backups - Content Providers: Overview – Role of Content Providers - Content Provider Example Program – Content Resolver

UNIT V APPLICATION DEVELOPMENT AND SENSOR 12 hours

App Widgets: Creation of Application Widgets - Interaction and Animation: Live Wallpaper and Handlers - Sensors: Sensor API in Android - Motion Sensor, Position Sensor, Environmental Sensor, Sensor Values, Sensor Manager Class, Sensor Class, Sensor Event class, Sensor Event Listener interface, Compass Accelerometer and orientation Sensors, Sensor Examples

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand the basics of mobile application development and Install android.
2. Develop a UI for mobile applications.
3. Explain the Broadcast Receivers and Services.
4. Utilize the Database in Android Applications.
5. Build widgets, Wall papers for an android application using sensors.

Text Book(s)

1. Android Programming-The Big Nerd Ranch Guide, Bill Philips, Christ Stewart, Kristin Mariscano, Big Nerd Ranch publishers, 3rd Edition**,** 2022.
2. Android Programming for Beginners, John Horton, PACKT publishers, 3rd Edition**,** 2021.

Reference Books

1. Reto Meier, Ian Lake, “Professional Android”, 4th Edition, Wrox, 2018.
2. Zigurd Mednieks, Laird Dornin, G. Blake Meike, Masumi Nakamura, “Programming Android”, O’Reilly, 2nd Edition, 2012.
3. Alasdair Allan, “Learning iOS Programming”, O’Reilly, Third Edition, 2013.
4. Bill Phillips, Chris Stewart, Brian Hardy, and Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, 4th edition, 2019.
5. Christian Keur, Aaron Hillegass, iOS Programming: The Big Nerd Ranch Guide, 6th Edition, O’Reilly, 2016.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP112 FUNDAMENTALS OF DATA SCIENCE

L	T	P	C
3	1	0	4

Pre-requisite **Mathematical Foundations of Computer Applications**

Course Description:

This course will introduce data science basic principles. Students will learn concepts and techniques they need to deal with various facts of data science practice, including data collection and integration, exploratory data analysis, predictive modeling, descriptive modeling, forecasting concepts, neural networks, and text mining.

Course Objectives:

This course enables students to

1. Understand Exploratory Data Analysis.
2. Illustrate Unsupervised Learning concepts.
3. Explain Supervised Learning concepts.
4. Demonstrate Forecasting methods.
5. Describe Neural network concepts and Text Mining methods.

UNIT I INTRODUCTION TO DATA SCIENCE 12 hours

Introduction to Data Science- Definition and scope of data science - Impact of Data Science - Data science workflow and Life Cycle - Data Science Toolkit.

Data Acquisition and Preprocessing: Data sources and types - Data collection methods - Data cleaning and preprocessing techniques.

Exploratory Data Analysis (EDA): Descriptive statistics - Data visualization for EDA - Handling missing data and outliers.

UNIT II DATA ANALYSIS 12 hours

Data Manipulation with Python: Introduction to Python libraries for data analysis (NumPy, Pandas) - Data manipulation and transformation - Data aggregation and summarization.

Machine Learning Algorithms: Regression algorithms (linear regression, decision trees) - Classification algorithms (logistic regression, random forests) - Clustering algorithms (k-means, hierarchical clustering).

UNIT III DATA INTEGRATION 12 hours

Data Integration and Transformation: Introduction to data integration - Data cleaning and preprocessing techniques - Handling missing data and duplicates.

Data Wrangling and Feature Engineering: Data transformation and reshaping - Feature extraction and selection - Handling categorical and textual data.

Data Integration Techniques: Joining and merging datasets - Entity resolution and record linkage - Dealing with data inconsistencies and conflicts.

UNIT IV DATA VISUALIZATION 12 hours

Principles of Data Visualization: Introduction to data visualization - Perceptual and cognitive principles - best practices and design considerations, Data Visualization Tools : Tableau, Power BI - Python libraries for visualization (matplotlib, seaborn, Plotly).
Static and Interactive Visualization: Creating static visualizations (bar charts, line plots, scatter plots) - Interactive visualizations (dashboards, interactivity, user engagement).
Geospatial Visualization: Mapping techniques and libraries (GeoPandas, Leaflet) - Visualizing geographic data and spatial relationships.

UNIT V APPLICATIONS 12 hours

Time Series Analysis and Forecasting: Introduction to time series data – Time Series Forecasting methods (Naïve, AR, MA, ARMA, ARIMA, Exponential Smoothing).
Natural Language Processing (NLP): Introduction to NLP and text mining - Text preprocessing and feature extraction - Sentiment analysis and text classification.
Case Studies and Applications: Application of data science in various domains (finance, healthcare, marketing).

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Examine Exploratory Data Analysis for given data set.
2. Implement Unsupervised Learning algorithms for the given data set.
3. Apply Supervised Learning algorithms for the given data set.
4. Design Forecasting methods.
5. Illustrate MLP using Back Propagation method and Text Mining methods.

Text Book(s)

1. Foster Provost and Tom Fawcett, Data Science for Business, 2013.

2. Laura Igual Santi Seguí, Introduction to Data Science, A Python approach to concepts and Applications, Springer-2017.

Reference Books

1. Joel Grus, Data Science from Scratch: First Principles with Python, O'Reilly Media, 2019.

2. Amir Aczel, Jayavel Souder Pandian, P. Saravanan, Complete Business Statistics, Mc Graw Hill publishers,2012.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP113 JAVA PROGRAMMING

L	T	P	C
3	1	0	4

Pre-requisites **Concepts of Object-Oriented Programming**

Course Description:

This course introduces core Java. It covers the history and evolution of java, basic constructs of the programming language. It gives exposure on arrays, string related classes, reusability of code, handling run time errors, multithreaded programming. It also introduces various classes and methods of input/output stream, networking, collections and swings.

Course Objectives:

Student will be able to:

1. Understand the evolution and basic constructs of Java.
2. Acquire knowledge on Arrays, Strings, code reusability and handling exceptions.
3. Understand the role of multithreading and applicability of wrapper classes.
4. Explore the utilities of I/O, Collections and Interface classes.
5. Build GUI through Swings and event handling.

UNIT I INTRODUCTION TO JAVA 12 hours

The Java Language: The History and Evolution of Java, Object and Class Definition, Encapsulation Inheritance and Polymorphism, Data Types, Variables, Operators, Control Statements, Methods and Classes, Method Overloading, Constructor, Constructor Overloading, Object as Parameters, Returning Objects, Recursion, Access Control, Static Keyword, final Keyword, Nested and Inner Classes

UNIT II ARRAYS, STRINGS AND EXCEPTION HANDLING 12 hours

Arrays, String and String Buffer Classes, Methods and Classes, Method Overloading, Super Keyword, Abstract Class and Final with Inheritance, The Object Class, Interface, Packages. Exception Handling: Fundamentals, Exception Types, Uncaught Exceptions, Try and Catch, Multiple catch Clauses, throw, throws, finally, Java Built-in Exceptions.

UNIT III MULTI-THREADING AND GENERICS 12 hours

Multithreaded Programming: The Java Thread Model, Thread Properties, The Main Thread, Life cycle of Thread, Creating Thread, creating Multiple Threads, using isAlive and join method, suspending, resuming and stopping threads. Generic methods, generic classes.

UNIT IV I/O STREAMS & COLLECTIONS FRAMEWORK 12 hours

Input/Output: The Java I/O Classes and Interfaces, The Character Streams: Reader, Writer, FileReader, FileWriter, CharArrayReader, CharArrayWriter, BufferedReader, BufferWriter, **The Collections Overview:** The Collection Interfaces, The Collection Classes – Collection Algorithms.

UNIT V SWINGS AND EVENT HANDLING 12 hours

GUI Programming with Java -Introduction to Swings, JLabel, JImageIcon, JTextField, JButton, JToggleButton, JCheckBoxes, JRadioButtons, JTabbedPane, JScrollPane, JList, JComboBox, JTrees, JTable. Event Handling: Event Handling Mechanisms, The Delegation Event Model: Events, Event Sources,Event Listeners, Event Classes, And Event Listener Interfaces.

Course Outcomes:

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

1. Implement java program using basic constructs of java.
2. Understand arrays, strings and inheritance.
3. Apply multithreaded and wrapper class concepts in developing applications.
4. Use various I/O, Collections and Interface classes appropriately.
5. Develop GUI components using Swings and event handling.

Textbook(s):

1. The Complete Reference Java to, Herbert Scheldt, 12th Edition, December 2021.

2. Core Java Volume I—Fundamentals, by Cay Horstmann, 11th Edition July 2020.

Reference Books:

1. Thinking in Java, Bruce Eckel, 4th Edition, 2008.

2. Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans,Kim Haase, William Markito, Java EE 7 Tutorial, 5th Edition, Prentice Hall, 2014.

3. E. Balaguruswamy, Programming with Java: A Primer, 3rd Edition, The McGraw Hill, 2012.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP206 MOBILE APPLICATION DEVELOPMENT LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite Object Oriented Programming Concepts

Course Description:

This course introduces the android application development. Emphasis is placed on the processes, tools and frameworks required to develop applications for current and emerging mobile computing devices. It covers the core components, android interface library, content providers, broadcast intents and various sensors.

Course Objectives:

Students will be able to

1. Understand the installation of Android SDK.
2. Get idea on the User Interface Design and their testing methodologies.
3. Know the components of Android Building Blocks and how to use them for different application developments.

List of Programs:

1. Develop an android application to display a simple text in the emulator

2. Develop an android application to display the internal keyboard in the emulator

3. Write an android program to display a message in the toast

4. Implement an application that creates an alert upon receiving a message

5.Develop an android application to process a student mark list by creating properUI using the necessary controls

6. Create a screen that has input boxes for User Name, Password, Address, Gender(radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button (use any layout)

7. Create an android UI that consists of Different Departments of a company namely Production, Finance, Marketing and HR. If the user clicks on any department it should show details of that department. Use indents.

8. Design an android application to display a list of items on the android screen. If the user clicks any one of the list items a dialogue box should show that the user has clicked that particular item (Use array adopters)

9. Develop an android application to show some categories such as education, entertainment, health, provisions etc., If the user clicks on any one of the items it should show the sub categories of the category and if is again clicked it should the details of those items. (Use indents and lists)

10. Develop an android application to demonstrate the database connectivity with the SQLite database to post and retrieve data through the User Interface (Example: Student mark list processing, Email Registration and Login, Products and sales)

11. Demonstrate the usage of Sensors in android by developing proper application.

Course Outcomes:

After completion of the above course, the student will be able to:

1. Install the Android SDK and write simple programs.
2. Write programs for different applications by creating different user interfaces.
3. Implement the Sensor programs and develop network connectivity.

Textbook(s)

1. Android Programming-The Big Nerd Ranch Guide, Bill Philips, ChristStewart, Kristin Mariscano, Big Nerd Ranch publishers, 3rd Edition, March 2017.

2. Android Programming for Beginners, John Horton, PACKT publishers, April 2021.

Reference Books

1. Android Application Development All in One for Dummies, Barry Burd, Wiley, 2nd Edition, 2015.

2. Android application Development-Black Book, Pradeep Kothari, Dreamtech, 2014.

3. Android Programming - Unleashed, B.M. Harwani, Pearson Education, 2013.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA II Year I Semester

22MCAP207 DATA SCIENCE USING PYTHON LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisites Basic Mathematics, Python Programming

Course Description:

This course will introduce the students to learn concepts and techniques to deal with various facts of data science practice including data collection and integration, exploratory data analysis, predictive modeling, descriptive modeling, and forecasting concepts.

Course Objectives:

Student will be able to:

1. Understand exploratory data analysis concepts.
2. Understand Supervised and Unsupervised Learning concepts.
3. Explain Time Series Forecasting methods and MLP.

List of Programs:

1. Implement Python program to clean the data by handling missing values, outliers, and inconsistencies in data frame.
2. Implement web scraping using libraries like BeautifulSoup or Scrapy to extract data from a website.
3. Implement Python code to perform descriptive statistics on a dataset, such as calculating measures of central tendency and variability for a numerical variable.
4. Implement simple and multiple linear regression model to predict a continuous target variable based on input features by using python.
5. Implement logistic regression model to train and evaluate on a labelled dataset by using python.
6. Implement python code to perform data wrangling and feature engineering.
7. Implement k means clustering algorithm for the given data set using python.
8. Implement hierarchical clustering algorithm for the given data set using python.
9. Create a dataset and visualize it with different customization options, such as colors, labels, titles, and annotations, to enhance the visual appeal of the following plots by using matplotlib, seaborn, and plotly library.
   (i) bar chart (ii) pie chart (iii) scatter plot and (iv) line plot
10. Implement geospatial data visualization using GeoPandas and the Leaflet library in Python.
11. Implement the visualization of time series forecasting methods by using python.
12. Implement sentiment analysis and classify it as positive, negative, or neutral by using python.
13. Implement healthcare data analysis in Python to identify or predict disease outcomes (affected or not affected).

Mini Project: Implement Mini Project consists of various classifiers and specify which is the best for given data set.

Course Outcomes:

After completion of the course the student will be able to:
1. Analyze Exploratory Data Analysis for given data set.
2. Implement Unsupervised and Supervised Learning algorithms for the given data set.
3. Design Time Series Forecasting methods and MLP.

Textbook(s)

1. Wes Mc Kinney, Python for Data Analysis, O'Reilly,2012.
2. Jake Vanderplas, Python - Data Science Handbook, O'Reilly,2016.

Reference Book

1. Andreas C. Muller, Sarah Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists, O'Reilly,2016.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA II Year I Semester

22MCAP208 JAVA PROGRAMMING LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite **Practical knowledge on OOPs concepts**

Course Description:

In this course, students gain extensive hands-on experience on Java programming. Students learn to create robust console applications using code reusability with multi-threading, applications of exception handling, I/O streams, and GUI implementation.

Course Objectives:

Student will be able to:

1. Implement java programs using basic constructs.
2. Build robust applications using object-oriented features.
3. Read and write data using Java streams.
4. Develop platform-independent GUIs.

List of Programs:

1. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1. Every subsequent value is the sum of the 2 values preceding it. Write a Java program that uses both recursive and non-recursive functions to print the nth value of the Fibonacci sequence?
2. Write a Java program that prints all real solutions to the quadratic equation ax2+bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions?
3. Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer. (use Scanner class to read input).
4. Write a java 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 contain only the method printArea( ) that prints the area of the given shape.
5. Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
6. Write a program to create interface A in this interface we have two method meth1 and meth2. Implements this interface in another class named MyClass.
7. Write a program to create interface named test. In this interface the member function is square. Implement this interface in arithmetic class.Create one new class called ToTestInt

in this class use the object of arithmetic class.

8. Write a java program using Package:

i. Write a program to create a package named mypack and import it in circle class.

ii. Write a program to create a package named pl, and implement this package in ex1 class.

9. Write a Java program that reads on file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes?

10. Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers.

11. Write a program to create a class named shape. In this class we have three sub classes circle, triangle and square each class has two member function named draw () and erase ().Create these using polymorphism concepts.

12. Write a program for example of try and catch block. In this check whether the given array size is negative or not.

13. Write a java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “stop” or “ready” or “go” should appear above the buttons in a selected colour. Initially there is no message shown.

14. Write a java program that works as a calculator. Use a Grid Layout to arrange Buttons for digits and for the + - * % operations. Add a text field to display the result.

15. Write a program that displays the menu bar and when You click the options it has to display a dialog box stating which option has been clicked.

16. Write a program that will display check boxes and option buttons they are numbered from1 to. Use a textbox to display the number those corresponding boxes or button checked.

17. Write a Java program for handling mouse events.

18. Write a Java program for handling key events using Adapter classes.

Course Outcomes:

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

1. Develop programs using basic constructs based on the Java structure..
2. Implement programs for managing I/O operations.
3. Write the program for various functionalities of GUI.

Textbook(s)

1. Java Programming for core and advanced learners, Sagayaraj, Dennis, Karthik and Gajalakshmi, University Press, 2018.
2. Programming with Java” T.V.Suresh Kumar, B.Eswara Reddy, P.Raghavan Pearson Education, 2010.

Reference Books

1. Deitel & Deitel, Java-How to Program, 9th edition, Pearson Education, 2012.

2. Herbert Schildt, Java: The Complete Reference, 10th Edition, McGraw-Hill Education, 2018.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA II Year I Semester

22MCAP701 MINI PROJECT

L	T	P	C
0	0	4	2

Course Description:

The Mini Project is a group-based task in the MCA program where students collaborate to design and develop a small-scale software solution, enhancing teamwork, technical skills, and practical understanding of real-world problems.

Course Outcomes:

After completion of the course the student will be able to

1. Identify problems within the chosen domain of interest.
2. Conduct a literature survey to analyse and understand the problem.
3. Design, develop, and evaluate a solution to the problem using the latest technologies.
4. Implement the project components while considering ethical responsibilities, cost, and time constraints.
5. Develop communication skills, both oral and written, to effectively present the project and prepare the project report.
6. Apply the results for sustainable development and the betterment of society.

MCA II Year II Semester

22MCAP702 COMPREHENSIVE VIVA

L	T	P	C
0	0	4	2

Course Description:

The Comprehensive Viva is an evaluative examination designed to assess a student's overall understanding, analytical abilities, and practical knowledge gained throughout the MCA program. It tests their proficiency in core concepts, problem-solving skills, interdisciplinary integration, and ability to communicate technical ideas effectively, ensuring readiness for professional practice in computer applications.

Course Outcomes:

After completion of the course the student will be able to

1. Demonstrate mastery of core MCA concepts and technologies.
2. Apply problem-solving skills to analyse and solve complex computing problems.
3. Integrate interdisciplinary knowledge for real-world applications.
4. Critically evaluate software systems and architectures.
5. Communicate technical concepts clearly and effectively.
6. Exhibit readiness for professional practice with ethical and emerging tech awareness.

MCA II Year II Semester

22MCAP703 INTERNSHIP/MAJOR PROJECT

L	T	P	C
0	0	4	2

Course Description:

The Major Project is a final-year MCA task where students create a software solution for a real-world problem, applying their skills and knowledge, and present their work to show they are ready for professional IT roles.

Course Outcomes:

After completion of the course the student will be able to

1. Identify problems within the chosen domain of interest.
2. Conduct a literature survey to analyse and understand the problem.
3. Design, develop, and evaluate a solution to the problem using the latest technologies.
4. Implement the project components while considering ethical responsibilities, cost, and time constraints.
5. Develop communication skills, both oral and written, to effectively present the project and prepare the project report.
6. Apply the results for sustainable development and the betterment of society

DISCIPLINE ELECTIVE-II

MCA II Year I Semester

22MCAP408 DEEP LEARNING

L	T	P	C
3	0	0	3

Pre-requisite **Machine Learning**

Course Description:

This course covers the theoretical foundations, algorithms and methodologies of various Deep Neural Network models.

Course Objectives:

This course enables students to

1. Understand basics of Machine Learning.
2. Infer the deep learning architectures.
3. Understand Convolution Neural networks.
4. Understand Recurrent Neural networks.
5. Analyse the applications using deep learning models.

UNIT I MACHINE LEARNING BASICS 9 hours

Learning algorithms, Building machine learning algorithm, Neural Networks Multilayer Perceptron, Back-propagation algorithm and its variants Stochastic gradient decent, Curse of Dimensionality.

UNIT II DEEP LEARNING ARCHITECTURES 9 hours

Machine Learning and Deep Learning, Representation Learning, Width and Depth of Neural Networks, Activation Functions: ReLU, LReLU, EReLU, Unsupervised Training of Neural Networks, Restricted Boltzmann Machines, Auto Encoders, Deep Learning Applications.

UNIT III CONVOLUTIONAL NEURAL NETWORKS 9 hours

Architectural Overview, Motivation, Layers, Filters, Parameter sharing, Regularization, Popular CNN Architectures: ResNet, AlexNet – Applications.

UNIT IV SEQUENCE MODELLING –RECURRENT AND RECURSIVE NEURAL NETWORK 9 hours

Recurrent Neural Networks, Bidirectional RNNs, Encoder-decoder sequence to sequence architectures - BPTT for training RNN, Long Short Term Memory Networks, Recursive Neural Networks.

UNIT V AUTO ENCODERS & DEEP GENERATIVE MODELS 9 hours

Under complete Auto encoder, Regularized Auto encoder, stochastic Encoders and Decoders, Contractive Encoders, Deep Belief networks, Boltzmann Machines, Deep Boltzmann Machine.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand the basics of Machine Learning algorithms.
2. Recognize the characteristics of deep learning models that are useful to solve real-worl problems.
3. Investigate the CNN for real life applications.
4. Apply RNN to solve problems.
5. Write different deep learning algorithms.

Textbook(s)

1. Ian Goodfellow, YoshuaBengio and Aaron Courville, "Deep Learning", MIT Press, 2017.
2. Josh Patterson, Adam Gibson "Deep Learning: A Practitioner's Approach", O'Reilly Media, 2017.

Reference Books

1. Umberto Michelucci "Applied Deep Learning. A Case-based Approach to Understanding Deep Neural Networks" Apress, 2018.
2. Kevin P. Murphy "Machine Learning: A Probabilistic Perspective", The MIT Press, 2012.
3. EthemAlpaydin,"Introduction to Machine Learning", MIT Press, Prentice Hall of India,Third Edition 2014.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA II Year I Semester

22MCAP409 WIRELESS SENSOR NETWORKS

L	T	P	C
3	0	0	3

Pre-requisite **Computer Networks**

Course Description:

This course covers the fundamentals of wireless sensor design and technology, the communication protocols used, compatible operating System and the application requirements of Wireless Sensor Networks

Course Objectives:

This course enables students to

1. Introduce the characteristics and basic concepts in wireless sensor networks.
2. Identify the trends and latest development of the technologies in the area.
3. Illustrate architecture and protocols in wireless sensor networks.
4. Analyse Transport control and middleware technology in WSN
5. Recognize the suitable OS for WSN.

UNIT I INTRODUCTION AND WSN STANDARDS 9 hours

Overview and Applications of Wireless Sensor Networks, WSN Technology, WSN Standards, IEEE 802.15.4, Zigbee. Network architectures for WSN, classification of WSN, protocol stack for WSN.

UNIT II WIRELESS SENSOR TECHNOLOGY 9 hours

Basic Wireless Sensor Technology and Systems: Sensor Node Technology, Sensor Taxonomy, WN Operating Environment, WN Trends, Wireless Transmission Technology and Systems: Introduction, Radio Technology Primer, Available Wireless Technologies.

UNIT III MAC AND ROUTING PROTOCOLS FOR WSN 9 hours

Fundamentals of MAC Protocols, MAC Protocols for WSNs, Sensor-MAC **case Study**:IEEE 802.15.4, LR-WPANs Standard, Routing Protocols for Wireless Sensor Networks, Data Dissemination and Gathering, Routing Challenges and Design Issues in WSNs, Routing Strategies in WSNs, **Case Study**: LEACH, HEED routing protocols.

UNIT IV TRANSPORT CONTROL AND MIDDLEWARE FOR WSN 9 hours

Traditional Transport Control Protocols, Transport Protocol Design Issues, Examples of Existing Transport Control Protocols, Performance of Transport Control Protocols. Middleware for Wireless Sensor Networks: WSN Middleware Principles, Middleware Architecture, Existing Middleware.

UNIT V NETWORK MANAGEMENT AND OPERATING SYSTEM FOR WSN **9 hours**

Introduction, Network Management Requirements, Traditional Network Management Models, Network Management Design Issues. Operating Systems for Wireless Sensor Networks: Introduction, Operating System Design Issues, Case Study: WSN Operating Systems.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand sensor networks for various applications and WSN Standard.
2. Classify the suitable medium access protocols, localization techniques.
3. Design suitable routing protocols for WSN.
4. Illustrate various Transport Control Protocols for WSN.
5. Analyse the suitable Operating Systems for WSN.

Textbook(s)

1. Kazem Sohraby, Daniel Minoli, Taieb Znati, “wireless sensor networks: technology, protocols and applications:, wiley , second edition (indian) , 2014.

2. Dr. Jayapandian N Dr. Piyush Charan, Mrs. Sonali Yugesh Pakhmode, Dr. Latika Desai. *IOT & Wireless Sensor Networks*. RK Publications, 2023.

Reference Books:

1. Ian F. Akyildiz, Mehmet Can Vuran "Wireless Sensor Networks", Wiley 2010

2. Feng Zhao & Leonidas J. Guibas, “Wireless Sensor Networks- An Information Processing Approach", Elsevier, 2007

3. Jun Zheng, Abbas Jamalipour, “Wireless Sensor Networks: A Networking Perspective”, Wiley, 2009.

4. Ibrahiem M. M. El Emary, S. Ramakrishnan, “Wireless Sensor Networks: From Theory to Applications”, CRC Press Taylor & Francis Group, 2013

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA II Year I Semester

22MCAP410 SOFTWARE QUALITY ASSURANCE AND TESTING

L	T	P	C
3	0	0	3

Pre-requisite **Software Engineering**

Course Description:

This course covers software quality assurance and testing practices which are applied in real time. These learning practices help the students to understand about how software quality assurance takes place and how to check software testing with various standard techniques. With concepts and knowledge gained from this course, the student can easily become part of industrial software production.

Course Objectives:

This course enables students to

1. Understand the basics of testing, test planning &design and test team organization.
2. Differentiate the various types of tests in the life cycle of the software product.
3. Build design concepts for system testing and execution.
4. Analyse the software quality assurance, metrics, defect prevention techniques.
5. Identify the techniques for quality assurance and applying for applications.

UNIT I SOFTWARE TESTING: CONCEPTS AND TECHNIQUES 9 hours

Introduction to testing, Defects and Classification, Testing Strategies, Verification and Validation, Terminologies, Objectives of Testing, Testing Activities, Test Case Selection, White-Box and Black Box testing, Test Planning and design, Test Tools and Automation. Test Team Organization and Management: Test Groups, Software Quality Assurance Group, System Test Team Hierarchy, Team Building.

UNIT II LEVELS OF TESTING 9 hours

System Testing - System Integration Techniques-Incremental, Top Down and Bottom-Up, Sandwich and Big Bang, Software and Hardware Integration, Built- in Testing. Functional testing - Testing a Function in Context. Boundary Value Analysis, Decision Tables. Acceptance testing - Selection of Acceptance Criteria, Acceptance Test Plan, Test Execution.

UNIT III FUNCTIONAL AND NON-FUNCTIONAL TESTING 9 hours

Functional Testing: Unit Testing – Integration Testing – System Testing – Regression Testing – Database Testing.

Non-functional Testing: Performance Testing – Load Testing – Stress Testing – Volume Testing – Scalability Testing.

UNIT IV SOFTWARE QUALITY STANDARDS 9 hours

Software quality – People's Quality Expectations, Frameworks and ISO 9126, McCall's Quality Factors and Criteria – Relationship. Quality Metrics. Quality Characteristics, ISO 9000:2000 Software Quality Standards. Maturity models- Test Process Improvement, Testing Maturity Model.

UNIT V SOFTWARE QUALITY ASSURANCE 9 hours

Quality Assurance - Root Cause Analysis, Modeling, technologies, standards and methodologies for defect prevention. Fault Tolerance and Failure Containment - Safety Assurance and Damage Control, Hazard analysis using fault-trees and event-trees. Comparing Quality Assurance Techniques and Activities. QA Monitoring and Measurement, Risk Identification for Quantifiable Quality Improvement. Case Study: FSM-Based Testing of Web-Based Applications.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Perform functional and non-functional tests in the life cycle of the software product.
2. Implement system testing and test execution process.
3. Identify defect prevention techniques and software quality assurance metrics.
4. Analyse quality and reliability metrics to ensure the performance of the software.
5. Apply techniques of quality assurance for typical applications.

Textbook(s)

1. Software Testing and Quality Assurance-Theory and Practice, Kshirasagar Nak Priyadarshi Tripathy, John Wiley & Sons Inc,2010.

2. Software Testing & Quality Assurance, Dr.Vijyakumar S.Bidore, Sheetal S.Kaul, 2020 edition.

Reference Books

1. Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement, Jeff Tian, John Wiley & Sons, Inc., Hoboken, New Jersey. 2005.

2. Software Quality Assurance - From Theory to Implementation, Daniel Galin, Pearson Education Ltd UK, 2004.

3. Software Quality Assurance, Milind Limaye, TMH, New Delhi, 2011.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

MCA II Year I Semester

22MCAP411 VIRTUAL REALITY

L	T	P	C
3	0	0	3

Pre-requisite **Computer Graphics and Multimedia Systems**

Course Description:

This course introduces the fundamentals of virtual reality concepts. It focuses on geometric modelling and transformation models. It describes the generic VR system and animation models. It explains simulation methods and human factors. It describes VR hardware, software, and applications

Course Objectives:

This course enables students to

1. Understand the basics of virtual reality through 3D graphics.
2. Interpret geometric modeling and transformations.
3. Explain the generic VR system and animations in VR environments.
4. Understand simulation models and human factors.
5. Analyze VR hardware, software, and applications.

UNIT I FUNDMENTALS OF VIRTUAL REALITY 9 hours

Virtual Reality and Virtual Environments: Introduction – Computer graphics – Real-time computer graphics – Virtually environments - Flight simulation.
The Historical Development of VR: Introduction – Scientific landmarks.
3D Computer Graphics: Introduction – The virtual world space – Positioning the virtual observer – The perspective projection – Human vision – Stereo perspective projection – 3D clipping – Colour theory – Simple 3D modeling – Illumination models – Reflection models – Shading algorithms – Radiosity – Hiddeny-surface removal – Realism – Stereographic image.
Tool: Develop VR models for building a basic science and Buidling a 360^{o} Image Gallery using A-frame.

UNIT II GEOMETRIC MODELING AND TRANSFORMATION 9 hours

Geometric Modelling: Introduction – Conversion from 2D to 3D – 3D space curves – 3D boundary representation.
Geometrical Transformations: Introduction – Frames of reference – Modelling transformations – Instances – Picking – Flying – Scaling the VE – Collision detection.

UNIT III VIRTUAL REALITY SYSTEM AND ANIMATIONS 9 hours

A Generic VR System: Introduction – The virtual environment – The computer environment – VR technology – Modes of interaction – VR systems.

Animating the Virtual Environment: Introduction – The dynamics of numbers – The animation of objects – Shape and object in between – Free-from deformation – Particle systems.

Tool: Develop an immersive VR game using UNITY.

UNIT IV SIMULATION MODELS 9 hours

Physical Simulation: Introduction – Objects falling in a gravitational field – Rotating wheels – Elastic collisions – Projectiles – Simple pendulums – Springs – Flight dynamics of an aircraft.

Human Factors: Introduction – The eye – The ear – The somatic senses – Equilibrium.

UNIT V VR HARDWARE, SOFTWARE AND APPLICATIONS 9 hours

Virtual Reality Hardware: Introduction – Sensor hardware – Head-coupled displays – Acoustic hardware – Integrated VR systems.

Virtual Reality Software: Introduction – Modelling virtual worlds – Physical simulation – VR toolkits. **Virtual Reality Applications**: Introduction – Engineering – Entertainment – Science – Training.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Explain basic concepts of virtual reality and 3D graphics.
2. Apply geometric modelling and transformation.
3. Design generic VR system and animation models.
4. Analyze various simulation models.
5. Elucidate VR hardware, software, and applications.

Textbook(s)

1. William R.Sherman, Alan Craig, "Understanding Virtual Reality, interface, Application and Design", Elsevier, Morgan Kaufmann, 2018.

2. John Vince, "Virtual Reality Systems", Pearson Education Ltd., ACM Press. 2017.

Reference Books

1. Bill Fleming,"3D Modeling and surfacing", Elsevier, Morgan Kauff man, 2021.

2. David M. Ewalt, Defying Reality: The Inside Story of the Virtual Reality Revolution
Blue Rider Press, 2018

3. David H.Eberly, "3D Game Engine Design Practical Approach to Real-Time Computer Graphics", Elsevier, 2017.

4 Matjaž Mihelj, Domen Novak, Samo Beguš, "Virtual Reality Technology and Applications", 2nd edition, Springer,2014.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP412 VIDEO ANALYTICS

L	T	P	C
3	0	0	3

Pre-requisite **Image Processing**

Course Description:

This course covers video analytics details and practices which are applied in real-time. These learning practices help the students to understand, what skills they required and how to apply themin video analytics. With these concepts and knowledge gained from this course, one can easily become part of IT and other relevant domains.

Course Objectives:

This course enables students to

1. Understand the concepts of Video Analytics.
2. Analyze the various foreground extraction techniques of video analytics.
3. Explore the various classifiers for video analytic system.
4. Acquire the knowledge in various security mechanisms using video analytics.
5. Analyze the smart monitoring applications.

UNIT I VIDEO ANALYTIC COMPONENTS 9 hours

Need for Video Analytics-Overview of video Analytics- Foreground extraction- Feature extraction-classifier - Pre-processing- edge detection- smoothening- Feature space-PCA-FLD-SIFT features.

UNIT II FOREGROUND EXTRACTION 9 hours

Background estimation- Averaging- Gaussian Mixture Model- Optical Flow based- Image Segmentation- Region growing- Region Splitting-Morphological operations- erosion-Dilation-Tracking in a multiple camera environment.

UNIT III CLASSIFIERS 9 hours

Neural networks: back propagation - Deep learning networks- Fuzzy Classifier- Bayesian classifier-HMM based classifier.

UNIT IV VIDEO ANALYTICS FOR SECURITY 9 hours

Abandoned object detection- Human behavioural analysis - Human action recognition- Perimeter security- Crowd analysis and prediction of crowd congestion.

UNIT V VIDEO ANALYTICS FOR BUSINESS INTELLIGENCE & TRAFFIC MONITORING AND ASSISTANCE 9 hours

Customer behaviour analysis - people counting- Traffic rule violation detection- traffic congestion identification for route planning- driver assistance- lane change warning.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Implement video analytic algorithms for security applications.
2. Apply video analytic algorithms for business intelligence.
3. Analyze various classifier algorithm.
4. Design custom made video analytics system for the given target application.
5. Explore the video analytics real time applications.

Textbook(s)

1. Zhihao Chen, Ye Yang, Jingyu Xue, Liping Ye, Feng Guo, "The Next Generation of Video Surveillance and Video Analytics: The Unified Intelligent Video Analytics Suite", CreateSpace Independent Publishing Platform, 2014.
2. Caifeng Shan, Fatih Porikli, Tao Xiang, Shaogang Gong, "Video Analytics for Business Intelligence", Springer, 2012.

Reference Books

1. Graeme A. Jones, Nikos Paragios, Carlo S. Regazzoni, "Video-Based Surveillance Systems: Computer Vision and Distributed Processing", Kluwer academic publisher, 2001.
2. Nilanjan Dey, Amira Ashour and Suvojit Acharjee, "Applied Video Processing in Surveillance and Monitoring Systems", IGI global, 2016.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP413 SOFTWARE DEVELOPMENT USING BLOCKCHAIN

L	T	P	C
3	0	0	3

Pre-requisite **Blockchain Technology**

Course Description:

This course provides a broad overview of the essential concepts of block-chain technology by initially exploring the Bitcoin protocol followed by the Ethereum protocol to lay the foundation necessary for developing applications and programming.

Course Objectives:

This course enables students to

1. Understand the history, types, and applications of Blockchain
2. Acquire knowledge about Blockchain Architecture and Consensus mechanism
3. Identify and design projects using Web3j.
4. Apply the cryptography and consensus algorithms.
5. Understand the concept of Ethereum Virtual Machine.

UNIT I INTRODUCTION TO BLOCKCHAIN 9 hours

Distributed DBMS – Limitations of Distributed DBMS, Introduction to Block chain – History, Definition, Distributed Ledger, Blockchain Categories – Public, Private, Consortium, Blockchain Network and Nodes, Peer-to-Peer Network, Mining Mechanism, Generic elements of Blockchain, Features of Blockchain, and Types of Blockchain.

UNIT II BLOCKCHAIN ARCHITECTURE 9 hours

Operation of Bitcoin Blockchain, Blockchain Architecture – Block, Hash, Distributer P2P, Structure of Blockchain- Consensus mechanism: Proof of Work (PoW), Proof of Stake (PoS), Byzantine Fault Tolerance (BFT), Proof of Authority (PoA) and Proof of Elapsed Time (PoET).

UNIT III BLOCKCHAIN-BASED FUTURE SYSTEM 9 hours

Project presentation- Futures smart contract: Blockchain oracles- Web3j: Setting up the Web3J-Installing web3j- Wallet creation, Java client: The wrapper generator- Initializing web3j- Setting up Ethereum accounts- Deploying the contract.

UNIT IV BLOCKCHAINS IN BUSINESS AND CREATING ICO 9 hours

Public versus private and permissioned versus permission less blockchains- Privacy and anonymity in Ethereum- Why are privacy and anonymity important? - The Ethereum Enterprise Alliance-

Blockchain- as-a-Service- Initial Coin Offering (ICO): Project setup for ICO implementation- Token contracts- Token sale contracts-Contract security and testing the code.

UNIT V DISTRIBUTED STORAGE IPFS AND SWARM 9 hours

Ethereum Virtual Machine- Swarm and IPFS: Installing IPFS, Hosting our frontend: Serving your frontend using IFPS, Serving your frontend using Swarm, IPFS file uploader project: Project setup the web page.

Course Outcomes:

Upon successful completion of the course, students will be able to
1. Understand the history, types and applications of Blockchain.
2. Utilize the cryptography and Consensus algorithms.
3. Set up Web3j, including installation, wallet creation, and Java client configuration.
4. Understand the Initial Coin Offering (ICO) on Ethereum.
5. Identify the blockchain based application with Swarm and IPFS.

Textbook(s)

1. Imran Bashir, “Mastering Blockchain: Distributed Ledger Technology, decentralization, and smart contracts explained”, 2nd Edition, Packt Publishing Ltd, March 2018.
2. Bellaj Badr, Richard Horrocks, Xun (Brian) Wu, “Blockchain by Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger”, Packt Publishing Limited, 2018.

Reference Books

1. Andreas M. Antonopoulos, “Mastering Bitcoin: Unlocking Digital Cryptocurrencies”, O’Reilly Media Inc, 2015.
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, “Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction”, Princeton University Press, 2016.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP414 ADVANCED WEB DEVELOPMENT

L	T	P	C
3	0	0	3

Pre-requisite **XML and Web Services**

Course Description:

This course covers the fundamentals of typescript language with angular, NODE.Js, EXPRESS.Js and MONGODB concepts. With concepts and knowledge gained from this course, the student can easily become part of industrial web development.

Course Objectives:

This course enables students to

1. Understand the basics to develop Typescript Application.
2. Create Single Page Application (SPA).
3. Communicate with a server over the HTTP protocol.
4. Identify the tools need to start building applications with Node.js.
5. Implement the Full Stack Development using MEAN Stack.

UNIT I FUNDAMENTALS & TYPESCRIPT LANGUAGE 9 hours

Server-Side Web Applications. Client-Side Web Applications. Single Page Application. About TypeScript. Creating TypeScript Projects. TypeScript Data Types. Variables. Expression and Operators. Functions. OOP in Typescript. Interfaces. Generics. Modules. Enums. Decorators. Enums. Iterators. Generators.

UNIT II ANGULAR 9 hours

About Angular. Angular CLI. Creating an Angular Project. Components. Components Interaction. Dynamic Components. Angular Elements. Angular Forms. Template Driven Forms. Property, Style, Class and Event Binding. Two way Bindings. Reactive Forms. Form Group. Form Controls. About Angular Router. Router Configuration. Router State. Navigation Pages. Router Link. Query Parameters. URL matching. Matching Strategies. Services. Dependency Injection. Http Client. Read Data from the Server. CRUD Operations. HTTP Header Operations. Intercepting requests and responses.

UNIT III CONFIGURING NODE.Js ENVIRONMENT 9 hours

About Node.js. Node Package Manager NPM. Modules. Asynchronous Programming. Call Stack and Event Loop. Call back functions. Call back errors. Abstracting call backs. Chaining call backs. File System. Synchronous vs. asynchronous I/O. Path and directory operations. File Handle. File Synchronous API. File Asynchronous API. File Call back API. Timers. Scheduling Timers. Timers Promises API. Node.js Events. Event Emitter. Event Target and Event API. Buffers. Buffers and Typed Arrays. Buffers and iteration. Using buffers for binary data. Flowing vs. non-flowing streams. JSON.

UNIT IV REACT.Js 9 hours

ES6 Overview, Introduction to React, Templating using JSX, Components, Working with state with props, Rendering lists, Event handling in React, Working with forms, Routing with react router, Performing CRUD operations in ReactJS.

UNIT V MONGODB 9 hours

Introduction to MongoDB. Documents. Collections. Subcollections. Database. Data Types. Dates. Arrays. Embedded Documents. CRUD Operations. Batch Insert. Insert Validation. Querying The Documents. Cursors. Indexing. Unique Indexes. Sparse Indexes. Special Index and Collection Types. Full-Text Indexes. Geospatial Indexing. Aggregation framework.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand the type script programming concepts.
2. Implement a front-end web application using Angular.
3. Utilize the Node JS API in applications.
4. Understand the React.JS Concepts and perform CRUD operations.
5. Utilize the MongoDB to Store complex relational data.

Textbooks(s)

1. Adam Freeman, Essential TypeScript, Apress, 2019.

Reference Books

1. Mark Clow, Angular Projects, Apress, 2018.

2. Alex R. Young, Marc Harter,Node.js in Practice, Manning Publication, 2014.

3. Pro Express.js, Azat Mardan, Apress, 2015.

4. MongoDB in Action, Kyle Banker, Peter Bakkum, Shaun Verch, Douglas Garrett, Tim Hawkins, Manning Publication, Second edition, 2016.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

DISCIPLINE ELECTIVE-III

MCA II Year I Semester

22MCAP415 REINFORCEMENT LEARNING

L	T	P	C
3	0	0	3

Pre-requisite **Machine Learning – Algorithms and Applications**

Course Description:

Reinforcement learning (RL) is an area of machine learning, where an agent or a system of agents learns to archive a goal by interacting with their environment. RL is often seen as the third area of machine learning, in addition to supervised and unsupervised areas, in which learning of an agent occurs as a result of its own actions and interaction with the environment.

Course Objectives:

This course enables students to

1. Understand RL task formulation.
2. Recognize various model-free prediction techniques.
3. Identify Function approximation solutions.
4. Devise Policy gradient from basic (REINFORCE) towards advanced topics.
5. Understand Model-based reinforcement learning.

UNIT I INTRODUCTION TO RL & MARKOV DECISION PROCESS 9 hours

The RL-Problem-Markov Process, Markov Reward Process, Markov Decision Process and Bellman Equations, Partially Observable MDPs Policy Evaluation, Value Iteration, Policy Iteration, DP Extensions and Convergence using Contraction Mapping.

UNIT II MODEL-FREE PREDICTION &MODEL-FREE CONTROL 9 hours

Monte-Carlo (MC) Learning, Temporal-Difference (TD) Learning, TD-Lambda and Eligibility Traces On-Policy MC Control, On-Policy TD Learning and Off-Policy Learning.

UNIT III VALUE FUNCTION APPROXIMATION& POLICY GRADIENT METHODS 9 hours

Incremental Methods and Batch Methods, Deep Q-Learning, Deep Q-Networks and Experience Replay Finite-Difference, Monte-Carlo and Actor-Critic Methods.

UNIT IV INTEGRATING PLANNING WITH LEARNING& HIERARCHICAL RL 9 hours

Model-based RL, Integrated Architecture and Simulation-based Search, Semi-Markov Decision Process, Learning with Options, Abstract Machines and MAXQ Decomposition.

UNIT V DEEP RL & MULTI-AGENT RL 9 hours

PPO, DDPG, Double Q-Learning, Advanced Policy Gradients, Cooperative vs. Competitive Settings, Mixed Setting, Games, MARL Algorithms.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand the core principles behind the RL, including policies, value functions, deriving Bellman equations.
2. Comprehend tabular methods to solve classical control problems.
3. Analyse the approximate solutions (deep Q network-based algorithms).
4. Indetify the policy gradient methods.
5. Recognize current advanced techniques and applications in RL.

Textbook(s)

1. Richard S. Sutton and Andrew G. Barto; Reinforcement Learning: An Introduction; 2nd Edition, MIT Press, 2020.

2. Csaba Szepesvári; Algorithms of Reinforcement Learning; Synthesis Lectures on Artificial Intelligence and Machine Learning, vol. 4, no. 1, 2010.

Reference Books

1. Dimitri P. Bertsekas; Reinforcement Learning and Optimal Control; 1st Edition, Athena Scientific, 2019.

2. Dimitri P. Bertsimas; Dynamic Programming and Optimal Control (Vol. I and Vol. II); 4th Edition, Athena Scientific, 2017

3. Leslie Pack Kaelbling, Michael L. Littman and Andrew W. Moore; Reinforcement Learning: A Survey; Journal of Artificial Intelligence Research, vol.4, pp. 237-285, 1996

4. .Andrew G. Barto and Sridhar Mahadevan; Recent Advances in Hierarchical Reinforcement Learning; Discrete Event Dynamic Systems, vol. 13, pp. 341–379, 2003.

5. Thomas G. Dietterich; Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition; Journal of Artificial Intelligence Research, vol. 13, pp. 227-303, 2000.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP416 PRIVACY AND SECURITY IN IOT

L	T	P	C
3	0	0	3

Pre-requisite **IoT Technology and Applications**

Course Description:

This course gives an overview of IoT system in security aspects. The course mainly focusses on current security risks in IoT domain and countermeasure available for the known issues.

Course Objectives:

This course enables students to

1. Define the scope of Software Project Management.
2. Select appropriate strategies for minimizing potential costs.
3. Prepare the project activity plan.
4. Monitor and control the progress of projects.
5. Identify factors that influence people's behaviour in a project environment.

UNIT I INTRODUCTION TO IOT SECURITY RISKS 9 hours

IoT Reference Model- Introduction -Functional View, IoT Security Challenges-Hardware Security Risks - Hardcoded/Default Passwords -Resource Constrained Computations -Legacy Assets Connections - Devices Physical Security, Software Security Risks -Software Vulnerabilities -Data Interception - Identification of Endpoints -Tamper Detection, Lack of Industrial Standards.

UNIT II IOT- SECURITY ATTACKS &VULNERABILITY ISSUES 9 hours

IoT Security Requirements -Data Confidentiality -Data Encryption -Data Authentication -Secured Access Control –IoT-Vulnerabilities – Secret-Key, Authentication/Authorization for Smart Devices Constrained System Resources -Device Heterogeneity -Fixed Firmware - IoT Attacks -Side-channel attacks -Reconnaissance -Spoofing -Sniffing -Neighbour -Discovery -Rogue Devices-Man-in-Middle.

UNIT III SECURED PROTOCOLS FOR IOT 9 hours

Infrastructure-IPv6 -LowPAN , Identification-Electronic Product Code -uCode, Transport-Bluetooth - LPWAN, Data -MQTT -CoAP, Multi-layer Frameworks-Alljoyn, - IoTivity.

UNIT IV SECURING INTERNET OF THINGS ENVIRONMENT 9 hours

IoT Hardware -Test Device Range-Latency and Capacity -Manufacturability Test -Secure from Physical Attacks, IoT Software -Trusted IoT Application Platforms, -Secure Firmware Updating -Network Enforced Policy -Secure Analytics, Visibility and Control.

UNIT V IOT ATTACKS - CASE STUDIES 9 hours

MIRAI Botnet Attack -Iran's Nuclear Facility Stuxnet Attack -Tesla Crypto Jacking Attack -The TRENDnet Webcam Attack -The Jeep SUV Attack -The Owlet Wi-Fi Baby Heart Monitor Vulnerabilities -St.Jude, Hackable Cardiac Devices.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand IoT general models and security challenges.
2. Recognize IoT security and vulnerability threats.
3. Analyze the different IoT protocols and their security measures.
4. Interpret how to secure an IoT environment.
5. Differentiate different types of attacks.

Textbook(s)

1. B. Rusell and D.Van Duren, “Practical Internet of Things Security,” Packt Publishing, 2016.
2. Mahmood, Zaigham, ed. Security, Privacy and Trust in the IoT Environment. Springer, 2019.

Reference Books

1. Gilchrist, Alasdair. *IoT security issues*. De| G Press, 2017.
2. Security and Privacy in the Internet of Things: Challenges and Solutions. (2020). Germany: IOS Press.
3. Dave, Mayank. Security and Privacy Issues in Sensor Networks and IoT. United States, IGI Global, 2019.
4. Xu, Li Da, and Li, Shancang. Securing the Internet of Things. United States, Elsevier Science, 2017.
5. Liyanage, Madhusanka, et al. IoT Security: Advances in Authentication. United Kingdom, Wiley, 2019.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP417 SOFTWARE PROJECT MANAGEMENT

L	T	P	C
3	0	0	3

Pre-requisite **Software Engineering**

Course Description:

This course describes the key aspects of managing the process of developing a software. It introduces the strategies required for managing projects from their genesis to completion. The course brings knowledge about Software Project Planning, Cost Estimation, scheduling and management tools.

Course Objectives:

This course enables students to

1. Define the scope of Software Project Management.
2. Evaluate the project risk and select appropriate strategies for minimizing potential costs.
3. Understand the project activity plan.
4. Monitor and control the progress of projects.
5. Identify factors that influence people's behaviour in a project environment.

UNIT I INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT 9 hours

Introduction to project and product, Software projects versus other types of projects, Activities Covered by Software Project Management, software projects categorization, the project as a system, what is Management? Problems with software projects, Management Control, Stakeholders, Information and control in organization, overview of project planning.

UNIT II PROJECT EVALUATION AND ESTIMATION 9 hours

Evaluation of plan, Cost Estimation, Cost Benefit Evaluation Techniques, Risk Evaluation, Software effort estimation: where are estimates done, Problems with over and under estimates, the basis for software estimating, Software effort estimation techniques.

UNIT III ACTIVITY PLANNING AND RISK MANAGEMENT 9 hours

Objectives of Activity Planning, when to Plan, Project Schedules, Sequencing and Scheduling Activities, Network Planning Models, PERT & CPM, Forward Pass – Backward Pass, identifying critical path, Activity Float, Shortening Project Duration, Activity on Arrow Networks. Risk Management, Nature of Risk, Categories of Risk, A framework for dealing with Risk, Risk Identification, Risk analysis and prioritization, risk planning and risk monitoring.

UNIT IV MONITORING AND CONTROL 9 hours

Creating the Framework, Collecting the Data, Review, Project Termination Review, Visualizing Progress, Cost Monitoring, Earned Value Analysis, Prioritizing Monitoring, Getting Project Back to Target, Change Control and Version Control.

UNIT V MANAGING PEOPLE AND ORGANIZING TEAMS 9 hours

Task Management, Understanding Behaviour, Organizational Behaviour: A Background, Selecting the Right Person for the Job, Instruction in the Best Methods, Motivation, Becoming a Team, Decision Making, Leadership.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Explain the practices and methods for successful software project management.
2. Identify techniques for requirements, policies and decision making for effective resource management.
3. Apply the evaluation techniques for cost estimation, planning, schedule, and risk.
4. Devise a framework for software project management plan for activities, risk, monitoring, and control.
5. Build a framework to manage people.

Textbook(s):

1. Bob Hughes, Mike Cotterell, Rajib Mall, “Software Project Management”, Fifth Edition, Tata McGraw Hill, 2017.

2. Software Engineering & Project Management, A.A.Puntambekar, Satish S.Banait, Dr.Dinesh B.Hanchate, 2020 edition.

Reference Books:

1. JackMarchewka,” Information Technology-Project Management”, Wiley Student Version, 4th Edition, 2013.

2. James P Lewis,”Project Planning, Scheduling & Control”, McGraw Hill, 5th Edition, 2011.

3. PankajJalote,” Software Project Management in Practise”, Pearson Education, 2002

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP418 GENERATIVE AI

L	T	P	C
3	0	0	3

Pre-requisite Artificial Intelligence, Machine Learning

Course Description:
Generative AI is a type of Artificial Intelligence that can create a wide variety of data, such as **images, videos, audio, text, and 3D models**. It does this by learning patterns from existing data, then using this knowledge to generate new and unique outputs. GenAI is capable of producing **highly realistic and complex content** that mimics human creativity, making it a valuable tool for many industries such as gaming, entertainment, and product design.

Course Objectives:
This course enables students to
1.Understand the Basics of Generative AI
2. Identify the importance of Generative adversarial Networks.
3. Understand the Style transfer and Deep fakes in GAN.
4. Learn the concepts of Text generation using NLP.
5. Learn the Emerging Applications of GAI.

UNIT I FUNDAMENTAL CONCEPTS OF AI 9 hours

Discriminative and Generative Models-Bayes theorem-Deep Neural Networks-CNN-RNN-Gradient Decent to ADAM-LSTM-Boltzmann Machines.

UNIT II GENERATIVE ADVERSARIAL NETWORKS (GAN) 9 hours

The discriminator Model-generator model-Deep Convolutional GAN-Conditional GAN-Wasserstein GAN-Progressive GAN-Challenges.

UNIT III STYLE TRANSFER AND DEEPFAKES WITH GAN 9 hours

Paired Style transfer-The U-Net generator -Patch-GAN generator-Loss-Unpaired Style transfer -Adversarial loss-cycle loss-Identity loss-Overall loss-Deep fakes-facial Action coding System-Repalcement of Autoencoders-Ethical issues and technical challenges.

UNIT IV METHODS FOR TEXT GENERATION WITH NLP 9 hours

Representing the Text-Language Modeling-Decoding strategies-Text Generation using LSTM-Using Transformers to Generate Text-Transformers-Overall architecture -Multi-head Self- attention -Positional Encoding.

UNIT V EMERGING APPLICATIONS IN GENERATIVE AI 9 hours

Music generation using LSTMs-MuseGAN-Video games with Generative AI: GAIL-finding new drugs with generative models-few shot learning for creative videos from images-generating recipes with Deep learning.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand the basics of GAI.
2. Utilize the Generative adversarial Networks.
3. Explain Deep fakes and style transfer using GAN.
4. Apply text translation in NLP applications
5. Analyse the GAI techniques in emerging Applications.

Text Book(s)

1. Generative AI with Python and Tensor Flow 2: Create images, text, and music with VAEs, GANs, LSTMs, Transformer models. Joseph Babcock, Raghav Bali 2021.Expert insight.

2. Generative Deep Learning, 2nd Edition by David Foster Released May 2023Publisher(s): O'Reilly Media, Inc.ISBN: 978109813418.

Reference Books

1. Exploring Deep fakes Get hands-on with generative AI for face replacement
Bryan Lyon, Matt Tora 2023.

2. Generative AI Art Beginner’s Guide to 10x Your Output with Smart Text Prompts
Oliver Theo bald 2023.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

MCA II Year I Semester

22MCAP419 COMPUTER VISION

L	T	P	C
3	0	0	3

Pre-requisite **Image Processing**

Course Description:

This course covers image processing techniques for computer vision in real-time. These learning practices help the students to understand how computer vision techniques work with various standard techniques. With concepts and knowledge gained from 3D reconstruction and recognition can easily become part of software and other relevant industries.

Course Objectives:

This course enables students to

1. Understand the fundamental concepts related to Image formation and processing.
2. Learn feature detection, matching and segmentation.
3. Recognize feature-based alignment and motion estimation.
4. Develop skills in 3D reconstruction.
5. Analyse image-based rendering and recognition.

UNIT I INTRODUCTION TO IMAGE FORMATION AND PROCESSING 9 hours

Computer Vision - Geometric primitives and transformations - Photometric image formation - The digital camera - Point operators - Linear filtering - More neighbourhood operators - Fourier transforms - Pyramids and wavelets - Geometric transformations - Global optimization.

UNIT II FEATURE DETECTION, MATCHING AND SEGMENTATION 9 hours

Points and patches - Edges - Lines - Segmentation - Active contours - Split and merge - Mean shift and mode finding - Normalized cuts - Graph cuts and energy-based methods.

UNIT III FEATURE-BASED ALIGNMENT & MOTION ESTIMATION 9 hours

2D and 3D feature-based alignment - Pose estimation - Geometric intrinsic calibration - Triangulation - Two-frame structure from motion - Factorization - Bundle adjustment - Constrained structure and motion - Translational alignment - Parametric motion - Spline-based motion - Optical flow - Layered motion.

UNIT IV 3D RECONSTRUCTION 9 hours

Shape from X - Active range - finding Surface representations - Point-based representations - Volumetric representations - Model-based reconstruction - Recovering texture maps and albedos.

UNIT V IMAGE-BASED RENDERING AND APPLICATIONS 9 hours

View interpolation Layered depth images - Light fields and Lumigraphs - Environment mattes - Video-based rendering-Object detection - Face recognition - Instance recognition - Category recognition - Context and scene understanding- Recognition databases and test sets.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Understand basic knowledge, theories and methods in image processing and computer vision.
2. Explore basic and advanced image processing techniques in OpenCV.
3. Apply 2D a feature-based based image alignment, segmentation, and motion estimations.
4. Investigate the real time 3D image reconstruction techniques.
5. Identify the image processing and computer vision applications.

Textbook(s)

1. Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer- Texts in Computer Science, Second Edition, 2022.
2. Computer Vision: A Modern Approach, D. A. Forsyth, J. Ponce, Pearson Education, Second Edition, 2015.

Reference Books

1. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004.
2. Christopher M. Bishop; Pattern Recognition and Machine Learning, Springer, 2006.
3. E. R. Davies, Computer and Machine Vision, Fourth Edition, Academic Press, 2012.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP420 CYBER SECURITY USING BLOCKCHAIN

L	T	P	C
3	0	0	3

Pre-requisite **Blockchain Technology**

Course Description:

This course will enable the students to acquire knowledge about security fundamentals and cryptographic algorithms, explore the basics of cryptocurrencies and use Ethereum programming.

Course Objectives:

This course enables students to

1. Understand the fundamentals of cyber security.
2. Recognize the fundamentals of Cryptography and symmetric ciphers.
3. Apply asymmetric ciphers and data integrity algorithms.
4. Explore the basics of cryptocurrencies and cryptography Techniques.
5. Understand the fundamentals of Ethereum programming.

UNIT I INTRODUCTION TO CYBER SECURITY 9 hours

Introduction to Cyber Security, Need for security, Concept of Cyber Space, Cyber Crimes and Cyber-attack. Fundamental security principles – threats, attacks and vulnerability. Key Security triad – Confidentiality, Integrity and Availability. Key components of cyber security network architecture. Introduction to basic Security Management and Policies - Authentication, Authorization, Access control, Identification and Accounting.

UNIT II SYMMETRIC CIPHERS 9 hours

Cryptography – Private key Cryptography - Classical Encryption Techniques - Substitution Techniques - Transposition Techniques - Rotor Machines - Steganography - Data Encryption Standard - Advanced Encryption Standard - Multiple Encryption and Triple DES.

UNIT III ASSYMMETRIC CIPHERS AND DATA INTEGRITY ALGORITHMS 9 hours

Public-Key Cryptography - RSA algorithm - Diffie-Hellman Key Exchange - Elgamal Cryptographic System - Elliptic Curve Arithmetic - Elliptic Curve Cryptography. MD5 message digest algorithm - Secure hash algorithm (SHA) Digital Signatures: Digital Signatures - authentication protocols - digital signature standards (DSS) - proof of digital signature algorithm.

UNIT IV CRYPTOCURRENCIES 9 hours

History, A basic crypto currency, Creation of coins, Payments and double spending, Bitcoin – Digital Signatures as Identities – eWallets – Personal Crypto security - Bitcoin Mining – Mining Hardware – Energy Consumption – Mining Pools – Mining Incentives and Strategies.

UNIT V ETHEREUM 9 hours

The Ethereum Network – Components of Ethereum Ecosystem – Ethereum Programming Languages: Runtime Byte Code, Blocks and Blockchain, Fee Schedule – Supporting Protocols – Solidity Language.

Course Outcomes:

Upon completion of this course, the students will be able to

1. Understand the network security fundamentals.
2. Utilize various symmetric ciphers.
3. Apply asymmetric ciphers and data integrity algorithms.
4. Explore the basics of cryptocurrencies.
5. Comprehend the Ethereum programming.

Textbook(s)

1. William Stallings, "Cryptography and Network security Principles and Practices", Pearson/PHI, 2017.

2. Arvind Narayanan, Joseph Bonneau, Edward Feder, Andrew Miller, and Steven Gold Feder, "Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction", Princeton University Press, July 2016.

Reference Books

1. William Stallings, Network Security Essentials (Applications and Standards), Pearson Education, India,2017
2. Imran Bashir, "Mastering Blockchain: Distributed Ledger Technology, Decentralization and Smart Contracts Explained", Second Edition, Packt Publishing, 2018.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

MCA II Year I Semester

22MCAP421 .NET FRAMEWORK AND C#

L	T	P	C
3	0	0	3

Pre-requisite XML and Web Services

Course Description:

The course is designed to provide the complete knowledge of C# language. Students will be able to develop logic which will help them to create programs, applications in C#. Microsoft has developed C# with features of popular languages to develop different types of .net applications. It has simplicity of Java and power of C++.

Course Objectives:

Students will be able to:

1. Understand the MS.NET Framework and its internals.
2. Utilize the basic programming concepts of C#.
3. Build strong concepts of OOP's and implement the same in C#.
4. Identify and usage of various libraries in C#.
5. Develop a web applications using C#

UNIT I THE .NET FRAMEWORK 9 hours

Introduction, The Origin of .Net Technology, Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS), Microsoft Intermediate Language (MSIL), Just-In –Time Compilation, Framework Base Classes and Library, Introduction to Windows Form.

UNIT II FUNDAMENTALS OF C- SHARP LANGUAGE (C#) 9 hours

Data Types, Identifiers, Variables, Constants, Literals, Conditionals and Loops: Enumerations and Data Types Conversion, Arrays and Dynamic Arrays, Operators Decision Making statements, select case, Switch and choose statement, Looping Statements. Procedures, Scope and Exception Handling.

UNIT III OBJECT ORIENTED PROGRAMMING 9 hours

Classes and Objects, Fields, properties, methods, Abstraction, Encapsulation Inheritance, Polymorphism, Overloading, Overriding, Shadowing, Constructors and Destructors, Interfaces.

UNIT IV C# USING LIBRARIES 9 hours

Multi-Threading, Networking and sockets, Managing Console I/O Operations, File Handling, Error Handling. Delegates and Events.

UNIT V ADVANCED FEATURES USING C# 9 hours

Web Services, Window Services, Asp.net Web Form Controls, ADO.Net. Distributed Application in C#, Unsafe Mode, Graphical Device interface with C#.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Understand the .NET framework and the technologies that constitute the framework.
2. Implement the basic programming concepts using C#.
3. Apply the object-oriented programming skills in C#.
4. Identify and use various libraries in C#.
5. Design and develop a web-based application.

Textbooks(s)

1. Herbert Schildt, C# 4.0: The Complete Reference, McGraw-Hill, New Delhi, 2010.

2. Marino Posadas, "Mastering C# and .Net Programming", Packt Publishing, Limited, 2016.

Reference Books

1. Jesse Liberty, Programming C#, Second edition, O'Reilly Media Inc, Cambridge, USA, 2002.

2. Paul Deitel, Harvey Deitel, C# 2010 For Programmers, Deitel Developer Series, Fourth Edition, Pearson Education, New Delhi, 2011.

3. Pradeep Tapadiya, ".Net Programming: A Practical guide using C#", Prentice Hall, New Jersey, 2002.

4. Art Gittleman, "Computing with C# and .Net Framework", Second edition, Jones & Bartlett Learning Publisher, 2011.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

OPEN ELECTIVE-I

Open Elective - I

22MBAP301 MANAGEMENT AND ORGANIZATIONAL BEHAVIOR

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

This course introduces the students the fundamentals of management, its process. The course also facilitates the students in understanding individual, group behavior and organizational culture and climate impacting organizational performance.

Course Objectives:

Students will be able to:

1. Understand the fundamentals of management and its ethical and social obligations.
2. Explain the dimensions of the planning-organizing-leading-controlling (P-O-L-C) framework.
3. Describe how individual personality and behaviour impacts the typical contemporary work experience.
4. Understand group behaviour in organizations, including communication, leadership, power and politics, conflict, and negotiations.
5. Explain the impact of stress, organizational culture, and climate on organizational performance.

UNIT I FOUNDATIONS OF MANAGEMENT 9 hours

Concepts and Evolution of Management thoughts Different Schools of Thoughts; Classical; Behavioral Science Approach; Systems Approach; Contingency and Modern Theories. Management Levels; Managerial Roles and Skills; Ethics and Social Responsibilities of Business.

UNIT II MANAGERIAL FUNCTIONS 9 hours

Process - Problems – Components – Planning – Making It Effective. Decision Making – Process – Techniques.; Organization- as a process and structure, Determinants of Organization Structure; Authority, Responsibility, Delegation, Centralization and De-centralization; Span of Control; Types of Organization Structures; Line & Staff, Functional, Divisional, Matrix and Network; Departmentations, Controlling- Process, types and techniques.

UNIT III INDIVIDUAL BEHAVIOUR 9 hours

Concept, Nature and scope; Understanding Human Behaviour: Personality:, Traits and types(Johari Window); Perception: Factors and Process, Learning – Theories and applications in organizations, Motivation – Concept, Nature and Process, Theories of Motivation: Need Priority Model, Two Factors, Porter and Lawler model, Mc Clelland theory and Mc Gregor’s theory “X” and “Y”.

UNIT IV GROUP BEHAVIOUR 9 hours

Groups –Types of groups, Formation of Groups, Group norms, Cohesiveness and Group effectiveness. Conflict, Types of Conflict and Conflict Resolution (Transactional Analysis). Leadership- Leadership styles, Likert's System theory, Managerial Grid, 2 D and 3D theories, Women Leadership in India, Contemporary issues in Leadership.

UNIT V ORGANIZATIONAL BEHAVIOUR 9 hours

Stress -potential sources, consequences and coping strategies, organizational culture, concept, types of culture, organizational climate VS organizations culture, factors contributing towards creating and sustaining culture.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Apply theoretical models and concepts to current management practices, problems and issues; and to use critical reflection to gain deeper understanding of issues.
2. Analyze major environmental and social pressures and challenges facing managers today; and reflect the same in the planning, organizing, leading, and controlling of the managerial activities.
3. Analyse and compare different models used to explain individual behavior related to motivation and rewards.

4. Assess and design the elements of group behavior including group dynamics, communication, leadership, power & politics and conflict & negotiation.
5. Critically evaluate and create a suitable organizational culture devoid of stress and conflict.

Textbook(s)

1. Management, Stephen P. Robbins, Mary Coulter, Agna Fernandez, Pearson Education, 2018.

2. Organizational Behavior, Fred Luthans, McGraw Hill, 2017.

Reference Books

1. Organizational Behaviour :Human Behaviour at Work, – John W. Newstrom, Tata McGraw Hill,2017.

2. Essentials of Management, Harold Koontz, Heinz Weihrich ,Mark V Cannice,2020.

3. Behavior in Organizations, Jerald Green Berg & Robert A. Baron, Pearson Education,2010.

4. Management and Organizational Behaviour, Subbarao P, Himalaya Publishing House,2017.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

Open Elective – I

22MBAP302 DESIGN THINKING

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

This course deals with various techniques of thinking, enhancement of soft skills and personality development; understanding and applying different concepts in Design Thinking Process of Stanford Model.

Course Objectives:

Students will be able to:

1. Understand the basic concepts of Design Thinking of Stanford Model.
2. Identify the basic concepts of Empathy and the process of sensitization.
3. Develop an understanding of the basic concepts of ideation techniques.
4. Familiarize with the basic concepts of prototyping and testing.
5. Apply the current knowledge from learning about (knowledge) vs. learning to become (skills and mindsets).

UNIT I INTRODUCTION TO DESIGN THINKING 9 hours

Open-mindedness; Developing Design Thinking Mindset; Principles of Design Thinking; Primer on Design Thinking; SWOC Analysis for Self-Awareness.

UNIT II EMPATHY & DEFINE 9 hours

Definition and Components of Empathy; Interrelatedness of Components; Steps in Empathy process; Assessment tools; Roots of Empathy (Case studies); Decision making process; Research Components; Hypothesis (Interview, team formation & benefits), Defining Problem Statement, Application of “How might we Statements”.

UNIT III IDEATION TECHNIQUES 9 hours

Innovation and Creativity: Ideation Techniques - Role-play; Brainstorming; Pooling Ideas-Idea Clustering; Prioritizing ideas; Evaluation of ideas - Pros and Cons; Criteria for idea Ranking; Analyzing; Synthesizing and integrating the ideas. Mind-mapping the experiences, Flaring & Focus; Introduction to “Yes but” – “Yes and”, Impact of Visuals; Exploring resources, Timeline, Lessons from Creative Business Legends: CEOs of Alibaba, Facebook, Apple, Microsoft, Space-X etc.

UNIT IV PROTOTYPING- BUSINESS MODELLING 9 hours

Innovation and Competitive uniqueness; Building artifacts; Real time evaluation; Bringing idea to the life; Use of Visual Clippings; Involve the tester in prototype; initial insight; Market Testing.

UNIT V REFLECTIVE THINKING 9 hours

Do it Now- Reflect- Do it Better; DT is a team sport; develop a coach-like stance; Altruistic Approach. Presentation of 1. My Business Idea (Big Picture- Vision- Mission (Connecting Dots)). 2. Business Model Presentation. 3. Assessment. 4. Dissertation/Record.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Understand the basic concepts of Design Thinking and develop Self Awareness.
2. Empathize, get sensitized and identify the problems.
3. Apply wild ideas, defer judgement and build on ideas of others.
4. Transform innovative ideas into tangible prototypes.
5. Understand, implement, and integrate Design Thinking principles into both personal and professional life.

Textbook(s)

1. Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation, Tim Brown, Harper Business, 2009.

2. The Design of Business: Why Design Thinking is the Next Competitive Advantage, Roger L. Martin, Harvard Business Review Press; Third Edition, 2009.

3. “Design Thinking-A Practical Approach” proprietary material-2018, Stanford Tool Kit.

Reference Books

1. “Fourth Eye” by Pradeep Khandwala.
2. “Action Research” by Eileen Ferrance, “Themes in Education” Northeast and Islands Regional Educational Laboratory Brown University.
3. “Introduction to Life Skills Education”- NCERT Training Package.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

Open Elective - I

22MBAP303 MANAGEMENT INFORMATION SYSTEM

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

This course introduces the various information and communication technologies. Students will study and practice using modern information technologies. Students will examine how information systems are used to solve problems and make better business decisions and apply these concepts to analyze business cases. Students will able to know information technology security tools and techniques.

Course Objectives:

Students will be able to:

1. Understand the basic concepts of Information Systems and how they can provide strategic advantages.
2. Recognize the database approach to enhance business processes and decision-making..
3. Explore the system development life cycle and strategies for business growth.
4. Identify system vulnerabilities and understand how communication impacts decision-making.
5. Analyse the management challenges, control methods, and how to create a security framework.

UNIT I INTRODUCTION AND FOUNDATION CONCEPTS 9 hours

Foundations of information systems (IS) in business System concepts, Components of an IS, IS Resources, fundamental roles of IS applications in business – trends in IS – types of IS – managerial challenges of information technology. Competing with information technology (IT) Fundamentals of strategic advantage – strategic uses of IT – the value chain and strategic IT – using IT for strategic advantages – the basics of doing business on the Internet.

UNIT II DATA BASE AND INFORMATION MANAGEMENT 9 hours

Data in a Traditional file Environment, The Database Approach to Data Management; Role of databases in business performance and decision making, Manage data Resources. The Role of Information System in Business Today, Perspectives on Information Systems, Contemporary Approaches to Information Systems, Organization and Information Systems.

UNIT III MIS DEVELOPMENT PROCESS 9 hours

System development – System Life cycle method, Structured Development method, Developing Business/IT Strategies Planning for competitive advantage – business models and planning – Business/IT planning – Business application planning – Implementing IT–IS development – the

Systems approach – the Systems Development Cycle – Prototyping – Systems development process – End-user development – implementing new systems – Software development.

UNIT IV INFORMATION SYSTEMS 9 hours

Computers in Management – MIS Office automation – Decision Support System – Decision support techniques, Decision making and Role of MIS, Group Decision Support Systems (GDSS).
Applications: Human Resource information system – Financial information system –Marketing information system, Technologies and Tools for Protecting Information Resources.

UNIT V SYSTEM AUDIT & MANAGEMENT CHALLENGES 9 hours

Security and ethical challenges– computer crime – privacy issues –health issues – Security management of IT – tools of security management -Verification and Validation–– security measures - Ethical and Social Issues in Information System - Enterprise and global management of IT Managing the IS function – failures in IT management – the international dimension in IT management – Cultural, political, and geo-economic challenges Global business/IT strategies and applications – global IT platforms.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Understand the Information system concepts and strategic advantage.
2. Elucidate and learn about database and Information Management
3. Differentiate the about the Systems development cycle and MIS Development Process.
4. Investigate the system vulnerabilities and various methods of communications in decision making process.
5. Analyze the management challenges and security issues.

Textbook(s)

1. Management Information System Paperback (2018) by C. Laudon Kenneth (Author), P. Laudon Jane (Author). Pearson Publications.
2. Management Information Systems Paperback 11 edition (2017) by James A. O'Brien (Author), George M. Marakas (Author), Ramesh Behl (Author). McGraw Hill Education
3. Stair, R. M. & Reynolds, G. W. (2001). Principles of Information Systems, 5e, Singapore:Thomson Learning.

Reference Books

1. Management Information Systems, Gordon B. Davis & Margrethe H. Olson, Tata McGrawHill, 2006.
2. Management Information Systems Text & Cases, W S Jawadekar, Tata McGraw-Hill, 2009
3. Introduction to Information Systems, Rainer, Turban, Potter, WILEY-India, 2006.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

Open Elective - I

22MBAP304 E-COMMERCE AND DIGITAL MARKETS

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

The course focusses on e-commerce and is divided into five sections to deliver key aspects of E – commerce such as business models, enablers, and supply chain, Social, Political & Ethical Issues of E-Commerce, E-commerce systems and Digital Markets. Each section delves deeply into various intricacies of doing business over the electronic media.

Course Objectives:

Students will be able to:

1. Introduce the concept of e-business and the business models used in e-commerce.
2. Elucidate about the e-commerce enablers and infrastructure.
3. Enable the students to learn about supply chain management used by e-commerce players.
4. Analyze the socio, political and ethical issues in e-commerce.
5. Develop an insight into e-markets and e-commerce systems.

UNIT I E – COMMERCE BUSINESS MODELS 9 hours

Introduction E-Business - Origin and Need of E-Commerce, – E– commerce v/s Traditional Commerce Factors affecting E -Commerce, Business dimension and technological dimension of E-Commerce, E-Commerce frame work. The Revolution Continues, E-commerce Business Models and Concepts, B2C business models, B2Bmodels, B2G, G2C, Business models for emerging Ecommerce area – customer to customer businessmodel, P2P business model, M-commerce models. IT in business – functional business systems – cross-functional enterprise systems and applications – e-Business models - Enterprise e-Business systems

UNIT II E – COMMERCE ENABLERS 9 hours

E- Commerce enablers, internet and its impact on business strategy Pre and Post Covid-19 Pandemic - industry structure, industry value chain, firm value chain. E-commerce Infrastructure: The Internet, Web, and Mobile Platform.

UNIT III SUPPLY CHAIN MANAGEMENT IN E – COMMERCE 9 hours

B2B E-commerce: Supply Chain Management and Collaborative Commerce. – Introduction to Customer relationship management (CRM) -Building an E-commerce Presence: Web Sites, Mobile Sites, and Apps, E-commerce Marketing Communications -Pre and Post Covid-19 Pandemic

UNIT IV SOCIAL, POLITICAL, AND ETHICAL ISSUES 9 hours

Ethical, Social, and Political Issues in E-commerce, Online Retailing and Services, Online Content and Media, Social Networks, Auctions, and Portals.

UNIT V E-MARKETS 9 hours

Factions, e-Markets Vs Traditional Market, e-Markets Success factors, e-Market Technology Solutions. E-Procurements: The purchasing process, Developments in IT purchasing, e-Procurement-Models, e- procurement- Solutions – E-Commerce systems: E-Commerce systems – Essential e-Commerce processes – electronic payment processes - e-Commerce application trends – Web store requirements – clicks-and-bricks in e-Commerce.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Understand the concepts of e-business and the business models used in e-commerce.
2. Learn about the e-commerce enablers and infrastructure.
3. Develop an insight into supply chain management.
4. Analyze into the socio, political and ethical issues in e-commerce.
5. Develop an understanding of e-markets and e-commerce payment systems.

Textbook(s)

1. Laudon Kenneth C., E-Commerce: Business, Technology, Society, prentice Hall of India, 15th edition, 2019.
2. Bhanver, J.,& Bhanver, K., Click!: The Amazing Story of India's E-commerce Boom and Where it's Headed.: Hachette, 2017.

Reference Books

1. Bhaskar, B. Electronic commerce: Framework, technologies and applications, 3rd edition New Delhi: Tata McGraw Hill Education, 2009.
2. Erisman, P., Six Billion Shoppers: The Companies Winning the Global ECommerce Boom. Macmillan, 2017.
3. Kalakota, R., & Whinston, A. B., Electronic commerce: A manager’s guide. New Delhi: Pearson Education, 2009.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

Open Elective - I

22MBAP305 ENTREPRENEURSHIP DEVELOPMENT AND PROJECT MANAGEMENT

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

This course presents the concepts, characteristics and role of the entrepreneur, and the various aspects of entrepreneurship, invention and importance of innovation in entrepreneurship and expansion, role, importance, and challenges of women entrepreneurs in India. This course provides an overview of various aspects of project management and different stages in the project management process.

Course Objectives:

Students will be able to:

1. Understand the theoretical foundations of entrepreneurship development and design thinking.
2. Develop the skills necessary to create a comprehensive business plan.
3. Familiarize with key concepts of project management.
4. Explore sources of project funding and the government's role in fostering entrepreneurship.
5. Explain effective methods for project implementation.

UNIT I FOUNDATIONS OF ENTREPRENEURSHIP 9 hours

Definition, Scope and Importance – Characteristics of entrepreneur - The Cultural and social Environment — Functions of Entrepreneur, Types of Entrepreneurship, Role of entrepreneurship in economic development, Theories of Entrepreneurship, Innovation and Entrepreneurship; managing forinnovation – "Design Thinking", Entrepreneurship as a career, Women Entrepreneurs: Challenges to Woman Entrepreneurs, Achievements of woman Entrepreneurs, role Models of Woman Entrepreneurs.Case studies of Entrepreneurs – successful, failed, turnaround ventures to be discussed in the class.

UNIT II ENTREPRENEURIAL BUSINESS PLANNING 9 hours

Business Planning Process - The business plan as an entrepreneurial tool; Elements of Business Plan, Objectives, Market Analysis, Development of product / idea, Marketing, Finance, Organization & Management, Ownership, Critical risk contingencies of the proposal, Scheduling and milestones. Promotion of a Venture, Stages of Promotion of a Venture.

UNIT III PROJECT MANAGEMENT 9 hours

Project Management: Meaning of project - concepts - categories - project lifecycle, phases - characteristics of a project – project manager - role and responsibilities of project manager. Technical,Financial, Marketing Personnel and Management feasibility Reports.

UNIT IV ROLE OF GOVERNMENT IN PROMOTING ENTREPRENEURSHIP 9 hours

Start Up India, Make in India, Incubation centers, MSMEs & MSME culture in India, Self-reliant India, Mudra NIESBUD, NEDB, SISI, SIDBI, KVIC, DIC, SFC, SIDC,TCO AND IIDC, Venture Capital Funding, and Angel Capitalist.

Various finance schemes available for the MSMEs by the Central and State government.

UNIT V PROJECT IMPLEMENTATION 9 hours

Implementation of projects: Graphic Representation of Project Activities, Introduction to Network Analysis, Management & control of projects, Project scheduling, MIS in project, problems of project implementation, project audit.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Understand the concepts of entrepreneurship development and design thinking.
2. Assess various methods for preparing a business plan.
3. Analyze project management concepts and gain insights into financial institutions in entrepreneurship development.
4. Explore project financing options and the government’s role in promoting entrepreneurship.
5. Evaluate different methods of project implementation.

Textbook(s)

1. Vasanth Desai, Dynamics of Entrepreneurship Development – Himalaya Publishing House, 2009.
2. David H. Holt, Entrepreneurship: New Venture Creation – PHI, 2009.
3. H. Nandan, Fundamentals of Entrepreneurship, PHI, First/e, New Delhi, 2009.

Reference Books

1. S.S. Khanka, Entrepreneurial Development, - S. Chand & Co. Ltd., Ram Nagar, New Delhi. 110055.
2. Raj Shankar, Entrepreneurship Theory and practice, - Vijay Nicole and Tata McGraw Hill,December 2011.
3. Sahay, M.S.Chhikara, New Vistas of Entrepreneurship: Challenges & Opportunities, Excel Books, 2007.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

AUDIT COURSE

Audit Course

22ENGP903 SOFT SKILLS

L	T	P	C
2	0	0	0

Pre-requisite None

Course Description:

Soft skills are the personal attributes that make a student a valuable employee and a wholesome personality. They include aspects like communication, teamwork, problem-solving, and time management. Employers are increasingly looking for employees with strong soft skills, as they are essential for success in the workplace. This course will help students analyze themselves and build soft skills needed for their personal and career success.

Course Objectives:

Students will be able to:

1. Analyze their strengths and skills and build confidence in presenting themselves.
2. Work seamlessly as a team and negotiate for solutions.
3. Think laterally and critically to evaluate a situation and present it with clarity.
4. Write business emails effectively.
5. Prepare holistically for a job interview.

UNIT I SELF-ANALYSIS AND DEVELOPMENT 9 hours

Personal ethics (politeness, empathy, and honesty); self-motivation / building confidence and assertiveness; identifying one's unique selling points (USPs) through skills introspection and recognizing strengths and weaknesses; nurturing strengths and fixing weaknesses; self-introduction.

UNIT II TEAM WORKING AND DYNAMICS 9 hours

Brainstorming techniques, team building, collaboration, and negotiation skills; team role plays (involving negotiation and decision making); group discussion etiquette (greetings and body language), idea generation, and common GD phrases; group discussion practice.

UNIT III THINKING AND REASONING SKILLS 9 hours

Lateral thinking, critical thinking and logical reasoning through texts, images, and videos; Speaking activities (e.g. JAM) involving lateral thinking and reasoning through thought-provoking pictures, videos, cartoons, comic strips or articles.

UNIT IV PRESENTATION SKILLS 9 hours

Presentation etiquette; slides design; and presentation practice.

UNIT V INTERVIEW SKILLS 9 hours

Preparing resume and cover letter for job interviews; interview etiquette: dress code, body language, tone, and greeting; HR interviews: answering common interview questions, practice for HR interviews.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Understand and express themselves with confidence.
2. Work as an active team member.
3. Think and express their views logically and speak on varied topics without hesitations.
4. Prepare business presentations and emails effectively.
5. Attend job interviews with confidence.

Textbook(s)

1. Sabina Pillai and Agna Fernandez; *Soft Skills and Employability Skills*; Cambridge University Press, 2018.
2. Archana Ram, *PlaceMentor*, 2018, Oxford University Press.

Reference Books

1. Karen Kindrachuk, *Introspection*, 2010, 1st Edition.

2. 2. Karen Hough, *The Improvisation Edge: Secrets to Building Trust and Radical Collaboration at work*, 2011, Berrett-Koehler Publishers.

3. 3. Colin Swatridge, *Oxford Guide to Effective Argument and Critical Thinking* 1st Edition, Oxford University Press.

Mode of Evaluation: Assignments and Mid Term Tests