Department of Computer Science

Syllabus
Master of Computer Applications (MCA)
(2021 BATCH)

CHRIST (Deemed to be University), Bangalore
Karnataka, India
www.christuniversity.in

Syllabus for Master of Computer Application (MCA) - 2021-22 approved by the Board of Studies, Department of Computer Science and Academic Council, CHRIST (Deemed to be University), Bangalore, India.

Published by the Centre for Publications, CHRIST (Deemed to be University), Hosur Road, Bangalore, 560 029, India. publications@christuniversity.in

2021

Index

Department Overview

Department of Computer Science of CHRIST (Deemed to be University) strives to shape outstanding computer professionals with ethical and human values to reshape nation's destiny. The training imparted aims to prepare young minds for the challenging opportunities in the IT industry with a global awareness rooted in the Indian soil, nourished and supported by experts in the field.

Vision

The Department of Computer Science endeavors to imbibe the vision of the University "**Excellence and Service"**. The department is committed to this philosophy which pervades every aspect and functioning of the department.

Mission

"To develop IT professionals with ethical and human values". To accomplish our mission, the department encourages students to apply their acquired knowledge and skills towards professional achievements in their career. The department also moulds the students to be socially responsible and ethically sound.

Introduction to the Programme

Master of Computer Applications is a Two year post graduate programme spread over six Trimesters. This programme strives to shape the students into outstanding computer professionals for the challenging opportunities in IT industry. It enables students to evolve from the stereo type thinking to better achievers and prepares them to scale the global standards. Curriculum incorporates the state of the art areas of IT industry to provide opportunity for extended study in an area of specialization.

Programme Objective

- To strengthen the concept of Computer Science and applications for career growth and employability.
- To provide multidisciplinary and application oriented programme.
- To inculcate in students professional and ethical attitude, team work and effective communication skills.
- Students are encouraged to implement independent projects of their own choice and to use latest tools.

Ethics and Human Values

1. Only proprietary or open source software would be used for academic teaching and learning purposes.
2. Copying of programs from internet, friends or from other sources is strictly discouraged since it impairs development of programming skills.
3. Unique Practical (Domain based) exercises ensures that the students don't involve in code plagiarism.
4. Projects undertaken by students during the course are done in teams to improve collaborative work and synergy between team members.
5. Projects involve modularization which initiates students to take individual responsibility for common goals.
6. Passion for excellence is promoted among the students, be it in software development or project documentation.
7. Giving due credit to sources during the seminar and research assignment is promoted among the students
8. The course and its design enforce the practice of good referencing technique to improve the sense of integrity.
9. Courses involving group discussions and debates on ethical practices and human values are designed to sensitize the students in dealing with customers and members within the organization.

Programme Outcomes

PO1: Computational Knowledge : Apply knowledge of computing fundamentals, computing specialisation, mathematics, and domain knowledge appropriate for the computing specialisation to the abstraction and conceptualisation of computing models from defined problems and requirements.

PO2: Problem Analysis: Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

PO3: Design/Development of Solutions: Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

PO4: Conduct Investigations of complex computing problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern Tool usage: Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.

PO6: Professional Ethics: Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices.

PO7: Life-long learning: Recognise the need, and have the ability, to engage in independent learning for continual development as a computing professional.

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

PO9: Communication Efficacy: Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.

PO10: Societal and Environmental Concern: Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practices.

PO11: Individual and Team work: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

PO12: Innovation and Enterpreneurship: Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

Programme Structure of MCA

Trimester I

Course Code	Course Title	No. of Hours /Week	Marks	Credits
MCA131	Digital Logic Fundamentals	4	100	3
MCA132	Probability and Statistics	4	100	3
MCA133	Operating Systems	5	100	4
MCA171	Python Programming (CIA Only)	9	150	5
MCA172	Programming in C (CIA Only)	7	150	5
MCA161A	Introduction to Programming and Problem Solving (Foundation Elective)	3	50	2
MCA161B	Linux Administration (Foundation Elective)			
HOL111	Holistic Education	1		1
Total		**33**	**650**	**23**

Trimester II

Course Code	Course Title	No. of Hours /Week	Marks	Credits
MCA231	Software Engineering	5	100	4
MCA232	Research Methodology (CIA Only)	3	50	2
MCA271	Microprocessor and Interfacing Techniques (CIA Only)	9	150	5
MCA272	Web Stack Development (CIA Only)	9	150	5
MCA273	Database Technologies (CIA Only)	7	150	5
Total		**33**	**600**	**21**

Trimester III

Course Code	Course Title	No. of Hours /Week	Marks	Credits
MCA331	Computer Networks	5	100	4
MCA371	Data Structures in C (CIA Only)	7	150	5
MCA372	Java Programming (CIA Only)	9	150	5
Elective – I (Choose Any One)				
MCA341A	**Stream A:** Introduction to Data Analytics	5	100	4
MCA373B	**Stream B:** Introduction to Artificial Intelligence			
MCA341C	**Stream C:** Introduction to Internet of Things			
MCA381	Project I	5	100	4
HOL311	Holistic Education	1		1
Total		**32**	**600**	**23**

Trimester IV

Course Code	Course Title	No. of Hours /Week	Marks	Credits
MCA471	Mobile Applications (CIA Only)	9	150	5
MCA472	Machine Learning (CIA Only)	7	150	5
Elective – II (Choose Any One)				
MCA441A	**Stream A:** Predictive Analytics	5	100	4
MCA441B	**Stream B:** Data Engineering and Knowledge Representation			
MCA441C	**Stream C:** System simulation for IoT and Sensor Networks			
Elective – III (Choose Any One)				
MCA473A	**Stream A:** Big Data Analytics (CIA Only)	7	150	5

MCA473B	**Stream B:** Natural Language Processing (CIA Only)			
MCA473C	**Stream C:** Embedded Systems and interfacing (CIA Only)			
MCA481	Seminar	3	50	2
Total		**31**	**600**	**21**

Trimester V

Course Code	Course Title	No. of Hours /Week	Marks	Credits
MCA571	Cloud Computing (CIA Only)	7	150	5
Elective – IV (Choose Any One)				
MCA572A	**Stream A:** Spatio-Temporal Data Analytics (CIA Only)	7	150	5
MCA572B	**Stream B:** Neural Networks and Deep Learning (CIA Only)			
MCA572C	**Stream C:** Sensors and Sensor Circuit Design (CIA Only)			
Elective – V (Choose Any One)				
MCA573A	**Stream A:** Quantum Machine Learning (CIA Only)	7	150	5
MCA573B	**Stream B:** Computer Vision (CIA Only)			
MCA573C	**Stream C:** IoT Data Analytics (CIA Only)			
MCA581	Specialization Project	5	100	4
MCA582	Project II	5	100	4
Total		**31**	**650**	**23**

Trimester VI

Course Code	Course Title	No. of Hours /Week	Marks	Credits
MCA681	Industry Project		300	12
Total			**300**	**12**

Trimester-I

MCA131: DIGITAL LOGIC FUNDAMENTALS

Total Teaching Hours for Trimester: 45
Max Marks: 100 **Credits: 3**

Course Objectives

To enable the students to learn the basic functions, principles and fundamental aspects of digital design in terms of digital logic elements and circuits. To provide deep knowledge in designing and analyzing combinational and sequential circuits. The course prepares students to perform the analysis and design of various types of data storage and data transfer circuits.

Course Outcomes

CO1: Interpret different number system, binary codes and digital logic elements
CO2: Acquaint with elementary postulates of Boolean algebra and methods for simplifying Boolean expressions
CO3: Illustrate the procedures for the analysis and design of sequential and combinational circuits

Unit-1 **Teaching Hours: 9**
NUMBER SYSTEM AND BINARY CODING
Number system representation: Decimal number system- Binary number system- octal number system- hexadecimal number system- number system representation- number system conversion- signed number representation- complement system: 1's complement – 2's complement- 9's complement – 10's complement- Binary arithmetic operations: addition- subtraction- multiplication- division- Coding schemes: BCD, Gray code and ASCII code.

Unit-2 **Teaching Hours: 9**
BOOLEAN LOGICS AND LOGIC GATES
Introduction - Boolean Logics and Logic Gates -Universal Gates and properties- Boolean Algebra Theorems - Boolean Function - Minterms- Maxterms- Karnaugh Map (K-Map)- Sum of Products (SOP) and Product of Sums (POS). Don't Care Conditions.

Unit-3 **Teaching Hours: 9**
COMBINATIONAL CIRCUITS
Introduction- Combinational logic- Half Adder – Full adder- Half subtractor-Binary adder -Binary adder subtractor- BCD adder- Binary multiplier- Encoder- Decoder- Multiplexer- Demultiplexer- BCD to seven segment display

Self-learning: Full subtractor and realization of adder, subtractor and multiplier using NAND gates.

Unit-4 **Teaching Hours: 9**

SEQUENTIAL CIRCUITS

Sequential logic- Introduction-Latches- Clock - Types of Clock – positive, Negative edge triggered - Flip-Flops (with Timing Diagram) - SR Flip Flop – D Flip Flop – JK Flip Flop -Edge Triggered Flip Flops- Master-Slave JK Flip-Flop-Timing diagram.

Unit-5 **Teaching Hours: 9**

REGISTERS AND COUNTERS

Introduction to Register and Counter – Shift registers – Serial Transfer – Modes of operations- Serial in Serial Out (SISO) -Serial in Parallel out (SIPO) – Parallel in Serial Out (PISO)- Parallel in Parallel out (PIPO)- Bidirectional Shift Register -Synchronous Counter - Asynchronous Counters - Binary Counters - Up/Down counter -BCD counter.

Self -learning: Shift register with Parallel Load

Essential Reading:

[1] Donald P Leach, Albert Paul Malvino, Goutam Saha, Digital Principles and Applications, 8th Edition, Tata Mc Graw-Hill, 2018

Recommended Reading:

[1] Mano, Morris M and Kime Charles R., Logic and Computer Design Fundamentals, Pearson education, 2nd edition, 2015.
[2] Bartee, Thomas C, Digital Computer Fundamentals, Tata Mc Graw-Hill, 6th edition, 2016.
[3] William Stallings, Computer Architecture and Organization, PHI, 8th Edition, 2016.
[4] David A. Patterson and John L.Hennessey, Computer Organization and Design, Morgan Kauffman / Elsevier, 5th edition, 2016.
[5] Ata Elahi ,Computer Systems Digital Design, Fundamentals of Computer Architecture and Assembly Language, Springer International Publishing, 2017

Web Resources:

[1] NPTEL - Youtube channel : Electronics - Digital Circuits and Systems
[2] https://www.youtube.com/watch?v=CeD2L6KbtVM&list=PL803563859BF7ED8C
[3] NESO Academy - Youtube Channel : Digital Electronics
[4]https://www.youtube.com/watch?v=M0mx8S05v60&list=PLBlnK6fEyqRjMH3mWf6kwqiTbT798eAOm

MCA132: PROBABILITY AND STATISTICS

Total Teaching Hours for Trimester: 45
Max Marks: 100 **Credits: 03**

Course Objectives
The main aim of this course is to provide the grounding knowledge of statistical methods for data analytics. Data summarization, probability, random variables with properties and distribution functions were included. Sampling distributions and their applications in hypothesis testing advanced statistical methods like ANOVA and correlation and regression analysis were included.

Course Outcomes
CO1: Summarize and present the data using exploratory data analysis
CO2: Establish the relationship between the frequency distributions(data) and distribution functions (Model) and important characteristics
CO3: The sampling distributions and their applications in hypothesis testing
CO4: Identify the relationship between the variables and modeling the same

Unit – 1 **Teaching Hours: 10**
EXPLORATORY DATA ANALYSIS
Definition of Statistics, applications, data types and measurements, graphical representation of data using histogram, line diagram, bar diagram, time series plots; measures of central tendency and dispersion; coefficient of skewness and kurtosis and their practical importance.

Unit – 2 **Teaching Hours: 15**
PROBABILITY AND RANDOM VARIABLES
Random experiment, sample space and events. Definitions of probability, addition and multiplication rules of probability, conditional probability and some numerical examples; Random variables: Definition, types of random variables, pmf and pdf of random variables; Mathematical expectation: mean, variance, covariance, mgf and cgf of a random variable(s); Probability distributions: Binomial, Poisson and Normal distributions with their important characteristics.

Unit – 3 **Teaching Hours: 10**
SAMPLING DISTRIBUTIONS
Concepts of population, sample, parameter, statistic, and sampling distribution of a statistics; Sampling distribution of standard statistics like, sample mean, variance, proportions etc. t, F and Chi- square distributions with statistical properties

Unit – 4 **Teaching Hours: 10**
TESTING OF HYPOTHESIS
Statistical hypotheses-Simple and composite, Statistical tests, Critical region, Type I and Type II errors, Testing of hypothesis – null and alternative hypothesis, level of significance, Test of significance using t, F and Chi-Square distributions (large sample case). Concept of interval

estimation and confidence interval construction for standard population parameters like, mean, variance, difference of means, proportions (only large sample case).

Essential Reading:

[1] Gupta S.C & Kapoor V.K, Fundamentals of Mathematical statistics, SultanChand & sons, 2020.
[2] Douglas C Montgomery, George C Runger, Applied Statistics and Probability for Engineers, Wiley student edition, 2004.

Recommended Reading:

[1] Freund J.E, Mathematical statistics, Prentice Hall, 2001.
[2] Levine, David M; Berenson, L Mark; Stephen, David, Statistics for Managers Using Microsoft Excel, 2nd Edition, PHI, New Delhi, 2012.

MCA133: OPERATING SYSTEMS

Total Teaching Hours for Trimester: 60
Max Marks: 100 **Credits: 04**

Course Objectives
To acquire the fundamental knowledge of the operating system architecture and components.

Course Outcomes
CO1: Demonstrate the fundamental principles of operating system, system structure, system calls, programs and system boot
CO2: Evaluate the process scheduling, Thread scheduling, scheduling criteria, critical section problems to calculate the processing time effectively
CO3: Implement deadlock system and multiple memory management strategies
CO4: Apply the appropriate file system for overall management of any operating system
CO5: Analyze the file management concepts using LINUX

Unit – 1 **Teaching Hours: 12**
FUNDAMENTALS
Operating system definition, Computer system organization, structure, architecture and operations, process and storage management, Protection and security, Distributed systems, Special purpose systems, Computing Environments, Linux Operating Systems. System structure: operating system services, user interface, system calls, system programs, OS design, Implementation and structure, virtual machines, system boot.

Unit – 2 **Teaching Hours: 12**
PROCESS SCHEDULING
Process concepts, scheduling, operations on processes, Inter process communication, Examples of IPC systems, Communication in client server systems, Threads, Multi threading models, threading issues, Basic concepts, scheduling criteria, scheduling algorithms, Thread scheduling, Multiple-processor scheduling.

Unit – 3 **Teaching Hours: 12**
PROCESS COORDINATION
Critical section problems, Peterson solution, Introduction to semaphores, classic problems of synchronization, Monitors, synchronization examples, atomic transaction, System model, deadlock characterization, methods for handling deadlock, deadlock prevention, avoidance, detection and recovery from deadlock.

Unit – 4 **Teaching Hours: 12**
MEMORY MANAGEMENT
Memory Management Strategies: Background, swapping, Memory allocation, Paging, Structure of the page table, Segmentation. Virtual Memory Management: Demand paging, Page replacement, allocation of frames, thrashing, memory mapped files, Allocating kernel memory.
Memory management concepts can be demonstrated using Linux.

Unit – 5 **Teaching Hours: 12**

FILE MANAGEMENT

File concepts, access methods, directory and disk structure, File system mounting, File sharing, Protection, directory implementation, allocation methods, free-space management. I/O Systems, I/O hardware, Application I/O Interface, Kernel I/O subsystem, Transforming I/O requests to hardware operations.

Essential Reading:

[1] Silberschatz, P.B. Galvin, G. Gagne, Operating System Concepts, Wiley-India, 9th Edition, 2015.
[2] Robert Love, Linux System Programming, O'Reilly, 2014.

Recommended Reading:

[1]William Stallings, Operating Systems: Internals and Design Principles, Pearson, 7th Edition, 2013.
[2] Andrew S Tanenbaum & Herbert Bos, Modern Operating Systems, Pearson, 4th Edition, 2014.

MCA171: PYTHON PROGRAMMING

Total Teaching Hours for Trimester: 105
Max Marks: 150 **Credits: 05**

Course Objectives
This course covers programming paradigms brought in by Python with a focus on Regular Expressions, List and Dictionaries. It explores the various modules and libraries to cover the landscape of Python programming.

Course Outcomes
CO1: Demonstrate the use of the built -in objects of Python
CO2: Demonstrate significant experience with the Python program development environment.
CO3: Understand and implement the basic methods of python modules like NumPy, Matplotlib.

Unit – 1 **Teaching Hours: 21**

INTRODUCTION TO PYTHON DATA STRUCTURES
Underlying mechanism of Module Execution- Sequences, Mapping and Sets- Dictionaries- Functions - Lists and Mutability- Problem Solving Using Lists and Functions. Custom and built-in modules.

Lab Exercises:

1. Demonstrate use of Python data structures.
2. Demonstrate Lists comprehensions
3. Demonstrate Dictionary comprehension

Unit – 2 **Teaching Hours: 21**

OBJECT ORIENTED PROGRAMMING USING PYTHON AND REGULAR EXPRESSIONS
Classes: Classes and Instances-Inheritance—Polymorphism- Abstract Classes-Exceptional Handling- Regular Expressions using "re" module.

Lab Exercises:

4. Demonstrate use of object- oriented programming concepts
5. Demonstrate exceptional handling.
6. Demonstrate use of lambda functions.

Unit – 3 **Teaching Hours: 21**

INTRODUCTION TO NUMPY, PANDAS
Computation on NumPy-Aggregations-Computation on Arrays-Comparisons, Masks and Boolean Arrays-Fancy Indexing-Sorting Arrays-Structured Data: NumPy's Structured Array. Introduction to Pandas Objects-Data indexing and Selection-Operating on Data in Pandas-Handling Missing Data-Hierarchical Indexing.

Lab Exercises:

7. Demonstrate use of custom modules.
8. Implement "re" module.
9. Demonstrate use of "Numpy".

Unit – 4 **Teaching Hours: 21**

MATPLOTLIB and GUI PROGRAMMING

Basic functions of Matplotlib-Simple Line Plot, Scatter Plot. Introduction to Tkiner module-Root Window-Widgets-Button-Label-Message-Text-Menu-Listboxes-Spinbox-Creating tables.

Lab Exercises:

10. Implement Pandas to demonstrate data handling and indexing.
11. Demonstrate use of "Matplotlib" modules to plot line and scatter plot.
12. Create an GUI application using all the appropriate widgets required.

Unit –5 **Teaching Hours: 21**

INTRODUCTION TO DJANGO FRAMEWORK AND DATABASE PROGRAMMING

Introduction-Web framework-creating model to add database service- Django administration application.

Basic Database Operations and SQL, Databases and Python, The Python DB-API, Connection Objects Databases and Python: Adapters Examples of Using Database Adapters, A Database Adapter Example Application.

Lab Exercises:

13. Create a web application using Django framework.
14. Establish database connectivity for an GUI application and demonstrate data manipulation and visualization.

Essential Reading:

[1] Zhang.Y, An Introduction to Python and Computer Programming, Springer Publications, 2017.
[2] Wesely J.Chun, Core Python Application Programming, Prentice Hall, 3rd Edition, 2019.

Recommended Reading:

[1] Mark Lutz , Programming Python, O'Reily Media Inc., 2019.
[2] T. R. Padmanabhan, Programming with Python, Springer Publications, 2019.

Web Resources:

[1] https://docs.python.org/3/tutorial/

MCA172: PROGRAMMING IN C

Total Teaching Hours for Trimester: 90
Max Marks: 150 **Credits: 5**

Course Objectives

Major objective of this course is to provide extensive knowledge of C programming language to the students. It helps in developing the ability to solve computational problems through programs. Lab component is included to give hands-on experience to the students.

Course Outcomes

CO1: Apply control structures appropriately to solve problems
CO2: Ability to understand functional code organization
CO3: Construct code involving arrays, structures and pointer concepts

Unit-1 **Teaching Hours: 18**

C CONTROL STRUCTURES

History of C - Memory concepts - Constants, variables, data types and keywords - Instructions and operators - Decision control structure - if… else construct - Loop control structure - For loop - While loop - Case control structure - Switch case - Break - Continue

Lab Exercises:

1. Implement decision control structure - if… else, nested if… else
2. Implement loop and case control structure

Unit-2 **Teaching Hours: 18**

FUNCTIONS AND POINTERS

Functions - Library functions - Function definitions - Prototype - Scope - Storage classes -Call by value - Pointers variable - Definition and initialization - Pointer operators - Calling function by reference - const qualifier with pointers - sizeof operator - Pointer arithmetic - Pointers to functions - Recursion - Recursion and stack

Lab Exercises:

3. Implement function concept
4. Implement pointer concept using function

Unit-3 **Teaching Hours: 18**

ARRAYS AND STRINGS

Arrays - Definition - Initialization - 2D arrays - Memory map of 2D arrays - Pointers and 2D arrays - Pointers to arrays - Passing Arrays to functions - Array of pointers - Three dimensional arrays - Strings - Characters - Character handling library - String I/O - String conversion - String

comparison - String search - Pointers and strings - 2D array of strings - Array of pointers to strings - Passing strings to functions

Lab Exercises:

5. Implement array concept - single and 2D
6. Implement string manipulations - library and user defined

Unit-4 **Teaching Hours: 18**

STRUCTURES, UNIONS, ENUMS AND BIT OPERATIONS

Structure definitions - Initializing structures - Accessing structure members - Array of structures - Pointers to structures - Using structures with functions - Self referential structures - typedef - Unions - Bitwise operators - Bit fields - Enumeration constants

Lab Exercises:

7. Implement concept of structures and union to understand difference between them
8. Implement bit wise operations

Unit-5 **Teaching Hours: 18**

CONSOLE I/O, FILE HANDLING AND PREPROCESSORS

Types of I/O - Formatted and unformatted console I/O functions - Printing integers, floats and strings - Conversion specifiers - Reading formatted input - Command line arguments - File processing - Data hierarchy - File and streams - File operations - Sequential-Access file - Random-Access file - Error handling - Stderr - Exit A case study - Preprocessors - symbolic constants and macros - File inclusion - Conditional compilation

Lab Exercises:

9. Implement different file related operations
10. Implement a sample case study: e.g., Bank transaction processing system, Hospital appointment system, Hotel booking system, etc

Essential Reading:

[1] P. J. Deitel, H. M. Deitel, C: How to Program, Pearson Prentice Hall, 9th Edition, 2021.
[2] Byron Gottfried, Programming with C, McGraw Hill, 4th Edition, 2018.

Recommended Reading:

[1] Herbert Schildt, The Complete Reference C, Mc Graw Hill, 4th Edition, 2000.
[2] Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language, Pearson, 2nd Edition, 2012.
[3] Yashavant Kanetkar, Let us C, BPB, 17th Edition, 2020

Web Resources:

[1] https://github.com/pdeitel/CHowToProgram9e
[2] https://www.programiz.com/c-programming

MCA161A: INTRODUCTION TO PROGRAMMING AND PROBLEM SOLVING

Total Teaching Hours for Trimester: 30
Max Marks: 50 **Credits: 2**

Course Objectives
The course introduces fundamentals of programming, different types of problem-solving concepts and programming structures to build logic for suitable computational problems.

Course Outcomes
CO1: Demonstrate the systematic approach for problem solving.
CO2: Apply different programming structure with suitable logic for computational problems.

Unit-1 **Teaching Hours: 10**
INTRODUCTION TO PROBLEM SOLVING AND PROGRAMMING
Types of problems, Problem solving in every day, Difficulties in with problem solving. Constants, variables, data types, Data storage, operators, expressions. Organizing the solution, testing the solution, software development life cycle.

Unit-2 **Teaching Hours: 10**
PROBLEM SOLVING WITH LOGIC STRUCTURES
Structuring a solution, modules, cohesion and coupling, local and global variables, Algorithm, flowchart, pseudocode, Sequential logic structure, Solution Development.

Unit-3 **Teaching Hours: 10**
PROBLEM SOLVING WITH DECISION AND LOOP STRUCTURES
The decision logic structure, Straight through logic structure, Positive logic, Negative logic, Logic conversion, Decision Tables. The loop logic structure, nested loops, recursion.

Essential Reading:

[1] Maureen Sprankle and Jim Hubbard, Problem solving and programming concepts, PHI, 9th Edition, 2012.

Recommended Reading:

[1] E Balagurusamy, Fundamentals of Computers, TMH, 2011.

MCA161B: LINUX ADMINISTRATION

Total Teaching Hours for Trimester: 30
Max Marks: 50 **Credits: 02**

Course Objectives
To Enable the students to excel in the Linux Platform

Course Outcomes
CO1: Demostrate the systematic approach for configure the Liux environment
CO2: Manage the Linux environment to work with open source data science tools

Unit-1 **Teaching Hours:10**
RHEL7.5,breaking root password, Understand and use essential tools for handling files, directories, command-line environments, and documentation - Configure local storage using partitions and logical volumes

Unit-2 **Teaching Hours:10**
Swapping, Extend LVM Partitions,LVM Snapshot - Manage users and groups, including use of a centralized directory for authentication

Unit-3 **Teaching Hours:10**
Kernel updations,yum and nmcli configuration, Scheduling jobs,at,crontab - Configure firewall settings using firewall config, firewall-cmd, or iptables , Configure key-based authentication for SSH ,Set enforcing and permissive modes for SELinux , List and identify SELinux file and process context ,Restore default file contexts

Essential Reading:

[1] https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/
[2] https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/

Trimester-II

MCA231: SOFTWARE ENGINEERING

Total Teaching Hours for Trimester: 60
Max Marks: 100 **Credits: 4**

Course Objectives
The Course provides solid fundamental knowledge of software engineering concepts to the students and it prepares them to develop the skills necessary to handle software projects. It also enables the students to apply software engineering principles to develop quality software applications.

Course Outcomes
CO1: Understand the importance of the stages in the software life cycle and the various process models.
CO2: Design software by applying the software engineering principles.
CO3: Develop the quality software using efficient project management.

Unit-1 **Teaching Hours: 12**

PROCESS MODELS, UNDERSTANDING REQUIREMENTS

A generic process model – Defining a framework activity, identifying a Task Set, Process Patterns, Process Assessment and improvement- Prescriptive Process Models-Specialized Process Models. Requirements Engineering-Establishing the groundwork, eliciting requirements, Developing use cases, Building the requirements model – Elements of the requirements Model, Analysis pattern, negotiating requirements, validating requirements-Latest development Methodology-RAD, DevOps, Fish Model-Agile development-Agile Process, Extreme programming, SCRUM Agile Modeling**.**

Unit-2 **Teaching Hours: 12**

DESIGN CONCEPTS

The design process – Software quality guidelines and attributes, The evolution of software design, Design concepts – Abstraction, Architecture, Patterns, Separation of concerns, Modularity, information hiding, Functional Independence, refinement, Aspects, Refactoring, Object Oriented design concepts Design classes, The design Model – Data Design elements, Architectural Design elements, Interface Design Elements, Component-Level Design elements, Deployments level Design elements. Software architecture – Brief taxonomy of Architectural styles, Architectural Patterns.

Unit-3 **Teaching Hours: 12**

COMPONENT LEVEL DESIGN, USER INTERFACE DESIGN

Component –Designing class based components – Basic Design Principles, Component-level Design guidelines, Cohesion, Coupling, Functional design at the Component level, Designing traditional components–Component based development-Domain Engineering, Component qualification, Adaptation, and Composition, Analysis and Design for reuse, classifying and

retrieving components. User Interface Design- The golden rules, User Interface Analysis and Design models, Interface Analysis and Design steps.

Unit-4 **Teaching Hours: 12**

QUALITY MANAGEMENT, TESTING

Software Quality- Achieving software quality- Software testing fundamentals- internal and external view of testing, White-box testing, Basic path testing - control structure testing - Black- box testing- Model Based Testing, Testing for specialized environments– Testing GUIs, Testing of Client-Server Architectures, Testing Documentation and Help facilities, testing for Real-Time Systems, Patterns for software testing.

Unit-5 **Teaching Hours: 12**

PROCESS AND PROJECT METRICS

Metrics in the process and project domains-Process metrics and Software Process improvement Project Metrics-software measurement- Metrics for software quality- Observations on estimation, The project planning process, Software scope and Feasibility, Resources, software project estimation, Decomposition techniques- Empirical estimation models – The structure of Estimation model, COCOMO II Model, Software equation.

Essential Reading:

[1] Pressman S Roger, Software Engineering A Practitioner's Approach, McGraw Hill International Editions, 8th Edition (Indian Edition), 2019.
[2] Sommerville, Ian, Software Engineering, Addison Wesley, 9th Edition, 2011.

Recommended Reading:

[1] Pankaj Jalote, Software Engineering: A Precise Approach, Wiley India, 2010.
[2] Stephen R. Schach, Software Engineering, Tata McGraw-Hill Publishing Company Limited, 2007.

Web Resources:

[1] www.nptel.ac.in

MCA232-RESEARCH METHODOLOGY

Total Teaching Hours for Trimester: 30
Max Marks: 50 **Credits: 2**

Course Objectives

This course starts with an introduction to the basic concepts in research and leads through the various methodologies involved in the research process. It focuses on finding out the research gap from the literature and encourages lateral, strategic, and creative thinking. This course also introduces computer technology and basic statistics required for conducting research and reporting the research outcomes scientifically, with emphasis on research ethics.

Course Outcomes

CO1: Understand the essence of research and the necessity of defining a research problem.
CO2: Apply research methods and methodologies including research design, data collection, data analysis, and interpretation.
CO3: Create scientific reports according to specified standards.

Unit – 1 **Teaching Hours: 6**

RESEARCH METHODOLOGY

Defining research problem: Selecting the problem- Necessity of defining the problem- Techniques involved in defining a problem- Ethics in Research.

Unit – 2 **Teaching Hours: 6**

RESEARCH DESIGN

Principles of experimental design- Working with Literature: Importance- finding literature- Using your resources- Managing the literature-Keep track of references- Using the literature- Literature review- On-line Searching: Database-SCI Finder- Scopus- Science Direct-Searching research articles- Citation Index -Impact Factor -H-index.

Unit – 3 **Teaching Hours: 6**

RESEARCH DATA

Measurement of Scaling: Quantitative-Qualitative,-Classification of Measure scales- Data Collection- Data Preparation.

Unit – 4 **Teaching Hours: 6**

SCIENTIFIC WRITING

Scientific Writing: Significance- Steps- Layout- Types- Mechanics and Precautions- Paper writing for international journals- Writing scientific report.

Unit – 5 **Teaching Hours: 6**

REPORT WRITING

Latex: Introduction-Text-Tables- Figures- Equations- Citations- Referencing and Templates (IEEE style)

Essential Reading:

[1] C. R. Kothari, Research Methodology Methods and Techniques, 4th Edition, New Age International Publishers, 2019.
[2] Zina O'Leary, The Essential Guide of Doing Research, 3rd Edition, SAGE Publications Ltd, 2017.

Recommended Reading:

[1] J. W. Creswell, Research Design: Qualitative, Quantitative, and Mixed Methods Approaches**,** 4th Edition, SAGE Publications, 2014.
[2] Kumar, Research Methodology: A Step by Step Guide for Beginners, 4th Edition, SAGE Publications Ltd, 2014.

MCA271: MICROPROCESSOR AND INTERFACING TECHNIQUES

Total Teaching Hours for Trimester: 105
Max Marks: 150 **Credits: 5**

Course Objectives
To enable the students to incorporate knowledge in the architecture and functional modules of 8085 microprocessor. To dispense an exposure to various 8085 basic and advanced programming techniques. The course further imparts knowledge in differ interfacing techniques using 8255A.

Course outcomes
CO1: Outline the basic elements, functions architecture of 8085 microprocessor and working of each module.
CO2: Examine programming techniques in developing the assembly language program
CO3: Critique and write effective ALP with counter, delay and interrupts**.**
CO4: Implement various peripherals interfacing techniques for microprocessor-based applications.

Unit-1 **Teaching Hours: 9L:12P**
INTRODUCTION TO 8085 AND ALP
Introduction to 8085
Introduction to Microprocessor 8085 –Signals -Address Bus, Data Bus. Block Diagram, Registers, Flags- Decoding and executing an instruction.
Introduction to 8085 programming 8085-programming model, Instruction Classification, Data Format and storage, 8085 instruction Set, Writing simple programs.
Lab Exercises:
1. Write a program to add N one byte numbers.
2. Write a program to add two 8-bit BCD numbers.
3. Write a program to multiply two 8 - bit numbers.
4. Write a program to check whether a byte belongs to the 2-out-of-5 codes. Display FF if it is a 2-out-of- 5 code otherwise 00. (Number is 2-out-of-5 code if the left most three bits are zero and in the remaining five bits there are exactly two 1's)

Unit-2 **Teaching Hours: 9L:12P**
8085 PROGRAMMING
8085 Machine cycles and bus Timings -Addressing Modes- Data Transfer Operations -Arithmetic Operations- Logic Operations - Branch Operations
Lab Exercises:
5. Write a program to add two 32 - bit binary numbers.
6. Write a program to perform linear search over a set of N numbers. Display FF if found otherwise display 00.
7. Write a program to find the first 10 terms of a Fibonacci sequence
8. Write a program to interchange N one bytes of data.

Unit – 3 **Teaching Hours: 9L:12P**

PROGRAMMING TECHNIQUES WITH ADDITIONAL INSTRUCTIONS

Additional data transfer and 16-bit Arithmetic Instructions, Arithmetic operations related to memory, Logic operations: Rotate, Compare, Counters and Time delays, Stack and Subroutines

Lab Exercises:

9. Write a program to subtract a 16 - bit BCD number from another 16 – bit BCD number.
10. Write a program to divide a 16 - bit number by an 8 - bit numbers.
11. Write a program to sort the numbers in ascending and in descending using bubble sort.
12. Write a program to prepare a look-up table for the squares of one -digit BCD numbers.

Unit – 4 **Teaching Hours: 9L:12P**

ARCHITECTURE AND INTERRUPTS OF 8085

Architecture of 8085 MPU

Control & status signals, Power supply and Frequency signals, Externally initiated signals, Serial I/O ports - **ALU**: Timing and Control Unit, Instruction Decoder, Serial I/O Control, Stack, PC, Address/Data Buffers

Interrupts

Introduction – INTR, TRAP, RST 7.5, 6.5, 5.5 – RST, SIM and RIM instructions

Lab Exercises:

13. Write a program to simulate a BCD counter to count from 0 to 100.
14. Write a program to check whether a one-byte number is a palindrome or not.
15. Write a program to simulate a stopwatch with a provision to stop the watch.
16. Write a program to determine the HCF of two one-byte numbers.

Unit – 5 **Teaching Hours: 9L :12P**

PROGRAMMABLE PERIPHERAL INTERFACE 8255A

Block Diagram – Control Logic, Control Word – Modes of operations with examples, Mode 0, Mode 1, BSR Mode, Control word for each mode of operation Programming in 8255A with an example.

Lab Exercises:

17. Write a program to display a rolling message.
18. Write a program to interface a keyboard using 8255A interface.
19. Write a program to interface Seven Segment Display using 8255A interface.

Essential Reading:

[1] Ramesh.S.Goankar, Microprocessor Architecture, Programming & Applications With 8085, 6th Edition, Penram International, 2013. ISBN 81-87972-88-2

Recommended Reading:

[1] Hall.D.V., Microprocessor and Digital System, McGraw Hill Publishing Company, 3rd Edition, 2017.

[2] Charles M Gilmore, Pal Ajit, Microprocessor Principles and Applications, Tata McGraw Hill, 2nd Edition, 2009.

Web Resources:

[1]https://www.youtube.com/watch?v=o6W0opScrKY&list=PLuv3GM6-gsE01L9yDO0e5UhQapkCPGnY3
[2]https://www.youtube.com/watch?v=7pCRYXEgMPQ&list=PLgwJf8NK-2e5vHwmowy_kGtjq9Ih0FzwN

MCA272: WEB STACK DEVELOPMENT

Total Teaching Hours for Trimester: 105
Max Marks: 150 **Credits: 05**

Course Objectives
On completion of this course, a student will be familiar with full stack and able to develop a web application using advanced technologies and cultivate good web programming style and discipline by solving the real world scenarios.

Course Outcomes
CO1: Apply JavaScript, HTML5, and CSS3 effectively to create interactive and dynamic websites
CO2: Describe the main technologies and methods currently used in creating advanced web applications
CO3: Design websites using appropriate security principles, focusing specifically on the vulnerabilities inherent in common web implementations
CO4: Create modern web applications using MEAN

Unit – 1 **Teaching Hours: 20**

OVERVIEW OF WEB TECHNOLOGIES AND HTML5
Internet and web Technologies- Client/Server model -Web Search Engine-Web Crawling-Web Indexing-Search Engine Optimization and Limitations-Web Services –Collective Intelligence – Mobile Web –Features of Web 3.0-HTML vs HTML5-Exploring Editors and Browsers Supported by HTML5-New Elements-HTML5 Semantics-Canvas-HTML Media
Self-Learning:
Introduction to CSS3-CSS2 vs CSS3
Lab Exercises:
1. Develop static pages for a given scenario using HTML
2. Creating Web Animation with audio using HTML5 & CSS3
3. Demonstrate Geolocation and Canvas using HTML5

Unit – 2 **Teaching Hours: 21**

XML AND AJAX
XML-Documents and Vocabularies-Versions and Declaration -Namespaces JavaScript and XML: Ajax-DOM based XML processing Event-Transforming XML Documents-Selecting XML Data:XPATH-Template based Transformations: XSLT-Displaying XML Documents in Browsers - Evolution of AJAX -Web applications with AJAX -AJAX Framework
Lab Exercises:
4. Write an XML file and validate the file using XSD
5. Demonstrate XSL with XSD
6. Demonstrate DOM parser

Unit – 3 **Teaching Hours: 22**

CLIENT SIDE SCRIPTING

JavaScript Implementation - Use Javascript to interact with some of the new HTML5 apis -Create and modify Javascript objects- JS Forms - Events and Event handling-JS Navigator-JS Cookies-Introduction to JSON-JSON vs XML-JSON Objects-Importance of Angular JS in web-Angular Expression and Directives-Single Page Application

Lab Exercises:

7. Write a JavaScript program to demonstrate Form Validation and Event Handling
8. Create a web application using AngularJS with Forms.
9. Implement a single page web application using AngularJS.

Unit – 4 **Teaching Hours: 22**

SERVER SIDE SCRIPTING

Introduction to Node.js-REPL Terminal-Package Manager(NPM)-Node.js Modules and filesystem-Node.js Events-Debugging Node JS Application-File System and streams-Testing Node JS with jasmine

Self-Learning:

Express JS

Lab Exercises:

10. CRUD Operation using AngularJS
11. Implement web application using AJAX with JSON
12. Demonstrate to fetch the information from an XML file with AJAX

Unit – 5 **Teaching Hours: 20**

NODE JS WITH MYSQL

Introduction to MySQL- Performing basic database operation(DML) (Insert, Delete, Update, Select)-Prepared Statement- Uploading Image or File to MySQL- Retrieve Image or File from MySQL

Self-Learning:

CRUD operation using MongoDB

Lab Exercises:

13. Demonstrate Node.js file system module
14. Demonstrate Node.js events
15. Implement Mysql with Node.JS
16. Implement CRUD Operation using MongoDB

Essential Reading:

[1] Internet and World Wide Web:How to Program, Paul Deitel , Harvey Deitel & Abbey Deitel, Pearson Education, 5th Edition, 2018.

[2] HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery), DT Editorial Services, Dreamtech Press, 2nd Edition, 2016.

Recommended Reading:

[1] Chris Northwood, The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer, Apress Publications, 1st Edition, 2018.
[2] Laura Lemay, Rafe Colburn & Jennifer Kyrnin, Mastering HTML, CSS & Javascript Web Publishing, BPB Publications, 1st Edition, 2016.
[3] Alex Giamas, Mastering MongoDB 3.x, Packt Publishing Limited, First Edition, 2017.

Web Resources:

[1] www.w3cschools.com
[2] http://www.php.net/docs.php

MCA273: DATABASE TECHNOLOGIES

Total Teaching Hours for Trimester: 90
Max Marks: 150 **Credits: 5**

Course Objectives
To provide a strong foundation for database application design and development by introducing the fundamentals of database technology.

Course Outcomes
CO1: Understand the basic concepts of database systems, database transactions and related database facilities like concurrency control, data object locking and protocols.
CO2: Analyze the database requirements and develop logical design of the database.
CO3: Apply structured query language to create, retrieve, update and manage a database.

Unit-1 **Teaching Hours: 18**
INTRODUCTION TO DATABASE SYSTEM CONCEPTS AND CONCEPTUAL MODELING
Data models, schemas and instances, DBMS architecture and data independence, Database languages and interfaces, database system environment, Classification of DBMS. Using High-Level Conceptual Data Models for Database Design - Entity Types, Entity Sets, Attributes, and Keys - Relationship Types, Relationship Sets, Roles, and Structural Constraints - Weak Entity Types - ER Diagrams, Naming Conventions, and Design Issues – Relationship Types of Degree Higher than Two - Subclasses, Superclasses, and Inheritance – Enhanced Entity Relationship Model - Relational Database Design by ER- and EER-to-Relational Mapping – Role of Information Systems in Organizations - Database Design and Implementation Process
Lab Exercises:
1. Design Entity Relationship Diagram
2. Basic data retrieval queries

Unit-2 **Teaching Hours: 18**
THE RELATIONAL DATA MODEL AND SQL
SQL Data Definition and Data Types, Specifying Constraints in SQL, Basic Retrieval Queries in SQL, INSERT, DELETE, and UPDATE Statements in SQL, Additional features of SQL. Complex Queries, Triggers, Views, and Schema Modification More Complex SQL Retrieval Queries, Specifying Constraints as Assertions and Actions as Triggers, Views (Virtual Tables) in SQL, Schema Change Statements in SQL.
Lab Exercises:
3. Data retrieval using referential integrity and JOIN
4. Advanced data retrieval using nested queries, sub queries and Views

Unit-3 **Teaching Hours: 18**
RELATIONAL DATA MODEL, DATABASE DESIGN AND INTRODUCTION TO FILE ORGANIZATION

Design Guidelines for Relation Schemas - Functional Dependencies - Normal Forms Based on Primary Keys - Second and Third Normal Forms - Boyce-Codd Normal Form – Multivalued Dependency and Fourth Normal Form - Join Dependencies and Fifth Normal Form – Inference Rules, Equivalence and Minimal Cover - Properties of Relational Decompositions - Nulls and Dangling Tuples - File Organization - Organization of Records in Files - Ordered Indices - B+ Tree Index Files - Static Hashing - Bitmap Indices

Lab Exercises:

5. Aggregate functions
6. Database Design Using Normalization

Unit-4 **Teaching Hours: 18**

TRANSACTION PROCESSING, CONCURRENCY CONTROL AND RECOVERY

Transaction - Introduction to transaction processing- transaction and system concept- Desirable properties of transaction- Transaction support in SQL- concurrency control techniques – Two phase Locking techniques for concurrency- Concurrency Control Based on Timestamp Ordering. Recovery Concepts- NO-UNDO/REDO Recovery Based on Deferred Update- Recovery Techniques Based on Immediate Update- Shadow Paging

Lab Exercises:

7. Stored Procedure – 1 (PL/SQL-1)
8. Stored Procedure – 2 (PL/SQL-2)

Unit-5 **Teaching Hours: 18**

DISTRIBUTED DATABASES AND NOSQL SYSTEMS

Introduction to Distributed database concepts- Types of Distributed Database Systems- Data Fragmentation- Replication- and Allocation Techniques for Distributed Database Design. Overview of Transaction Management in Distributed Databases- Overview of Concurrency Control and Recovery in Distributed Databases

NOSQL Databases

Introduction to NOSQL Systems, The CAP Theorem, Document-Based NOSQL Systems and MongoDB, NOSQL Key-Value Stores, Column-Based or Wide Column NOSQL Systems

Lab Exercises:

9. NOSQL Exercise - 1
10. NOSQL Exercise - 2

Essential Reading:

[1] Elmasri & Navathe, Fundamentals of Database Systems, Addison-Wesley, 7th Edition, 2016.

Recommended Reading:

[1] Korth F. Henry and Silberschatz Abraham, Database System Concepts, McGraw Hill, 6th Edition, 2010.

[2] O'neil Patric, O'neil Elizabeth, Database Principles, Programming and Performance, Argon Kaufmann Publishers, 2nd Edition, 2002.

[3] Ramakrishnan and Gehrke, Database Management System, McGraw-Hill, 3rd Edition, 2003.

Web Resources:

1. www.w3cschools.com
2. https://archive.ics.uci.edu

Trimester - III

MCA331 – COMPUTER NETWORKS

Total Teaching Hours for Trimester: 60
Max Marks: 100 **Credits: 4**

Course Objectives
This course aims to set the foundation for Data Communication in Networks by introducing the network components, topologies, network models, and important protocols based on the TCP/IP model for the internet.

Course Outcomes
CO1: Demonstrate in depth knowledge of network communications based on TCP/IP models.
CO2: Demonstrate a critical understanding of network models with related key protocols, services and applications
CO3: Evaluate different techniques / algorithms of standard network models
CO4: Analyze network protocols for data transmission in various types of networks

Unit-1 **Teaching Hours: 12**
INTRODUCTION TO NETWORKS, THE PHYSICAL LAYER
Introduction: Network Topology, Network Hardware, Network Software: Protocol Hierarchies, Design issues, Connection Oriented Vs Connection less, Service primitives, OSI Reference Model, TCP/IP
Wireless Transmission, Ethernet, Transmission Media, Digital Modulation and Multiplexing, Line codes, Switching.

Unit-2 **Teaching Hours: 12**
THE DATA LINK LAYER
Error Detection and Correction: Types of Error, Error Detection, Parity Check, The Internet Checksum, Cyclic Redundancy Check, Forward Error Correction. Data Link Control Protocols: Flow Control, Error Control, HDLC. ADSL, xDSL. Medium Access Control Sublayer: Static Channel Allocation, Assumptions for Dynamic Channel Allocation, Multiple Access Protocols – Aloha, CSMA, Collision free Protocols, Limited Contention Protocols. Ethernet, Wireless LANS, Repeaters, Hubs, Bridges, Switches, Routers, and Gateways.

Unit-3 **Teaching Hours: 12**
NETWORK LAYER
Routing Algorithms: The Optimality Principle, Shortest Path Algorithm, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing, Broadcast Routing, Multicast Routing.
The Network Layer in the Internet: IPv4 Protocol, IP Addresses, IPv6 Protocol, Internet Control Protocols - ARP, RARP, Label Switching and MPLS, OSPF Protocol, BGP Protocol,

Unit-4 **Teaching Hours: 12**

TRANSPORT LAYER

Transport Service: Transport Service Primitives, Berkeley Sockets. Elements of Transport Protocols: Addressing, Connection Establishment, Connection Release, Error and Flow Control.
The Internet Transport Protocols (UDP): Introduction to UDP, Remote Procedure Call, Real-Time Transport Protocols. The Internet Protocols (TCP): Introduction to TCP, TCP Service Model, TCP Segment Header, TCP Connection Establishment, TCP Connection Release, TCP Connection Management Modelling, TCP Sliding Window Protocol.

Unit-5 **Teaching Hours: 12**

INTERNET APPLICATIONS AND ADVANCED NETWORKS

Electronic Mail, DNS and HTTP: Electronic Mail - SMTP and MIME, Internet Directory Service - DNS, Web Access and HTTP. Internet Multimedia Support: Real-Time Traffic, Voice Over IP, Session Initiation Protocol, Real-Time Transport Protocol (RTP).
Advanced Networks -Case study: IoT, Mobile Networks, SDN.

Essential Reading:

[1] Data and Computer Communications, William Stalling, Pearson International, 10th Edition, 2014.
[2] Computer Networks, Andrew S. Tanenbaum, David J. Wetherall, Pearson New International, 5th Edition, 2014.

Recommended Reading:

[1] Forouzan, Behrouz A., Mosharraf Firouz., Computer Networks A Top-Down Approach, Tata McGraw Hill publications, 1st Edition, 2012.
[2] Prakash C. Gupta, Data communications and Computer Networks, 1st Edition, 5th Reprint, PHI, 2009.

Web Resources:

[1] https://www.geeksforgeeks.org/computer-network-tutorials
[2] https://www.tutorialspoint.com/data_communication_computer_network/index.htm
[3] https://www.guru99.com/data-communication-computer-network-tutorial.html

MCA371-DATA STRUCTURES IN C

Total Teaching Hours for Trimester: 90
Max Marks: 150 **Credits: 05**

Course Objectives
To explore elementary data structures in computer science, and learn to implement them in C. The data structures include linked lists, stacks, queues, trees, heaps, hash tables, and graphs. It also introduces different techniques for searching, traversing trees, hashing, manipulating priority queues, sorting, finding shortest paths in graphs.

Course Outcomes
CO1: Describe common applications for arrays, linked structures, stacks, queues, trees, and graphs
CO2: Illustrate various techniques for searching, sorting and hashing
CO3: Design and implement an appropriate data structures to solve real world problems

Unit – 1 **Teaching Hours: 18**
ELEMENTARY DATA STRUCTURES
Introduction to Pseudo code - Overview of Time & Space Complexity – Recursion – Abstract Data Type - Array - Stack - Queue - Linked lists - Traversing - Searching - Insertion - Deletion - Circular Linked list - Two-way Lists (Doubly) - Linked List Implementation of Stack and Queue - Application of stacks and Queues.
Lab Exercises:
1. Write a program to convert an infix expression to the postfix form.
2. Implement linked list and its operations.

Unit – 2 **Teaching Hours: 18**
SORTING AND SEARCHING
Bubble Sort – Insertion – Selection – Quick – Merge – Linear Search – Binary search – Hashing – Chaining – Collision Resolution – Open Addressing – String Matching Algorithms: Naive, KMP
Lab Exercises:
3. Implement the concept of sorting technique
4. Implement the concept of searching/pattern matching technique

Unit - 3 **Teaching Hours: 18**
GRAPHS & TREES
Representation of Graphs - Operations on Graphs - Depth First and Breadth First Search - Topological Sort - Minimum Spanning Tree Algorithms - Binary Tree - Traversing Binary Trees - Binary Heap - Priority Queue - Heap sort.
Lab Exercises:
5. Implementation of Minimum Spanning Tree
6. Implementation of BFS and DFS

Unit – 4 **Teaching Hours: 18**
SEARCH TREES

Binary Search Trees - Searching, Inserting and deleting in Binary Search Trees - AVL Trees - AVL Balance Factor, Balancing Trees, AVL node structure, AVL Tree Rotate Algorithms

Lab Exercises:

7. Implementation of BST
8. Implementation of AVL Tree

Unit –5 **Teaching Hours: 18**

ADVANCED DATA STRUCTURES

B Trees – Operations on B Trees - B+ Trees - Red-Black Trees - Properties of Red-black Trees - Rotations - Insertion - Deletion operations

Lab Exercises:

9. Implementation of B Trees
10. Implementation of B+ Trees

Essential Reading:

[1] Gilberg, F Richard & Forouzan, A Behrouz, Data Structures A Pseudocode approach with C, Cengage. 2nd Edition, 2008.
[2] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, MIT Press, 3rd Edition, 2009.

Recommended Reading:

[1] Horowitz Sahni Anderson-Freed, Fundamental of Data Structures in C, Universities Press, Reprint, 2008.
[2] Yashavant Kanetkar , Data Structures Through C, BPB Publications, 2019.
[3] Robert Sedgwick, Kevin Wayne, Algorithms, Addison-Wesley Publishing Company. 4th Edition, 2011.

Web Resources:

[1] https://www.hackerrank.com/domains/data-structures
[2] https://nptel.ac.in/Programming and Data Structure by Dr.P.P. Chakraborty, Department of Computer Science and Engineering, IIT Kharagpur

MCA372: - JAVA PROGRAMMING

Total Teaching Hours for Trimester: 105
Max Marks: 150 **Credits: 05**

Course Objectives
This course will help the learner to gain a sound knowledge in object-oriented principles, GUI application design with data base, enterprise application design with java beans and Servlets.

Course Outcomes
CO1: Understanding and applying the principles and practice of object-oriented programming in the construction of robust maintainable programs
CO2: Competence in the use of Java Programming Language in the development of small to medium sized applications that demonstrate professionally acceptable coding and performance standards CO3: To prepare the students to address the challenging requirements coming from the enterprise applications

Unit – 1 **Teaching Hours: 21**

Introduction to Object Oriented Programming (OOP) and Classes

Introduction to Object Oriented Programming(OOP)

Object-Oriented Programming(OOP) Principles- The Evolution of Java- Buzzwords of Java - Class Fundamentals - Declaring Objects - Introducing Methods - Overloading methods – Constructors - Parameterized Constructors - this Keyword

Class features

Garbage Collection - the finalize () Method - Introducing Access Control - Understanding static - Introducing nested and inner classes - String class - String Buffer Class - Command Line Arguments

Lab Exercises:

1. Identify a domain of your choice, list out ten entities in the domain. For each entity, identify minimum 10 attributes and assign the data type for each attribute with proper justification.
2. Implement the concept of class, data members, member functions and access specifiers.
3. Implement the concept of function overloading & Constructor overloading.

Unit – 2 **Teaching Hours: 21**

Inheritance in Java, Interfaces and Packages, Exception Handling in Java

Inheritance in Java

Inheritance Basics - Multilevel Hierarchy- Using super - Method overriding - Dynamic Method Dispatch- Abstract keyword- Using final with inheritance - the Object Class.

Interfaces and Packages

Inheritance in java with Interfaces – Defining Interfaces - Implementing Interfaces - Extending Interfaces- Creating Packages - CLASSPATH variable - Access protection - Importing Packages - Interfaces in a Package.

Exception Handling in Java

try-catch-finally mechanism - throw statement - throws statement - Built-in-Exceptions – Custom Exceptions.

Lab Exercises:

4. Implement the static keyword – static variable, static block, static function and static class
5. Implement String and String Buffer classes.
6. Implement this keyword and command line arguments.

Unit – 3 **Teaching Hours: 21**

Multithreading, Generics and The Collections Framework

Multithreading

Java Thread Model - Life cycle of a Thread - Java Thread Priorities - Runnable interface and Thread Class- Thread Synchronization – Inter Thread Communication.

Generics

Generics Concept - General Form of a Generic Class – Bounded Types – Generic Class Hierarchy - Generic Interfaces – Restrictions in Generics

The Collections Framework

The Collections Overview – Collection Interface – List Interface – Set Interface – SortedSet Interface – Queue Interface - ArrayList Class – LinkedList Class – HashSet Class – Using an Iterator – The For Each Statement

Lab Exercises:

7. Implement the concept of inheritance, super, abstract and final keywords
8. Implement package and interface
9. Implement Exception Handing in java

Unit – 4 **Teaching Hours: 21**

Introducing GUI Programing with Swing, Event Handling and Database Programming

Introducing GUI Programing with Swing

Swing Basics – Components and Containers – JLabel and ImageIcons- JTextField – Swing Buttons – JTabbedPane – JScrollPane – JList – JComboBox – JTable – Swing Menus

Event Handling

Delegation Event Model - Event Classes – Key Event Class – Event Listener Interface - Adapter Classes

Database Programming

Connecting to and querying a database – Automatic driver recovery- Connecting to the database - Creating a Statement for executing query - Executing a query - Processing a Query's ResultSet – PreparedStatements.

Lab Exercises:

10. Implement multithreading – Thread class, Runnable interface, thread synchronization and thread communication.
11. Implement generic class and interface
12. Implement collection Interfaces and classes

Unit – 5 **Teaching Hours: 21**

Java Servlets

Servlets Basics – Life Cycle of a Servlet –A Simple Servlet,The Servlet API – Servlet Interfaces – Generic Servlet Class- HttpServletRequest Interface – HttpServeltResponse Interface – HttpServet Class – The Cookie Class – Handling HTTP GET Request – Handling HTTP POST Request

Lab Exercises:

13. Implement Swing components and containers with event handling.
14. Implement basic CRUD operations in JDBC
15. Implement Java Servlets

Essential Reading:

[1] Schildt Herbert, Java: The Complete Reference, Tata McGraw-Hill, 10th Edition, 2017.

Recommended Reading:

[1] Paul Deitel, Java How to Program, Pearson Education Asia, 11th Edition, 2017.
[2] Cay S Horstmann, Core Java Volume 1 Fundamentals, Prentice Hall, 11th Edition, 2018.

Web Resources:

[1] www.w3cschools.com
[2] https://www.javatpoint.com/
[3] http://stackoverflow.com/

MCA341A: INTRODUCTION TO DATA ANALYTICS

Total Teaching Hours for Trimester: 60
Max Marks: 100 **Credits: 4**

Course Objectives

Introduction to Data Analytics course delivers the basics of analytics concepts and various techniques to discover new and hidden knowledge from the data set. The course also covers the concepts of data mining algorithms that play a major part in the CRISP model. This course provides insight into the complete research process in phases as research methodology, data exploration, modeling, evaluation and visualization. R programming, Python programming, MATLAB and Excel are the suggestive tools for implementation.

Course Outcomes

CO1: Understand the fundamental techniques in data analytics

CO2: Perform an exploratory data analysis

CO3: Apply suitable supervised and unsupervised algorithms to real world problems

CO4: Interpret the results of developed models using different visualization techniques

Unit-1 **Teaching Hours: 12**

DATA, RELATIONS AND PREPROCESSING

Introduction; Data and Relations - Scales, relations and measures; Data preprocessing - Errors, Filtering, Data Transformation and Integration, Data Reduction.

Additional Reading: Probability Distributions & Inferential Statistics

Unit-2 **Teaching Hours: 12**

CORRELATION AND REGRESSION

Correlation - Linear Correlation, Correlation and Causality, Chi-Square Test; Regression - Linear Regression, Robust Regression, Neural Networks, Radial Basis Function Networks, Cross Validation, Feature Selection.

Additional Reading: Least Square Problems and Optimization

Unit-3 **Teaching Hours: 12**

FORECASTING AND CLASSIFICATION

Forecasting - Finite StateMachines, Recurrent Models, Autoregressive Models. Classification - Classificaiton Criteria - Naive Bayes Classifier - Linear DiscriminantAnalysis - Support Vector Machine - Nearest Neighbor Classifier - Learning Vector Quantization -Decision Trees.

Additional Reading: Stochastic and Kernel Methods

Unit-4 **Teaching Hours: 12**

CLUSTERING

Clustering - Clustering Partitions - Sequential Clustering - Prototype-based Clustering - Fuzzy Clustering -Relational Clustering - Cluster Tendency Assessment - Cluster Validity - Self-Organizing Map.

Additional Reading: Mining Frequent Patterns

Unit-5 **Teaching Hours: 12**

VISUALISATION AND CASE STUDY

Visualization - Visualizing Amounts, Distributions, Proportions, x-y relationships, Geospatial Data, Uncertainty.

Example Caselets: Dr Hans Gosling - Visualizing Global Public Health

Case Study Topics; Text Analytics; Image Analytics, Business Analytics

Additional Reading: Open Source solutions from Kaggle, GitHub resources and Popular Research Labs

Essential Reading:

[1] Runkler, Thomas. A, Data Analytics: Models and Algorithms for Intelligent Data Analysis, Springer Vieweg, 2012.

[2] Han, Jiawei; Kamber, Micheline and Pie, Jian, Data Mining Concept and Techniques, Morgan and Kaufmann Publisher, Third Edition, 2012.

[3] Wilke, Claus O., Fundamentals of Data Visualization A Primer on Making Informative and Compelling Figures, O'Reilly, 2019.

Recommended Reading:

[1] Michael Berthhold, David J. Hand, Intelligent Data Analysis - An Introduction, Springer Publications, 2nd Edition, 2002.

[2] Leskovec, Jure; Rajaraman, Anand; Ullman, Jeffrey D., Mining of Massive Datasets, Cambridge University Press, 2014.

MCA 341B: INTRODUCTION TO ARTIFICIAL INTELLIGENCE

Total Teaching Hours for Trimester: 60
Max Marks: 100 **Credits: 4**

Course Objectives
This course aims at developing an understanding about the issues involved in defining and simulating perception, identifying the problems where AI is required and the different methods available, to compare and contrast different AI techniques available, to define and explain learning algorithms and to provide the student additional experience in the analysis and evaluation of complicated systems.

Course Outcomes
CO1: Express the modern view of AI and its foundation
CO2: Illustrate Search Strategies with algorithms and Problems.
CO3: Implement Proportional logic and apply inference rules.
CO4: Apply suitable techniques for NLP and Game Playing.

Unit-1 **Teaching Hours: 12**
INTRODUCTION
Introduction to AI, The Foundations of AI, AI Technique -Tic-Tac-Toe. Problem characteristics, Production system characteristics, Production systems: 8-puzzle problem. Intelligent Agents: Agents and Environments, Good Behavior: The concept of rationality – The nature of Environments, The Structure of Agents

Unit-2 **Teaching Hours: 12**
LOCAL SEARCH ALGORITHM
Searching: Uninformed search strategies – Breadth first search, depth first search. Generate and Test, Hill climbing, simulated annealing search, Constraint satisfaction problems, Greedy best first search, A* search, AO* search.

Unit-3 **Teaching Hours: 12**
KNOWLEDGE REPRESENTATION
Propositional logic - syntax & semantics - First order logic. Inference in first order logic, propositional Vs. first order inference, unification & lifts, Clausal form conversion, Forward chaining, Backward chaining, Resolution.

Unit-4 **Teaching Hours: 12**
GAME PLAYING AND PLANNING
Overview, Minimax algorithm, Alpha-Beta pruning, Additional Refinements. Classical planning problem, STRIPS- basic process and working of system – Planning and Acting in the Real World

Unit-5 **Teaching Hours: 12**
Natural Language Processing

Introduction, Syntax processing, Semantic Analysis, Pragmatic and DisCourse Description: Analysis - Perception.

Essential Reading:

[1] E. Rich and K. Knight, Artificial Intelligence, 3rd Edition, New york: TMH, 2019.
[2] S. Russell and P. Norvig, Artificial Intelligence A Modern Approach, 3rd Edition, Pearson Education, 2019.

Recommended Reading:

[1] Eugene Charniak and Drew McDermott, Introduction to Artificial Intelligence, 2nd Edition. Singapore: Pearson Education, 2005.
[2] George F Luger, Artificial Intelligence Structures and Strategies for Complex Problem Solving, 4th Edition. Singapore: Pearson Education, 2008, ISBN-13 9780321545893.
[3] N.L. Nilsson, Artificial Intelligence: A New Synthesis, 1st Edition, USA: Morgan Kaufmann, 2000.

Web Resources:

[1] https://ai.google/education/
[2] https://intellipaat.com/blog/tutorial/artificial-intelligence-tutorial/
[3] https://www.javatpoint.com/artificial-intelligence-tutorial

MCA341C-INTRODUCTION TO INTERNET OF THINGS

Total Teaching Hours for Trimester: 60
Max Marks: 100 **Credits: 4**

Course Objectives
In the current era, billions of devices are Internet-connected and Internet of Things (IoT) standards and protocols are stabilizing, and hence technical professionals must increasingly solve real problems with IoT technologies. To latch on to the varied applications in the field of IoT, this course offers an introduction to the underlying concepts of IoT, the basic architecture, challenges, use cases and the ways of connecting smart objects.

Course Outcomes
CO1: Understand the components and characteristics of IoT and enabling technologies
CO2: Gain knowledge on applications and various challenges in IoT
CO3: Develop an understanding of sensor network and its fundamental communication protocols

Unit-1 **Teaching Hours: 12**
INTRODUCTION TO IoT
Genesis of IoT - IoT and Digitization -IoT Impact -Convergence of IT and OT -IoT Challenges- Security Priorities: Integrity, Availability, and Confidentiality,Introduction to measure the physical quantities, IoT Enabling Technologies - Wireless Sensor Networks, Cloud Computing Big Data Analytics, Communication Protocols- Embedded System- IoT Levels and Deployment Templates.
IoT Network Architecture and Design- Drivers Behind New Network Architectures- Constrained Devices and Networks- Comparing IoT Architectures: The M2M IoT Standardized Architecture- The IoT World Forum (IoTWF) Standardized Architecture- A Simplified IoT Architecture - The Core IoT Functional Stack- IoT Data Management and Compute Stack

Unit-2 **Teaching Hours: 12**
SMART OBJECTS
The "Things" in IoT: Sensors, Actuators, and Smart Objects- Micro-Electro-Mechanical Systems (MEMS), Sensor Networks, Wireless Sensor Networks (WSNs), Communication Protocols for Wireless Sensor Networks,Introduction to Smart Systems using IoT - IoT Design Methodology- Intoduction to IoT Boards (Rasberry Pi, Arduino) and IDE
Case Study: Power Utility Industry , Smart and Connected Cities, Transportation, Mining, Public safety, Weather Monitoring

Unit-3 **Teaching Hours: 12**
CONNECTING SMART OBJECTS
Communications Criteria Range - Frequency Bands - Power Consumption - Topology - Constrained Devices - Constrained-Node Networks- Data Rate and Throughput -Latency and Determinism- Overhead and Payload
IoT Access Technologies - IEEE 802.15.4 -Standardization and Alliances -Physical Layer- MAC Layer -Topology - Security -Competitive Technologies

Unit-4 **Teaching Hours: 12**

IP AS THE IOT NETWORK LAYER

The Key Advantages of Internet Protocol - Adoption or Adaptation of the Internet Protocol - The Need for Optimization - Constrained Nodes - Constrained Networks - IP Versions - Optimizing IP for IoT - From 6LoWPAN to 6Lo- RPL- Authentication and Encryption on Constrained Nodes- Internet Protocol for Smart Objects (IPSO) Alliance

Unit-5 **Teaching Hours: 12**

APPLICATION PROTOCOLS FOR IOT

The Transport Layer - IoT Application Transport Methods -Application Layer Protocol SCADA - A Little Background on SCADA- Adapting SCADA for IP- Tunneling Legacy SCADA over IP Networks -SCADA Protocol Translation-SCADA Transport over LLNs with MAP-T - Generic Web-Based Protocols - IoT Application Layer Protocols - CoAP - Message Queuing Telemetry Transport (MQTT)

Essential Reading:

[1] David Hanes, Gonzalo Salgueiro, IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, Cisco Press, 2017.
[2] Arshdeep Bahga and Vijay Madisetti, Internet of Things: Hands-on approach, Hyderabad University Press, 2015.
[3] KazemSohraby, Daniel Minoli and TaiebZnati, Wireless Sensor Networks: Technology. Protocols and Application, Wiley Publications, 2010.
[4] Waltenegus Dargie and Christian Poellabauer, Fundamentals of Wireless Sensor Networks: Theory and Practice, A John Wiley and Sons Ltd., 2010.

Recommended Reading:

[1] Edgar Callaway, Wireless Sensor Networks: Architecture and Protocols, Auerbach Publications, 2003.
[2] Michael Miller, The Internet of Things, Pearson Education, 2015.
[3] Holger Karl and Andreas Willig, Protocols and Architectures for Wireless Sensor Networks, John Wiley & Sons Inc., 2005.
[4] Erdal Çayirci and Chunming Rong, Security in Wireless Ad Hoc and Sensor Networks, John Wiley and Sons, 2009.
[5] Carlos De Morais Cordeiro and Dharma Prakash Agrawal, Ad Hoc and Sensor Networks: Theory and Applications, World Scientific Publishing, 2011.
[6] Waltenegus Dargie and Christian Poellabauer, Fundamentals of Wireless Sensor Networks Theory and Practice, John Wiley and Sons, 2010
[7] Adrian Perrig and J. D. Tygar, Secure Broadcast Communication: In Wired and Wireless Networks, Springer, 2006.