VIGNAN’S INSTITUTE OF INFORMATION TECHNOLOGY (A)

Duvvada, Visakhapatnam-530049

P.G. Course

MASTER OF COMPUTER APPLICATIONS (MCA)

(Two years course)

PROGRAM STRUCTURE& I MCA SYLLABUS

VR20

(Applicable for batches admitted from 2019-20)

MCA PROGRAM STRUCTURE (VR20)

MCA 1st Year (I-Semester)

S. NO.	Subject Code	Subject Name	Theory	Tutorial	Practical	Credits
1	4098201100	Computer Organization	3	0	0	3
2	4098201101	Discrete Mathematical Structures & Graph Theory	3	0	0	3
3	4098201102	Statistical Programming with R	3	0	0	3
4	4098201120	C Programming & Data Structures	3	0	3	4.5
5	4098201121	OOPS Through Java	3	0	3	4.5
6	4098201122	Database Management Systems	3	0	3	4.5
		Total	**18**	**0**	**9**	**22.5**

MCA 1st Year (II-Semester)

S. NO.	Subject Code	Subject Name	Theory	Tutorial	Practical	Credits
1	4098201200	Software Engineering	3	0	0	3
2	4098201201	Data Warehousing and Data Mining	3	0	0	3
3	4098201202	Computer Networks	3	0	0	3
4	4098201220	Python Programming	3	0	3	4.5
5	4098201221	Advanced Java & Web Technologies	3	0	3	4.5
6	4098201222	UNIX Programming	3	0	3	4.5
7	4098201270	Mini Project	0	0	0	2
	4098201280	Seminars				
		Total	**18**	**0**	**9**	**24.5**

MCA 2nd Year (III-Semester)

S. NO.	Subject Code	Subject Name	Theory	Tutorial	Practical	Credits
1	4098202120	Internet of Things	3	0	3	4.5
2	4098202121	Advanced Data Structures	3	0	3	4.5
3	4098202100	Machine Learning	3	0	0	3
4	4098202101	Cryptography & Network Security	3	0	0	3
5 (Professional Elective-1)	4098202150	Object Oriented Analysis and Design Through UML	3	0	0	3
	4098202151	Artificial Intelligence	3	0	0	
	4098202152	Big Data Analytics	3	0	0	
	4098202153	Web Designing using PHP	3	0	0	
6 (Professional Elective-2)	4098202154	Data Science using Python	3	0	0	3
	4098202155	Principles and Practices of Management	3	0	0	
	4098202156	Cloud Computing	3	0	0	
	4098202157	Full Stack Development (MERN)	3	0	0	
		Total	**18**	**0**	**6**	**21**

MCA 3rd Year (IV-Semester)

S. No.	Subject Code	Subject Name	Theory	Tutorial	Practical	Credits
1	4098202280	Technical Seminar	0	0	0	2
2	4098202290	Internship (or)	0	0	0	10
	4098202270	Major Project				
						Total Credits : **12**
						Grand Total Credits:**80**

Total Credits = 22.5+24.5+21+12=80

MCA Syllabus (VR20)

(I MCA I Semester Syllabus)

CourseCode	COMPUTER ORGANIZATION	L	T	P	C
4098201100		3	0	0	3

Course Description and Objectives:

The course aims to teach a student the fundamental components used in a Digital Computer and its functioning.

Course Outcomes:

	Course outcome	Skill	PO
CO1	Identify the logic gates and their functionality and performing conversions from one system to another system	Remember/Understand /Apply	PO1,PO2,PO3
CO2	Design basic electronic circuits (combinational circuits).	Apply	PO2, PO3,PO5
CO3	The basic components of a computer, including CPU, memories, and input/output and their organization representation of data, addressing modes, instruction sets.	Apply/Understand	PO2, PO3,PO5
CO4	Extrapolate memory organization and input, output organizations	Create	PO2,PO3,PO4,PO5,PO12

Syllabus

UNIT – 1

NUMBER SYSTEM AND CODES: Decimal Numbers, Binary Numbers, Decimal to binary Conversions, Binary Arithmetic, 1’s and 2’s complements of Binary Numbers, Signed Numbers, Arithmetic Operations with Signed numbers, Hexadecimal Numbers, Octal Numbers.

LOGIC GATES: The Inverter, The AND gate, The OR gate, The NAND gate, NOR gate, The Exclusive–OR gate and Exclusive-NOR gate; Boolean Algebra and Logic Simplification - Boolean Operations and Expressions, Laws and Rules, DeMorgan’s Theorems, Boolean Expressions and Truth tables. **[15 Hours]**

UNIT – 2

COMBINATIONAL LOGIC ANALYSIS: Basic combinational Logic Circuits, Implementing Combinational Logic, The Universal Property of NAND and NOR Gates. Functions of Combinational Logic - Basic Adder, Parallel Binary Adders, Comparators, Decoders, Encoders.

Basic Structure Of Computers: Functional unit, Basic Operational concepts, Bus structures, System Software, Performance, The history of computer development. **[15 Hours]**

UNIT – 3

Machine Instruction and Programs: Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types, Addressing Modes, Basic Input/output Operations, The role of Stacks and Queues in computer programming equation.

Type of Instructions: Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations **[12 Hours]**

UNIT – 4

INPUT/OUTPUT ORGANIZATION: Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus. **[10 Hours]**

UNIT - 5

MEMORY AND STORAGE: Memory Basics, The RAM, The ROM, Programmable ROMs, The Flash Memory, Memory Expansion, Special Types of Memories, Magnetic and Optical Storage. **[10 Hours]**

Text Book:

1. Floyd, Thomas L, "Digital Computer Fundamentals", 10th Edition, University Book Stall, 1997.
2. Computer Organization, Carl Hamacher, ZvonksVranesic, SafeaZaky, 5th Edition, McGraw Hill.
3. Computer Architecture and Organization , John P. Hayes ,3rd Edition, McGraw Hill.

Reference Books:

1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI

2. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson

3. Fundamentals or Computer Organization and Design, - SivaraamaDandamudi Springer Int.

Edition.

CourseCode	Discrete Mathematical Structures &Graphy Theory	L	T	P	C
4098201101		3	0	0	3

Course Description and Objectives:

The purpose of this course is to provide the students with solid foundations in the basic concepts of counting techniques, mathematical logic, predicates, graph theory concepts, and algorithms.

The main objective of the course is to teach the students how to implement the concept and how to design the given data and algorithms that are appropriate for problems.

Course Outcomes:

	Course outcome	Skill	PO
CO1	Understanding different counting Techniques.	understanding	PO1,PO4
CO2	Apply different methods to solve homogeneous and non-homogeneous recurrence relations	Applying	PO1,PO2,PO3
CO3	Apply graph theory concepts in core subjects such as data structures and network theory effectively	Applying	PO1,PO2,PO9
CO4	Analyse logical structure and able to apply inference theory to verify the consistence of data	Applying	PO1,PO2

Syllabus

UNIT – 1:

Mathematical Logic:Statements and notations, Connectives, Well-formed formulas, Truth tables, tautology,equivalence implication, Normal forms, Theory of inference for the statement calculus, Rules ofinference, Consistency of premises and indirect method of proof, Predicate calculus: Predicates,statement functions, variables and quantifiers, predicate formulas, free & bound variables,universe of discourse, inference theory of predicate calculus.

UNIT-II:

Relations: Introduction,

Relations and ordering, Properties of binary Relations, Equivalence,Compatibility Relations, Partial ordering, Hasse diagram, Lattice and its Properties.

Functions: composition of functions, Inverse Function, Recursive Functions.

UNIT-III:

Elementary Combinatorics:Basis of counting, Enumeration of Combinations &Permutations, Enumerating of Combinations& Permutations with repetitions and constrained repetitions, Binomial Coefficients, BinomialMultinomial theorems, principles of Inclusion –Exclusion.

UNIT-IV:

Recurrence Relations:Generating Function of Sequences, Calculating Coefficient of generating functions, RecurrenceRelations, solving recurrence relation by substitution and Generating functions, the method ofCharacteristic roots, Solution of Inhomogeneous Recurrence Relation.

UNIT-V:

Graph Theory: Representation of Graph, Spanning Trees, BFS, DFS, Kruskals Algorithm, Binary trees, PlanarGraphs, Graph Theory and Applications, Basic Concepts, Isomorphism and Sub graphs, MultiGraphs and Euler circuits, Hamiltonian graphs.

Textbooks:

1. Discrete Mathematical Structures with Applications to computer science J.PTrembley, R.Manohar, TMH

2. Discrete Mathematical for computer Scientists & Mathematicians “ J.L. Molt, A.Kandel,

3. T.P.Baker, PHI.

Reference textbooks:

1. Elements of Discrete Mathematics, C L Liu, D P Mohanpatra,TMH

2. Discrete Mathematics, Schaum’s Outlines, Lipschutz, Lipson, TMH.Discrete

3. Mathematical Structures, Kolman, Busby, Ross, 6thed., PHI, 2009

4. Discrete Mathematics, Johnsonbaugh, 6th ed., Pearson, 2005

5. Discrete Mathematics, Malik, Sen, 6th ed., Cengage Learning, 2004

6. Discrete Mathematics for computer science, Bogart, Stein and Drysdale,

CourseCode	STATISTICAL PROGRAMMING WITH R	L	T	P	C
4098201102		3	0	0	3

Course Description and Objectives:

- Developing programs in R Language
- Students will learn control structures, graphics functions in R
- Simplifying statistical concepts in R using graphs
- Solving statistical problems using probability concepts

Course Outcomes:

	Course outcome	Skill	PO
CO1	Understand the basics of R programming	Understand	PO1,PO2
CO2	Knowledge on R programming control statements	Understand	PO1,PO2,PO3
CO3	Knowledge on Graphics	Understand	PO3,PO5
CO4	Awareness on statistical concepts	Understand/Apply	PO3,PO4,PO6,PO12

SYLLABUS

UNIT-I: Introduction, How to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays,

Classes**.**

UNIT-II: R Programming Structures, Control Statements, Loops, - Looping Over Nonvector Sets,- If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values, Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective, No Pointers in R, Recursion, A Quick sort Implementation- Extended Example: A Binary Search Tree.

UNIT-III: Doing Math and Simulation in R, Math Function, Extended Example Calculating Probability- Cumulative Sums and Products-Minima and Maxima- Calculus, Functions for Statistical Distribution, Sorting, Linear Algebra Operation on Vectors and Matrices, Extended Example: Vector cross Product- Extended Example: Finding Stationary Distribution of Markov Chains, Set Operation, Input /output, Accessing the Keyboard and Monitor, Reading and writer Files,

UNIT-IV: Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot () Function –Customizing Graphs, Saving Graphs to Files.

UNIT-V: Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions Other Distribution, Basic Statistics, Correlation and Covariance, T-Tests,-ANOVA. Linear Models, Simple Linear Regression, -Multiple Regression Generalized Linear Models, Logistic Regression, - Poisson Regression- other Generalized Linear Models-Survival Analysis, Nonlinear Models, Spines- Decision- Random Forests,

Text Books:

1) The Art of R Programming, Norman Matloff, Cengage Learning

2) R for Everyone, Lander, Pearson

Reference Books:

1) R Cookbook, PaulTeetor, Oreilly.

2) R in Action,RobKabacoff, Manni

COURSECODE	C PROGRAMMING AND DATA STRUCTURES	L	T	P	C
4098201120		3	0	3	4.5

Course Overview and Objectives:

- Students will learn how to analyze the problem and writing a code for given algorithm
- Students will learn different operators, control structures and familiarize with the large number of built-in functions available in C language
- Students will learn the various data structure concepts like searching, sorting, stack, queue etc.

Course Outcomes:

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO number mapped**
CO1	Analyze problems and develop solutions by writing algorithms.	Remember/Understand	PO1,PO2
CO2	Design various dynamic allocation memory programs.	Understand & Apply	PO1,PO2,PO3
CO3	Develop simple real-time applications to get familiarity of the programming environment.	Apply	PO2, PO3,PO5
CO4	Solve problems using various data structures like linear list, stack, queue, trees.	Create	PO2,PO3,PO4,PO5,PO12

SYLLABUS

UNIT-1: COMPUTER BASICS: Introduction to Computers, HW and SW Concepts, Problem Solving Techniques, Program Development Steps, **C BASICS**: Basic structure of C Program, History of C, identifiers, basic data types and sizes, constants, variables, different types of operators, type conversions, expressions, precedence and order of evaluation. **CONTROL STRUCTURES:** if statement- switch and go to statement, labels and looping statements.

UNIT-II: ARRAYS: Basic concepts of arrays- declaration, definition, accessing elements, storing elements, 1-D arrays, 2-D arrays, **STRINGS:** String concepts, String manipulations, character arrays, multi-dimensional arrays, and string handling functions. **FUNCTIONS:** Basic concepts on functions, types of functions, categories of functions, Function parameters, Parameter passing techniques, scope rules, block structure, Storage classes, Recursion concept, recursive functions.

UNIT-III: POINTERS: Definition, declaration and initialization of pointer variables, the & and * operators, void pointer, pointer-to-pointer, pointers and function arguments, passing by address, character pointer and functions, pointer and multidimensional arrays, dynamic memory management functions, array of pointer, command line arguments.

STRUCTURES and UNIONS: Definition, initialization, accessing structure elements, array of structures and array within the structure, nested structures, self-referential structure, Union's concept, differences between structures and unions, type-def, bit fields, enum data type.

FILES: Concepts of a files, text and binary files, opening and closing a file, file I/O operations, formatted I/O operations.

UNIT-IV : DATA STRUCTURES: Introduction to Data Structures, Time complexity, Space complexity, Pattern matching, naïve method, Robin Karp Algorithm-Searching-Linear and Binary Search methods, Sorting- Bubble sort, selection sort, insertion sort, quick sort, merge sort.

STACKS and QUEUES: Stack concept, applications of stack-infix to postfix conversion, postfix expression evaluation, implementation of stack using array- Queue concept, types of queues, applications of queues, queue implementation using array.

UNIT-V: LINKED LISTS: Singled linked lists, doubly linked lists, circular linked list, linked stack and linked queue. **TREES:** Basic concepts of trees, binary trees, terminology, representation, traversals.

Text Books:

1. C and Data Structures: A snapshot oriented treatise using live engineering examples, N B Venkateswarlu, E. V Prasad, S Chand & Co.
2. Computer science, A structured programming approach using C, B.A. Forouzan and R. F. Gilberg, Third edition, Thomson.

Reference Books:

1. Fundamentals of Data Structures in C, Horowitz, Sahni, Anderson- Freed, 2nd ed., universities Press, 2008.

2. The C Programming Language, B.W. Kernighan, Dennis M. Ritchie, PHI/ Pearson.
3. Data Structures Using C , A. S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/ Pearson
4. C Programming and Data Structures, E Balaguruswamy, TMH, 2008.

LAB EXPERIMENTS

Exercise l

a) Write a C Program to find the sum of individual digits of a positive integer.

b) A Fibonacci sequence is defined as follow: the first second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C Program to generate the first N terms of the sequence.

C) Write a C Program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

d) Write a C Program which checks a given integer is Fibonacci number or not.

Exercise 2

a) Write a C Program to calculate the following sum:

Sum=a-x^2/2!+x^4/4!-x^6/6!+x^8/8!-x^{10}/10!

b) Write a C program to find the roots of a quadratic equation.

c) Write a C Program to implement Newton Raphson method for a quadratic equation

Exercise 3

a) Write a C Program that use both recursive and non-recursive functions

(i) To find the factorial of a given number

(ii) To find the GCD (greatest common divisor) of two given integers.

(iii)To Solve Towers of Hanoi problem

(iv)To generate Fibonacci sequence

Exercise 4

a) The total distance travelled by vehicle in ‘t’ seconds is given by distance = ut+1/2at^2

where ‘u’ and ‘a’ are the initial velocity (m/sec.) and acceleration (m/sec2). Write a C Program to find the distance travelled at regular intervals of time given the values of ‘u’ and ‘a’. The program should provide the flexibility to the user to select his own time intervals and repeat the calculations values of ‘u’ and ‘a’.

b) Write a C program, which takes two integer operands and one operator form the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement)

Exercise 5

a) Write a C program to find both the largest and smallest number in a list of integers.

b) Write a C program that uses functions to perform the following:

i) Addition of Two Matrices.

ii) Multiplication to Two Matrices.

iii) Checking symmetricity of a square matrix

iv) Calculating transpose of a matrix in-plane manner.

Exercise 6

a) Write a C program to determine if the given string is a palindrome or not.

b) Write a C program to count the lines, words and characters in a given text.

c) Write a C program to swap two numbers using pointers

d) Write a C program to find the reverse of a given string using pointers.

Exercise 7

Write a C program that uses functions to perform the following operations using Structure:

i) Reading a complex number	ii) Writing a complex number
iii) Addition of two complex numbers	iv) Multiplication of two complex numbers

Exercise 8

a) Write a C program which copies one file to another.

b) Write a C program to reverse the first n characters in a file using command line arguments

Exercise 9

a) Write C program that use both recursive and non-recursive functions to perform Linear search for a Key value in a given list.

b) Write C program that use both recursive and non-recursive functions to perform Binary search for a Key value in a given list.

c) Write C program that use both recursive and non-recursive functions to perform Fibonacci search for a Key value in a given list.

Exercise 10

a) Write C program that implement Bubble sort, to sort a given list of integers in ascending order

b) Write C program that implement Quick sort, to sort a given list of integers in ascending order

c) Write C program that implement Insertion sort.

Exercise 11

a) Write C program that implement stack (its operations) using arrays

b) Write C program that implement stack (its operations) using Linked list

Exercise 12

a) Write a C program that uses Stack operations to convert infix expression into postfix expression

a) Write C program that implement Queue (its operations) using arrays.

b) Write C program that implement Queue (its operations) using linked lists

Exercise 13

a) Write a C program that uses functions to create a singly linked list

b) Write a C program that uses functions to perform insertion operation on a singly linked list

c) Write a C program that uses functions to perform deletion operation on a singly linked list

Exercise 14

a) Adding two large integers which are represented in linked list fashion.

b) Write a C program to reverse elements of a single linked list.

c) Write a C program to store a polynomial expression in memory using linked list

d) Write a C program to representation the given sparse matrix using arrays.

e) Write a C program to representation the given sparse matrix using linked list

Exercise 15

a) Write a C program to create a Binary Tree of integers

b) Write a recursive C program for traversing a binary tree in preorder, inorder and postorder.

c) Write a non-recursive C program for traversing a binary tree in preorder, inorderandpostorder.

Exercise 16

a) Write a C program to create a BST

b) Write a C program to insert a node into a BST.

c) Write a C program to delete a node from a BST.

CourseCode	OOPS Through Java	L	T	P	C
4098201121		3	0	3	4.5

Course Description and Objectives:

This course aims at applying the concepts of Object-Oriented programming in Java, its importance in finding solutions for specific problems. The properties of Object-Oriented systems can be experienced by designing Object-Oriented applications

Course Outcomes:

	Course outcome	Skill	PO
CO1	Apply OOP concepts and basics of Java programming.	Remember/Understand	PO1,PO2
CO2	Use the concepts of Java programming in problem solving.	Remember/Understand	PO1,PO2,PO3
CO3	Extend the knowledge of Java programming in developing futuristic applications.	Apply	PO2, PO6,PO12

Syllabus

UNIT-I:

JAVA CONCEPTS: Creation of Java, Byte code, Java buzzwords, OOP Principles-Encapsulation, Inheritance and Polymorphism, Compiling and running of simple Java program, Data types, Variables, declaring variables, Dynamic initialization, Scope and life time of variables, Arrays, Operators, Control statements.

UNIT-II:

CLASSES AND OBJECTS: Class fundamentals, Declaring objects, Assigning object reference variables, Introducing methods, Constructors, the 'this' keyword, Garbage collection, 'finalize' method, Overloading methods and Constructors, Call by value, Recursion.

UNIT-III:

INHERITANCE AND PACKAGES: Access control, Usage of 'static' with data and methods, Usage of 'final' with data, exploring the String class, using command line arguments; Basic concepts of inheritance - Member access rules, Usage of super key word, Forms of inheritance, Method overriding, Abstract classes, Dynamic method dispatch, Using final with inheritance, the Object class; Defining, Creating and Accessing a Package, Understanding CLASSPATH, Importing packages.

UNIT-IV:

INTERFACES AND EXCEPTION HANDLING: Defining an interface, Implementing interfaces, Applying interfaces, Variables in Interfaces, Extending Interfaces; Exception handling concepts - Types of exceptions, Usage of try, Catch, Throw, Throws and Finally keywords, Built-in Exceptions, Creating own exception sub-classes.

UNIT-V:

MULTITHREADING: Thread Introduction, Thread class, Runnable Interface, Extending Thread, Creating Multiple Threads, Is Alive(),Join(), Thread priorities, Synchronization. Inter thread communication, Deadlock, Suspending, resuming and stopping Threads.

Text Book:

- Herbert Schildt, “The Complete Reference Java J2SE”, 7th ed., TMH Publishing Company Ltd, New Delhi, 2008.

Reference Books:

- Cay Horstmann, “Big Java”, 2nd ed., John Wiley and Sons, 2006.
- Joe Wiggles worth and Paula McMillan, “Java Programming Advanced Topics”, 3rd ed., TMH, 2009.

LAB EXPERIMENTS

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 (WAJP) that uses both recursive and non recursive functions to print the nth value of the Fibonacci sequence.

2. WAJP to demonstrate wrapper classes and to fix the precision.

3. WAJP that prompts the user for an integer and then prints out all the prime numbers up to that Integer.

4. WAJP that checks whether a given string is a palindrome or not. Ex. MALAYALAM is a palindrome.

5. WAJP for sorting a given list of names in ascending order.

6. WAJP to check the compatibility for multiplication, if compatible multiply two matrices and find its transpose.

7. WAJP that illustrates how runtime polymorphism is achieved.

8. WAJP to create and demonstrate packages.

9. WAJP, using String Tokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.

10. WAJP that reads on file name form the user then displays information about whether the file exists, whether the file is readable/writable, the type of file and the length of the file in bytes and display the content of the using FileInputStream class.

11. WAJP that displays the number of characters, lines and words in a text/text file.

12. Write an Applet that displays the content of a file.

13. WAJP that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +-*?% operations. Add a text field to display the result.

14. WAJP for handling mouse events.

15. WAJP demonstrating the life cycle of a thread.

16. WAJP that correctly implements Producer-Consumer problem using the concept of Inter Thread Communication.

17. WAJP that lets users create Pie charts. Design your own user interface(with Swings & AWT).

18. WAJP that allows user to draw lines, rectangles and ovals.

19. WAJP that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result and then sends the result back to the client. The client displays the result on the console. For ex: The data send form the client is the radius of a circle and the result produced by the server is the area of the circle.

20. WAJP to generate a set of random numbers between two numbers x1 and x2, and x1>0.

21. WAJP to create an abstract class named shape, that contains an empty method named number Of Sides(). Provide three classes named Trapezoid, Triangle and Hexagon, such that each one of the classes contains only the method number Of Sides(), that contains the number of sides in the given geometrical figure.

22. WAJP to implement a Queue, using user defined Exception Handling (also make use of throw, throws).

23. WAJP that creates 3 threads by extending Thread class. First thread displays “Good Morning” every 1 sec, the second thread displays “Hello” every 2 seconds and the third displays “Welcome” every 3 seconds. (Repeat the same by implementing Runnable).

24. Create an inheritance hierarchy of Rodent, Mouse, Gerbil, Hamsteretc. In the base class provide methods that are common to all Rodents and override these in the derived classes to perform different behaviours, depending on the specific type of Rodent. Create an array of Rodent, fill it with different specific types of Rodents and call your base class methods.

CourseCode	Database Management Systems	L	T	P	C
4098201122		3	0	3	4.5

Course Description and Objectives:

This course introduces database design and creation using a DBMS product. Emphasis is on data dictionaries, normalization, data integrity, data modeling, and creation of simple tables, queries, reports, and forms. Upon completion, students should be able to design and implement normalized database structures by creating simple database tables, queries, reports, and forms.

Course Objectives:

1. Provide students with theoretical knowledge and practical skills in the use of database and database management systems in information technology applications.
2. The logical design, physical design and implementation of relational databases are covered.

Course Outcomes:

	Course Outcome	Skill	PO
CO1	Student can able to describe the Architecture of Database Management Systems	Remember/Understand	PO1,PO2
CO2	Student can design different ER Models	Understand	PO2,PO3
CO3	Student can able to differentiate the knowledge in TRC & DRC	Analysis	PO2, PO6,PO12
CO4	Student can differentiate different indexing techniques in real time	Apply	PO5

Syllabus

UNIT-I:Introduction to Database & ER Diagrams

Introduction to Database: Database System Applications, Purpose of Database Systems, View of Data – Data Abstraction, Instances and Schemas, Data Models – the ER Model, Relational Model, Other Models – Database Languages –DDL, DML, Database Access from Applications Programs, Transaction Management, Data Storage and Querying, Database Architecture, Database Users and Administrators, History of Data base Systems. Introduction to Data base design,

ER diagrams: Introduction, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, and Conceptual Design for Large enterprises, Logical Database Design.

UNIT-II:Relational Algebra and Calculus & Relational Model

Relational Algebra and Calculus: Relational Algebra – Selection and Projection, Set operations, Renaming, Joins, Division, Examples of Algebra Queries, Relational calculus – Tuple relational Calculus – Domain relational calculus

Relational Model: Introduction to the Relational Model – Integrity Constraints over Relations, Enforcing Integrity constraints, Introduction to Views – Destroying /altering Tables and Views.

UNIT-III:SQL & Schema Refinement

SQL: Form of Basic SQL Query – Examples of Basic SQL Queries, Introduction to Nested Queries, Correlated Nested Queries, Set – Comparison Operators, Aggregate Operators, NULL values – Comparison using Null values – Logical connectives – AND, OR and NOT – Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Complex Integrity Constraints in SQL Triggers.

Introduction to Schema Refinement – Problems Caused by redundancy, Decompositions – Problem related to decomposition, Functional Dependencies - Reasoning about FDS, Normal Forms – FIRST, SECOND, THIRD Normal forms – BCNF –Properties of Decompositions- Loss less join Decomposition, Dependency preserving Decomposition, Schema Refinement in Data base Design – Multi valued Dependencies – FOURTH Normal Form, Join Dependencies, FIFTH Normal form, Inclusion Dependencies.

UNIT-IV: Transaction Management & Crash Recovery

Overview of Transaction Management: The ACID Properties, Concurrent execution of Transactions. Concurrency Control Techniques: Scheduling, Serializability, Basic concepts on Locking Techniques, Lock Based Concurrency Control, Deadlocks – Performance of Locking, Concurrency Control without Locking.

Crash recovery: Introduction to Crash recovery, Introduction to ARIES.

UNIT-V:Storage & Indexing

Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes, Index data Structures – Hash Based Indexing, Tree based Indexing. Storing data: Disks and Files: -The Memory Hierarchy – Redundant Arrays of Independent disks. Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM) B+ Trees: A Dynamic Index Structure, Search, Insert and Delete. Hash Based Indexing: Static Hashing, Extendable hashing, Linear Hashing, Extendible vs. Linear Hashing.

List of Experiments

1. Execute a single line and group functions for a table.

2. Execute DCL and TCL Commands.

3. Create and manipulate various DB objects for a table.

4. Create views, partitions and locks for a particular DB.

5. Write PL/SQL procedure for an application using exception handling.

6. Write PL/SQL procedure for an application using cursors.

7. Write a DBMS program to prepare reports for an application using functions.

8. Write a PL/SQL block for transaction operations of a typical application using triggers.

9. Write a PL/SQL block for transaction operations of a typical application using package.

10. Design and develop an application using any front end and back end tool (make use of ER diagram and DFD).

11. Create table for various relation

12. Implement the query in sql for a) insertion b) retrieval c) updation d) deletion

13. Creating Views

14. Writing Assertion

15. Writing Triggers

16. Implementing operation on relation using PL/SQL

17. Creating Forms

18. Generating Reports

19. Typical Applications – Banking, Electricity Billing, Library Operation, Pay roll, Insurance, Inventory etc.

Text Books:

1. Database Management and Design by G.W. Hansen and J.V. Hansen, 2nd edition,1999, Prentice-Hall of India, Eastern Economy Edition.

2. Data base System Concepts, A. Silberschatz, H.F. Korth,S. Sudarshan, McGraw hill, VI edition, 2006.

3. Fundamentals of Database Systems 5th edition, RamezElmasri, Shamkant .Navathe, Pearson Education, 2008.

Reference Books:

1. Database Management Systems by A.K. Majumdar and P. Bhattacharyya, 5 th edition, 1999, Tata McGraw-Hill Publishing.

2. A Guide to the SQL Standard, Date, C. and Darwen,H. 3rd edition, Reading, MA: 1994, AddisonWesley

3. Database Systems, A Practical approach to Design Implementation and Management Fourth edition, Thomas Connolly, Carolyn Begg, Pearson education.
4. Database Management System Oracle SQL and PL/SQL, P.K. Das Gupta, PHI.
5. Introduction to Database Management, M. L. Gillenson and others, Wiley Student Edition.
6. Introduction to Database Systems, C. J. Date, Pearson Education.

MCA Syllabus (VR-20)

(I MCA II Semester Syllabus)

COURSECODE	SOFTWARE ENGINEERING	L	T	P	C
4098201200		3	0	0	3

Course Description and Objectives:

This course focuses on the concepts of software life cycle, role of process models and methods to prepare software requirement specification document. In addition to that, it also imparts knowledge of design, development and testing of software. The objective of this course is to enable the student to develop efficient, cost effective, feasible software as per client requirements.

Course Outcomes:

	Course outcome	**Skill**	**PO**
CO1	Prepare a Software Requirement Specification (SRS) document foranysoftwareproject.	Remember/Understand	PO1,PO2
CO2	Identify the importance of system analysis and design in solving complexproblems.	Understand	PO1,PO2,PO3
CO3	Distinguish between object-oriented approach and traditional approach in system analysis and design.	Understand	PO3,PO5
CO4	Analyze various quality assurance and standards along with metrics of software	Understand/Apply	PO3,PO4,PO6,PO12

Syllabus

UNIT-1

The evolving role of software, Changing Nature of Software, Software myths. (Text Book 3) The software problem: Cost, schedule and quality. Process and project, component software process, **Software development process models** : Waterfall model, prototyping, iterative development, relational unified process, time boxing model, Extreme programming and agile process.

UNIT-2

Software requirement analysis and specification

Value of good SRS, requirement process, requirement specification, functional specifications with use-cases, other approaches for analysis, validation.

Planning a software project: Effort estimation, project schedule and staffing, quality planning, risk management planning, project monitoring plan, detailed scheduling.

UNIT-3

Role of software architecture, architecture views, components and connector view, architecture styles for C & C view, documenting architecture design, evaluating architectures.

Design: Design concepts, function-oriented design, object oriented design, detailed design, verification and metrics .

UNIT-4

Coding and Unit testing Programming principles and guidelines, incrementally developing code, managing evolving code, unit testing, code inspection and metrics.

Testing: Testing concepts, testing process, black-box testing, white-box testing and metrics.

UNIT-5

Software Quality Assurance and Standards: The Software Quality challenge, what is Software Quality, Software Quality factors, The components of Software Quality Assurance system, Software Quality Metrics, Costs of Software Quality, Quality Management Standards.

Text Books:

1) A Concise introduction to software engineering (undergraduate topics in computer science), PankajJalote, Springer International Edition.

2) Software Engineering, A Precise approach, PankajJalote, Wiley

3) Software Engineering, 3/e ,& 7e Roger S.Pressman , TMH

4) Software Quality Assurance – From Theory to Implementation, Daniel Galin, Pearson Education, 2009.

Reference Books:

1) Software Engineering, 8/e, Sommerville, Pearson.

2) Software Engineering principles and practice, W S Jawadekar, TMH

3) Software Engineering concepts, R Fairley, TMH

4) Effective Methods for Software Testing, Third edition, William E. Perry, Wiley India, 2009

COURSECODE	DATA WAREHOUSING AND DATA MINING	L	T	P	C
4098201201		3	0	0	3

Course Overview:

This course introduces the basic concepts, principles, methods, implementation techniques, and applications of data mining, with a focus on three major data mining functions: (1) Association rule mining (2) Classification and (3) cluster Analysis. In the first part of the course student will learn why Association rule mining is important, what are the major methods are for computing association rules and analyzing their importance. In classification student will learn basic concepts of classification and methodologies for classification. This includes KNN, Naive Bayes, Decision tree and Neural Network based methods. In clustering students will learn different clustering methods. This includes K-Means, Hierarchical and Density based methods.

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

- Apply proximity measures on various data types
- Discover Association rules among the Items
- Apply classification and clustering methods
- Compare the different classification methods performance

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO number mapped
CO1	Ability to identify, understand and investigate various patterns that can be extracted from different types of data.	Remember/Understand	PO4(2)
CO2	Apply various pre-processing techniques and classification algorithms on different domains of data	Understand & Apply	PO4(2), PO5(3)
CO3	Build decision making systems using data mining algorithms for a given real time data set.	Understand & Apply	PO3(2), PO4(3), PO5(3)
CO4	Construct models using modern tools.	Apply	PO6(4)

UNIT- I:

Introduction to Data Mining: Motivation for Data Mining, Data Mining-Definition &

Functionalities, Classification of DM systems, DM task primitives, Integration of a Data Mining system with a Database or a Data Warehouse, Major issues in Data Mining. **Data Warehousing**

(Overview Only): Overview of concepts like star schema, fact and dimension tables, OLAP operations, From OLAP to Data Mining.

UNIT -II:

Data Preprocessing: Why? Descriptive Data Summarization, Data Cleaning: Missing Values,Noisy Data, Data Integration and Transformation. Data Reduction:-Data Cube Aggregation,Dimensionality reduction, Data Compression, Numerosity Reduction, Data Discretization and Concept hierarchy generation for numerical and categorical data.

UNIT- III:

Mining Frequent Patterns, Associations, and Correlations: Market Basket Analysis, Frequent Itemsets, Closed Itemsets, and Association Rules, Frequent Pattern Mining, Efficient and Scalable Frequent Itemset Mining Methods, The Apriori Algorithm for finding Frequent Itemsets Using Candidate Generation, Generating Association Rules from Frequent Itemsets, Improving the Efficiency of Apriori, Frequent Itemsets without Candidate Generation using FP Tree, Mining Multilevel Association Rules, Mining Multidimensional Association Rules, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.

UNIT- IV:

Classification & Prediction: What is it? Issues regarding Classification and prediction

Classification methods: Decision tree, Bayesian Classification, Rule based Prediction: Linear and non linear regression, Accuracy and Error measures, Evaluating the accuracy of a Classifier or Predictor.

Cluster Analysis: What is it? Types of Data in cluster analysis, Categories of clustering

methods, Partitioning methods iñ K-Means, K-Mediods. Hierarchical Clustering- Agglomerative and Divisive Clustering, BIRCH and ROCK methods, DBSCAN, Outlier Analysis.

UNIT- V:

Data Mining for Business Intelligence Applications: Data mining for business Applications

like Balanced Scorecard, Fraud Detection, Click stream Mining, Market Segmentation, retail industry, telecommunications industry, banking & finance and CRM etc.

OUTCOMES

- Design systems for sourcing and structuring data to provide an integrated, non-volatile collection of data for decision support using data warehouses;
- Design multidimensional data models and implement them using star schemas and relational databases;
- Communicate and foster realistic expectations of the role of OLAP technology and business intelligence systems in management and decision support;
- Explain the need for evolutionary development approaches to developing business intelligence and data warehouse systems;
- Develop a simple business intelligence system using an OLAP tool;
- Apply theories and principles of data visualization to encourage high quality analysis of business information to inform decision making;

TEXT BOOKS:

1. Han, Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann 2 nd Edition

2. P. N. Tan, M. Steinbach, Vipin Kumar, introduction to Data Miming, Pearson Education

REFERENCE BOOKS:

1. MacLennan Jamie, Tang ZhaoHui and CrivatBogdan, Data Mining with Microsoft SQL Server 2008î, Wiley India Edition.

2. G. Shmueli, N.R. Patel, P.C. Bruce, ìDataMining for Business Intelligence: Concepts, Techniques and Applications in Microsoft Office Excel with XLMinerî, Wiley India.

3. Michael Berry and Gordon LinoffìData Mining Techniquesî, 2nd Edition Wiley Publications.

4. Alex Berson and Smith, ìData Mining and Data Warehousing and OLAPî, McGraw Hill Publication.

5. E. G. Mallach, ìDecision Support and Data Warehouse Systems", Tata McGraw Hill.

6. Michael Berry and Gordon LinoffìMastering Data Mining- Art & science of CRMî, Wiley Student Edition

7. Arijay Chaudhry & P. S. Deshpande, ìMultidimensional Data Analysis and Data Mining Dreamtech Press

8. VikramPudi&Radha Krishna, ìDataMiningî, Oxford Higher Education.

9. Chakrabarti, S., ìMining the Web: Discovering knowledge from hypertext dataî,

10. M. Jarke, M. Lenzerini, Y. Vassiliou, P. Vassiliadis (ed.), ìFundamentals of Data Warehousesî, Springer-Verlag, 1999.telligence.ng Data Mining & OLAP, Alex Berson, Stephen Smith, MH

CourseCode	COMPUTER NETWORKS	L	T	P	C
4098201202		3	0	0	3

Course Overview: This course introduces basic routing algorithms and related protocols for Network Management and define the functionality of different type of network layers and it efficient usage.

Course Objectives: Students will learn the basics of networks and its topological design.

Students will learn different type of error detection and control for data link layer protocol and understanding the different type of routing algorithm.

	Course outcome	Skill	PO
CO1	Students will able to understand the network topology and its structure	Understand and Analyze	PO1,PO2
CO2	Students will able to understand different types of network standards and protocols	Understand	PO1,PO2,PO3
CO3	Students will able design new routing technique base on exiting study	Understand and Analyze	PO3,PO5
CO4	Students will able to understand the application of networks in MANETS, Adhoc Networks, Wireless Sensors network etc.	Understand/Apply	PO3,PO4,PO6,PO12

Unit-I: Network reference model

Introduction to Networks, Network H/w, OSI-reference models, Network Standardization, Transmission media, Narrowband ISDN, Broad band ISDN, ATM. **Physical Layer:** Guided transmission media – Magnetic media, Twisted Pair, coaxial cable, fiber optics.

Unit-II:The data Link layer

Design Issues, Error detection and correction, Elementary Data Link Protocols, Sliding window protocols in HDLC.
Channel allocation methods: TDM, FDM, ALOHA, Carrier sense Multiple access protocols, Collision Free protocols – IEEE standard BO2 for LANS – Ethernet, Token Bus, Token ring, Bridges.

Network layer Routing Algorithms: Shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, DSR, AODV, Broadcast routing, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Choke packets and Load shedding.

Unit-III:Internet Working

Tunneling, internetworking, Fragmentation, network layer in the internet – IP protocols, IP address, Subnets, Internet control protocols, DSPF, BOP, Internet multicasting, Mobile IP. Network layer in the ATM Networks – cell formats, connection setup, routing and switching, service categories, and quality of service, ATM LANs.

Unit-IV:Transport Layer

Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, end to end protocols: UDP, reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, TCP extension, Remote Procedure Call Performance Issues.

Unit-V:Application Layer

Network Security. Security Mechanisms: Authentication Protocols, Firewalls, Cryptographic Algorithms: DES, RSA, Digital Signatures, Domain Naming system Hierarchy, Name servers. Traditional Applications: SMTP, MIME, World Wide Web: HTTP, Management of public keys.

Text Books:

1. Computer Networks Andrew, Tanenbaum, 4/e, Pearson

Reference Books:

1. Data communications and networking Forouzan, 4/e, TMH

CourseCode:	PYTHON PROGRAMMING	L	T	P	C
4098201220		3	0	3	4.5

Course Overview:

The course is designed to provide an introduction to the python programming language. The focus of the course is to provide students with an introduction to programming, I/O, functions, visualization using the Python programming language.

Course Objectives:

- Introduction to Scripting Languages
- Understanding the operators, functions, key concepts of OOP in Python.
- Exposure to various problems solving approaches of Computer Applications.

Course Outcomes: After completion of course students will be able to

CO1	Construct Software easily right out of the box
CO2	Experiment with an interpreted Language
CO3	Build software for real needs
CO4	Explain to testing Orielly

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	-	1	1	-	2	-	-	1	-	1
CO2	2	2	-	1	2	-	1	-	1	-	-	-
CO3	1	1	2	3	1	2	1	1	1	1	1	3
CO4	-	1	-	1	-	2	-	1	-	1	1	2

SYLLABUS

Unit-I: Introduction

No. of lecture hours: 9

History of Python, Need of Python Programming, Applications Basics of Python, Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

Unit-II:Types, Operators and Expressions

No. of lecture hours: 9

Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while, break, continue, pass.

Unit-III: Data Structures **No. of lecture hours: 8**

Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences, Comprehensions.

Unit-IV: Functions **No. of lecture hours: 9**

Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful, Functions(Function Returning Values), Scope of the Variables in a Function - Global and Local variables.

Modules: Creating modules, import statement, from. Import statement, name spacing.

Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages.

Unit-V: **No. of lecture hours: 10**

Object Oriented Programming OOP in Python: Classes, 'self-variable', Methods, Constructor, Method, Inheritance, Overriding Methods and Data hiding.

Error and Exceptions: Difference between an error and Exception, Handling Exception, try, except block, Raising Exceptions, User Defined Exceptions

Brief Tour of the Standard Library - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics.

Introduction to Relational databases: An introduction to relational databases, SQL statements for data manipulation, creating a GUI that handles an event, working with components, Illustrative programs.

LAB EXPERIMENTS

Week 1: Exercise 1 - Basics

a) Running instructions in Interactive interpreter and a Python Script

b) Write a program to purposefully raise Indentation Error and Correct it

Week 2: Exercise 2 - Operations

a) Write a program to compute distance between two points taking input from the user (Pythagorean Theorem)
b) Write a program add.py that takes 2 numbers as command line arguments and prints its sum.

Week 3: Exercise - 3 Control Flow

a) Write a Program for checking whether the given number is a even number or not.

b) Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, . . . , 1/10
c) Write a program using a for loop that loops over a sequence. What is sequence?

d) Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

Week 4: Exercise 4 - Control Flow - Continued

a) Find the sum of all the primes below two million.

Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be:

1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

b) By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Week 5: Exercise - 5 - DS

a) Write a program to count the numbers of characters in the string and store them in a dictionary data structure
b) Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

Week 6: Exercise - 6 DS - Continued

a) Write a program combine_lists that combines these lists into a dictionary.
b) Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

Week 7: Exercise – 7 Files

a) Write a program to print each line of a file in reverse order.
b) Write a program to compute the number of characters, words and lines in a file.

Week 8: Exercise - 8 Functions

a) Write a function ball_collide that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.

Hint: Represent a ball on a plan e as a tuple of (x, y, r), r being the radius
If (distance between two balls centers) <= (sum of their radii) then (they are colliding) b) Find mean, median, mode for the given set of numbers in a list.

Week 9: Exercise - 9 Functions - Continued

a) Write a function nearly_equal to test whether two strings are nearly equal. Tw o strings a and b are nearly equal when a can be generated by a single mutation on b.
b) Write a function dups to find all duplicates in the list.

c) Write a function unique to fin d all the unique elements of a list

.

Week 10: Exercise - 10 - Functions - Problem Solving

a) Write a function cumulative product to compute cumulative product of a list of numbers.

b) Write a function reverse to re verse a list. Without using the reverse function.

c) Write function to compute gcd , lcm of two numbers. Each function shouldn't exceed one line.

Week 11: Exercise 11 - Multi-D Lists

Write a program that defines a matrix and prints

a) Write a program to perform addition of two square matrices

b) Write a program to perform multiplication of two square matrices

Week 12: Exercise - 12 - Modules

a) Install packages requests, flask and explore them. using (pip)

b) Write a script that imports requests and fetch content from the page. Eg. (Wiki)

c) Write a simple script that serves a simple HTTP Response and a simple HT ML Page

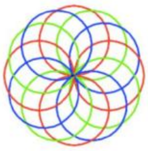
Week 13: Exercise - 13 OOP

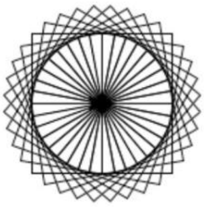
a) Class variables and instance variable and illustration of the self variable

i) Robot
ii) ATM Machine

Week 14: Exercise - 14 GUI, Graphics

1. Write a GUI for an Expression Calculator using tk
2. Write a program to implement the following figures using turtle





Week 15: Exercise - 15 - Testing

a) Write a test-case to check the function even _numbers which return True on passing a list of all even numbers

b) Write a test-case to check the function reverse _string which returns the reversed string

Week 16: Exercise - 16 - Advanced

a) Build any one classical data structure.

b) Write a program to solve knapsack problem.

Text Books:

1. Python Programming: A Modern Approach, VamsiKurama, Pearson

Learning Python, Mark Lutz, Orielly

Reference Books:

1. Think Python, Allen Downey, Green Tea Press

2. Core Python Programming, W. Chun, Pearson.

3. Introduction to Python, Kenneth A. Lambert, Cengage

CourseCode	Advanced Java & Web Technologies	L	T	P	C
4098201221		3	0	3	4.5

Course Description and Objectives:

1. To understand the concepts of HyperText Markup Language and Cascading Style Sheets.
2. To learn JavaScript for creating dynamic websites.
3. To learn the operations perform on data among web applications using XML
4. To acquire knowledge on creation of software components using JAVA Beans.
5. To learn Server-Side Programming using Servlets and Java Server Pages.
6. To learn the creation of pure Dynamic Web Application using JDBC.

Course Outcomes:

	Course outcome	Skill	PO
CO1	Implement web based applications using features of HTML and XML	Understand	PO1,PO2
CO2	Develop reusable component for Graphical User Interface applications	Understand	PO1,PO2,PO3
CO3	Apply the concepts of server side technologies for dynamic web applications.	Analysis	PO3,PO8
CO4	Implement the web based applications using effective data base access with rich client interaction.	Understand/Apply	PO2,PO3,PO4,PO6,

Syllabus

UNIT - I:

Basic HTML Tags, Java Swing package - use of System class - Applet Context,object serialization - shallow and deep copying - Java collections - Iterators - Array Lists - sets - hash set - hash table, vector class.

UNIT - II:

Java Beans: Introduction to Java Beans, Advantages of Java Beans, BDK Introspection, Using Bound properties, Bean Info Interface, Constrained properties Persistence, Customizers, Java Beans API.

UNIT - III:

Introduction to Servlets: Lifecycle of a Servlet, Servlet API, The javax.servlet Package, Reading Servlet parameters, Reading Initialization parameters. The javax.servlet.HTTP Package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues.

UNIT - IV:

Introduction to JSP: The Problem with Servlet. The Anatomy of a JSP Page, JSP Processing. JSP Application Design with MVC. Installing Tomcat Server & Testing Tomcat.

JSP Application Development: Generating Dynamic Content, Using Scripting Elements Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Error Handling and Debugging, Passing Control and Data between JSPPages – Sharing Session and Application Data – Memory Usage Considerations.

UNIT - V:

Database Access: Role of JDBC,JDBCAPI, Types of JDBC Drivers, Database Programming using JDBC, Accessing MySQL database- Accessing MS Access database- Accessing a Database from a JSP Page Application, Deploying JAVA Beans in a JSP Page. Introduction to struts framework.

Lab Experiments

1.Write a program to prompt the user for a hostname and then looks up the IP address for the hostname and displays the results.

2. Design the following static web pages

- Login page
- Registration page

3**.** Write programs for TCP server and Client interaction as per given below.
A program to create TCP server to send a message to client.

4. Write a program in Java Beans to add a Button to the Bean and display the number of times the button has been clicked.

5. Write a program in java bean to calculate simple Interest.

6. Write a program to display a greeting message in the browser by using Http Servlet.

7. Write a program to receive two numbers from a HTML form and display their sum in the browser by using Http Servlet.

8. Write a program to store the user information into Cookies. Write another program to display the above stored information by retrieving from Cookies.

9. Program to demonstrate dynamic html using java Servlet.

10.Basic JSP program

11. Write a JSP program to print even and odd numbers.

12. Program to verify the particular user and redirect to welcome page if credentials are valid else print proper message.

13. Write a program by using JDBC to execute a SQL query for a database and display the results.

14. Write a program by using JDBC to execute an update query without using Prepared Statement and display the results.

15. Write a program by using JDBC to execute an update query by using Prepared Statement and display the results.

TEXT BOOKS:

1. Professional Java Server Programming, J2EE 1.3 edition, APRESS publications
2. The complete Reference Java 8th Edition by Patrick Naughton and Herbert Schildt. TMH

REFERENCE BOOKS:

1. Programming world wide web-Sebesta, Pearson.

2. Core SERVLETS ANDJAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES by Marty Hall and Larry Brown Pearson.

3. Internet and World Wide Web – How to program by Dietel and Nieto PHI/Pearson Education Asia.

4. Murach’s beginning JAVA JDK 5, Murach, SPD.

5. An Introduction to web Design and Programming –Wang-Thomson.

6. Web Applications Technologies Concepts-Knuckles,John Wiley.

7. Web Warrior Guide to Web Programmming-Bai/Ekedaw-Thomas.

8. Beginning Web Programming-Jon Duckett WROX.

I MCA II SEMESTER	MCA – VR20	L	T	P	C
CourseCode: 4098201222	UNIX Programming	0	0	3	4.5

Course Overview:

This course introduces basic understanding of UNIX OS, UNIX commands and File system and to Familiarize students with the Linux environment To make student learn fundamentals of shell Scripting and shell programming Emphases are on making student familiar with UNIX environment and issues related to it.

Course Objectives:

1. You will be able to run various UNIX commands on a standard UNIX/LINUX Operating
   System (We will be using Ubuntu flavor of the Linux operating system).
2. You will be able to run C / C++ programs on UNIX.
3. You will be able to do shell programming on UNIX OS.
4. You will be able to understand and handle UNIX system calls

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Explain the architecture and features of UNIX Operating System and differentiate it from other Operating Systems	Understand/Remember	PO1
CO2	Demonstrate UNIX commands for file handling and process control	Understand	PO1,PO10
CO3	Build Regular expressions for pattern matching and apply them to various filters for a specific task	Apply	PO1,PO2,PO3,PO5
CO4	Analyze a given problem and apply requisite facets of SHELL programming in order to devise a SHELL script to solve the problem	Analyze	PO1,PO2,PO4,PO12

Syllabus

UNIT-I

Introduction to Unix-Brief History-What is Unix-Unix Components-Using Unix-Commands in Unix-Some Basic Commands-Command Substitution-Giving Multiple Commands. File Handling Utilities, Text Processing Utilities, Process Utilities, Disk Utilities

UNIT-II

The File system –The Basics of Files-What's in a File-Directories and File Names-Permissions-I Nodes-The Directory Hierarchy, File Attributes and Permissions-The File Command knowing the File Type-The Chmod Command Changing File Permissions-The Chown Command Changing the Owner of a File-The Chgrp Command Changing the Group of a File.

UNIT-III

Using the Shell-Command Line Structure-Meta characters-Creating New Commands-Command Arguments and Parameters-Program Output as Arguments-Shell Variables- -More on I/O Redirection-Looping in Shell Programs.

UNIT-IV

Shell Programming-Shell Variables-The Export Command-The Profile File a Script Run During Starting-The First Shell Script-The read Command-Positional parameters-The $? Variable knowing the exit Status-More about the Set Command-The Exit Command-Branching Control Structures-Loop Control Structures-The Continue and Break Statement-The Expr Command: Performing Integer Arithmetic-Real Arithmetic in Shell Programs.

UNIT-V

Unix Process :Threads and Signals: What is process, process structure, starting new process, waiting for a process, zombie process, process control, process identifiers, system call interface for process management, -fork, vfork, exit, wait, waitpid, exec, system, Threads, -Thread creation, waiting for a thread to terminate, thread synchronization, condition variables, cancelling a thread, threads vs. processes, Signals-, Signal functions, unreliable signals, interrupted system calls, kill and raise functions, alarm, pause functions, abort, sleep functions.

List of Experiments

1. Program using basic network commands.
2. Program using system calls: create, open, read, write, close, stat, fstat, lseek .
3. Write a shell script for sorting, searching and insertion/deletion of elements in a list
4. Write a program to display the good morning, good afternoon, good evening and good night depending on the users log on time
5. Create two processes to run a for loop, which adds numbers 1 to n, say one process adds odd numbers and the other even all lines containing the word.
6. Write a program to print prime numbers between x and y
7. Write a shell script which works similar to the wc command. This script can receive the option -l, -w, -c to indicate whether number of lines/ words/characters

8. Write a shell script that accepts any number of arguments and prints them in the reverse order.

9. Write a shell script that determines the period for which a specified user is working on the system

10. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

11. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

12. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
13. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.

14. Write a shell script which displays a list of all files in the current directory to which you have read, write and execute permissions

15. Write a shell script for renaming each file in the directory such that it will have the current shell's PID as an extension. The shell script should ensure that the directories do not get renamed

16. Write a shell script to list all of the directory files in a directory

17. Write a shell script that copies multiple files to a directory.

18. Write a shell script that counts the number of lines and words present in a given file.

TEXT BOOKS:

1. Unix and shell Programming, N B Venkateswarlu, Reem

2. Unix Concepts and Applications, 3/e, Sumitabha Das, TMH

REFERENCE BOOKS:

1. Unix and shell Programming, Sumitabha Das, TMH

2. A Beginner's Guide to Unix, N.P.Gopalan, B.Sivaselva, PHI

3. Unix Shell Programming, Stephen G.Kochan, Patrick Wood, 3/e, Pearson

4. Unix Programming, Kumar Saurabh, Wiley,India

5. Unix Shell Programming, Lowell Jay Arthus& Ted Burns,3/e, GalGotia

6. Nix Concepts and Applications, Das, 4/e, TMH

MCA Syllabus (VR-20)

(II MCA I Semester Syllabus)

CourseCode:	MCA – VR20	L	T	P	C
4098202120	Internet of Things (IOT)	3	0	3	4.5

Course Description:

The course covers IOT basic devices, connectivity, design principles and focuses on how to build and deploy IOT solutions. Practical case studies are included to ensure that the candidate develops an ability to work through practical real-life scenarios.

Courseoutcomes:

CO'S	Course outcome	Skill	PO
CO1	Understand the IOT connectivity principles and application areas	Remember/Understanding	PO1,PO2
CO2	Conceptually identify revolution of IOT in cloud, wireless sensors including recent attacks involving the Internet of Things	Analyzing	PO1,PO2,PO3
CO3	Build a real time IOT application	Applying	PO2, PO6,PO12
CO4	Evaluate the communication protocols for IOT	Applying	PO2, PO6,PO12

UNIT-I:The Internet of Things:

Introduction to Internet of things, Devices of IOT, Sources of IOT, M2MCommunication, Examples of IOT, Technology Behind IOT. Design Principles for Connected Devices, Communication Technologies Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

Course Outcomes: ToUnderstand the IOT connectivity principles and application areas

Activity: Introducing different types of sensors

UNIT-II:

Business Models for Business Processes in the Internet of Things, IOT/M2M systems layers and designs standardizations, Modified OSI Stack for the IOT, Dat Enrichment and Consolidation, Security issues in IOT, IoTcommunicationmodels.

Course Outcomes: To compare and contrast IOT and M2M

Activity: Introducing micro controller boards

UNIT-III:

Design Principles for the Web Connectivity for connected-Devices, Web Communication Protocols for Connected Devices, Message Communication protocols for Connected Devicesand Introduction about the Raspberry Pi Board.

Course Outcomes: To understand the web connectivity protocols

Activity: Basic experiment on Aurdino

UNIT-IV:

Data Acquiring, Organizing and Analytics in IOT/M2M, Applications/Services/Business Processes, IOT/M2M Data Acquiring and Storage, Organizing Data, Transactions, Business Processes.

Course Outcomes: To analyze the data acquiring and organizing in IOT/M2M

Activity: Interacting about different IOT devices

UNIT-V:

Data Collection, Storage and Computing Using cloud platform Services, IOT cloud-based services using the Xively, Case study- Building an IOT application.

Case study: Home Automation, smart parking, Environment – Weather monitoring system – Air Pollution Monitoring – Forest Fire Detection, Agriculture – smart irrigation system.

Course Outcomes: To apply the cloud platform services to the IOT

Activity: Application of IOT on various systems

Text Books:

1. Internet of Things: Architecture, Design Principles and Applications, Raj kamal, McGraw Hill Higher Education.
2. Internet of Things, A. Bahgya and V. Madisetti, University Press, 2015.

Reference Books:

1. Designing the Internet of Things, Adrian McEwen and Hakim Cassimally, Wiley.
2. Getting Started with the Internet of Things CunoPfister ,Oreilly.

LIST OF EXPERIMENTS

Course Overview:
Following are some of the programs that a student should be able to write and test on an Arduino/Raspberry Pi, but not limited to this only.

Course Objectives:

- To introduce the terminology, technology and its applications
- To introduce the concept of M2M (machine to machine) with necessary protocols
- To introduce the Python Scripting Language which is used in many IoT devices
- To introduce the Raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of web based services on IoT devices.

Course Outcomes:

- Identify problems that are amenable to solution by various methods, and which different methods may be suited to solving a given problem.
- Formalize a given problem in the language/framework of different methods (e.g., as a search problem, as a constraint satisfaction problem, as a planning problem, as a Markov decision process, etc).
- Implement basic algorithms (e.g., standard search algorithms or dynamic programming).
- Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports.

List of Experiments :

1 Start Raspberry Arduino/Pi and try various Linix commands in command terminal window: ls, cd, touch, mv, rm, man, mkdir, rmdir, tar, gzip, cat, more, less, ps, sudo, cron, chown, chgrp, ping etc.

2. Run some python programs on Arduino/Pi like:

a) Read your name and print Hello message with name
b) Read two numbers and print their sum, difference, product and division.
c) Word and character count of a given string
d) Area of a given shape (rectangle, triangle and circle) reading shape and appropriate values from standard input
e) Print a name 'n' times, where name and n are read from standard input, using for and while loops.
f) Handle Divided by Zero Exception.
g) Print current time for 10 times with an interval of 10 seconds.
h) Read a file line by line and print the word count of each line.

3. Light an LED through Python program
4. Get input from two switches and switch on corresponding LEDs
5. Flash an LED at a given on time and off time cycle, where the two times are taken from a file.
6. Flash an LED based on cron output (acts as an alarm)
7. Switch on a relay at a given time using cron, where the relay's contact terminals are connected to a load.
8. Get the status of a bulb at a remote place (on the LAN) through web.

The student should have hands on experience in using various sensors like temperature, humidity, smoke, light, etc. and should be able to use control web camera, network, and relays connected to the Arduino/Pi.

II MCA I SEMESTER	MCA – VR20	L	T	P	C
CourseCode: 4098202121	**Advanced Data Structures**	3	0	3	4.5

Course Description:

The purpose of this course is to provide the students with an exploration of advanced data structures (particularly persistent structures) using C. Course reviews main-memory data structures such as hash tables, Graphs and trees. Disk-based structures such as persistent hash tables and dictionaries. Digital Search Structures such as tries.

Courseoutcomes:

CO'S	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	To understand graph representations, Minimum Spanning Trees and traversals	Understanding	PO1, PO2
CO2	Understand dictionaries, hashing mechanism which supports faster retrieval.	Understanding	PO1, PO2
CO3	Implement heaps, queues and their operations, B Trees and B+ Trees	Applying	PO1,PO2,PO3,PO5
CO4	Illustration of tries which share some properties of table look up, various issues related to the design of file structures	Analyzing	PO1,PO2,PO3,PO4

UNIT-I:

GRAPHS: The Graph Abstract Data Type, Introduction, Definition, Graph Representation, Elementary Graph Operation, Depth First Search, Breadth First Search, Connected Components, Spanning Trees, Biconnected Components, Minimum Cost Spanning Trees, Kruskal's Algorithm, Prim s Algorithm Sollin's Algorithm, Shortest Paths and Transitive Closure, Single Source/All Destination: Nonnegative Edge Cost, Single Source/All Destination: General Weights, All-Pairs Shortest Path, Transitive Closure.

Outcome: To understand graph representations, Minimum Spanning Trees and traversals.

Experiments

a) Implement operations on Graphs.
   - i. Vertex insertion
   - ii. Vertex deletion
   - iii. Finding vertex
   - iv. Edge addition and deletion

b) To implement Prim's algorithm to generate a min-cost spanning tree.

c) To implement Krushkal's algorithm to generate a min-cost spanning tree.

d) To implement Dijkstra's algorithm to find shortest path in the graph.

e) Implement Depth for Search

f) Implement Breath for Search

UNIT-II:

HASHING: Dictionary ADT, Introduction-Static Hashing- Hash Table- Hash Functions- Secure Hash Function- Overflow Handling- Theoretical Evaluation of Overflow Techniques, Dynamic Hashing- Motivation for Dynamic Hashing -Dynamic Hashing Using Directories- Directory less Dynamic, Hashing.

Outcome: Understand dictionaries, hashing mechanism which supports faster retrieval.

Experiment

a) To implementation of Static Hashing (Use Linear probing for collision resolution)

b) To implement of Huffman coding.

UNIT-III:

PRIORITY QUEUES AND EFFICIENT BINARY SEARCH TREES:
Priority Queue ADT, Model, Simple Implementation, Binary Heap, Applications of Priority Queues- The Selection Problem Event Simulation Problem, Binomial Queues- Binomial Queue Structure – Binomial Queue Operation- Implementation of Binomial Queues. Binary Search Tree, AVL Tree, Insertion into a AVL Tree, Deletion from a AVL Tree, Red-Black Trees, Definition- Representation of a Red- Black Tree- Searching a Red-Black Tree- Inserting into a Red Black Tree- Deletion from a Red Black Tree, Splay Trees.

Outcome:
- ➢ Comprehension of heaps, queues and their operations
- ➢ Illustration of Balanced trees and their operations

Experiment:
- ➢ To perform various operations i.e., insertions and deletions on AVL trees.
- ➢ To Implement Red Black Trees

UNIT-IV:

MULTIWAY SEARCH TREES: M-Way Search Trees, Definition and Properties- Searching an M-Way Search Tree, B-Trees, Definition and Properties- Number of Elements in a B-tree- Insertion into B-Tree- Deletion from a B-Tree- B+-Tree Definition- Searching a B+-Tree- Insertion into B+-tree- Deletion from a B+-Tree.

Outcome: Incorporate data structures into the applications such as B Trees and B+ Trees

Experiment: To implement of B-tree.

UNIT-V:

DIGITAL SEARCH STRUCTURES: Digital Search Trees, Definition- Search, Insert and Delete- Binary tries and Patricia, Binary Tries, Compressed Binary Tries- Patricia, Multiway Tries- Definitions- Searching a Trie-Sampling Strategies- Insertion into a Trie- Deletion from a Trie- Keys with Different Length-Height of a Trie- Space Required and Alternative Node Structure- Prefix Search and Applications- Compressed Tries- Compressed Tries With Skip Fields- Compressed Tries With Labelled Edges- Space Required by a Compressed Tries, Tries and Internet Packet Forwarding ,-IP Routing- 1-Bit Tries- Fixed-Stride Tries-Variable-Stride Tries.

Outcome: Illustration of tries which share some properties of table look up, various issues related to the design of file structures

Experiment
Construct tries for the implementation of English Dictionary and Perform Searching of a word in dictionary.

Text Books:

1. Data Structures, a Pseudocode Approach, Richard F Gilberg, Behrouz A Forouzan, Cengage.
2. Fundamentals of DATA STRUCTURES in C: 2nded, , Horowitz , Sahani, Anderson-freed, Universities Press
3. Data structures and Algorithm Analysis in C, 2nd edition, Mark Allen Weiss, Pearson

Reference Books:

1. Web : http://lcm.csa.iisc.ernet.in/dsa/dsa.html
2. http://utubersity.com/?page_id=878
3. http://freevideolectures.com/Course/2519/C-Programming-and-Data-Structures
4. http://freevideolectures.com/Course/2279/Data-Structures-And-Algorithms
5. File Structures :An Object oriented approach with C++, 3rded, Michel J Folk, Greg Riccardi, Bill Zoellick
6. C and Data Structures: A Snap Shot oriented Treatise with Live examples from Science and Engineering, NB Venkateswarlu& EV Prasad, S Chand, 2010.

II MCA I SEMESTER	MCA – VR20	L	T	P	C
CourseCode: 4098202100	**Machine Learning**	3	0	0	3

Course Description:

This course provides how to recognize the characteristics and applications of machine learning. The course provides brief information about supervised, unsupervised, clustering algorithms and Artificial neural networks.

1. Familiarity with a set of well-known supervised, unsupervised and semi-supervised learning.
2. The ability to implement basic machine learning algorithms.
3. Understanding of how machine learning algorithms are evaluated.
   Applying new concepts in machine learning.

Courseoutcomes:

CO'S	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Recognize the characteristics of machine learning that make it useful to real-world Problems	Understand/Remember	PO1,PO3,PO4
CO2	Characterize machine learning algorithms as supervised, semi-supervised, and Unsupervised	Understand/analyze	PO1,PO5,PO9,
CO3	Be able to use support vector machine, regularized regression algorithms	Create	PO3,PO5,PO9,PO2
CO4	Understand the concept behind neural networks for learning non-linear functions	Evaluate/Apply	PO2,PO3,PO5,PO12

UNIT-I:Introduction to Machine Learning

Introduction to machine learning, Definition, traditional programming vs machine learning algorithms, learning a system, supervised learning, unsupervised learning and reinforcement learning, application areas.

Outcome: Familiarize with a set of well-known problem and feature of machine learning.

A**ctivity:**Apply characteristics with real world problems.

UNIT-II: Classification and Regression Models

Linear separability and decision regions, linear discriminants, linear regression, logistic regression, decision trees-ID3 and C4.5, KNN.

Outcome: Able to understand the difference between classification and regression problems.

A**ctivity:**Regression analysis and hypothesis space.

UNIT-III: Dimensionality reduction and Support vector machines

Dimensionality reduction and Feature selection, Dimensionality reduction algorithms: LDA and PCA, Margin of a classifier, Support Vector Machine, Learning nonlinear hypothesis using kernel functions.

Outcome: Ability to implement basic machine learning algorithms.

A**ctivity:**Implement tree models,rule models and linear models on simple dataset.

UNIT-IV: Clustering and Ensemble Methods

Introduction to clustering: K-means clustering, Gaussian mixture model, Ensemble Methods: bagging and boosting, Random forest and AdaBoost algorithms and Bayesian learning algorithm.

Outcome: Able to understand how to evaluate machine learning algorithms.

A**ctivity:**Implement classification algorithms on a dataset.

UNIT-V: Artificial Neural Networks

Introduction,Theperceptron,the perceptron learning algorithm, Multilayer neural networks,activation functions, Back Propagationalgorithm and introduction to Deep learning models: CNN.

Outcome: Able to understand the ensemble methods and ANN algorithms.

A**ctivity:**Understand the concept of neural networks for learning non-linear functions.

Text Books:

1. Tom Mitchell, *"Machine Learning"*, McGrawHill publications, 1997
2. Machine Learning: The art and science of algorithms that make sense of data, Peter Flach,Cambridge.
3. Introduction to Machine Learning with Python By Andreas C. Müller, Sarah Guido O'Reilly Media
4. Deep Learning by Josh Patterson, Adam Gibson ,O'Reilly Media

Reference Books:

1. UnderstandingMachine Learning: From Theory toAlgorithms, ShaiShalev-Shwartz, Shai Ben- David, Cambridge.

2. Machine Learning in Action, Peter Harington, 2012, Cengage.

II MCA I SEMESTER	MCA – VR20	L	T	P	C
CourseCode: 4098202101	**Cryptography & Network Security**	3	0	0	3

Course Description:

The objective of this course is to equip the students with principles and practice of cryptography and network security, Classical systems, symmetric block ciphers (DES, AES, other contemporary symmetric ciphers), Public-key cryptography (RSA, Discrete logarithms), Algorithms for factoring and discrete logarithms, cryptographic protocols, hash functions, authentication, key management, key exchange, signature schemes, Email and web security.

1. Understanding the requirement of security in modern communication and information systems.
2. Mastering the concept of security attack, services and mechanisms.
3. Mastering concepts of confidentiality using cryptography with mathematical background.
4. Mastering concept of authentication using hash algorithms and digital signature
   To be familiar with network security designs using available secure solutions (PGP, SSL, IPSec).

Courseoutcomes:

CO'S	Course outcome	Skill	PO
CO1	Understand the principles and practices involved in cryptography and network security	Understand/Remember	PO1
CO2	Understand the various symmetric and Asymmetric encryption algorithms.	Understand	PO1,PO2
CO3	Identifying cryptographic protocols, hash functions, authentication, key management, key exchange, signature schemes	Apply	PO2,PO3,PO4,PO12
CO4	Design of network security solutions for E-mail Security like PGP,S/MIME and web security like SSL,TLS .	Evaluate/Create	PO1,PO2,PO3,PO4, PO6,PO12

UNIT-I:

Security attacks, services & mechanisms, fundamental security principles, A Model for Network Security, Symmetric Cipher Model, Substitution Techniques Transportation Techniques, Rotor Machines, steganography.

Outcome: Demonstrate a systematic and critical understanding of the theories, principles and practices of Cryptography and network security.

A**ctivity:**Implement a simple cryptographic function.

UNIT-II:

Secret Key Cryptography: Traditional Block Cipher Structure**,** Data Encryption Standard (DES), Block Cipher Design Principles, Triple DES, Blowfish, AES, Steam ciphers, RC4, Modes of Operation.

Outcome: Understand fundamentals of symmetric cryptographic algorithms like DES, AES, and BLOWFISH etc.

A**ctivity:**Implement a AES algorithm in Python Language.

UNIT-III:

Number Theory: Prime and Relatively Prime Numbers, Modular Arithmetic, Fermat's and Euler's Theorems, the Chinese Remainder Theorem, Discrete Logarithms. Public Key Cryptography: Principles of Public Key Cryptosystems, RSA Algorithm, Diffie-Hellman Key Exchange, Introduction to Elliptic Curve Cryptography.

Outcome: Understand the asymmetric cryptography algorithms like RSA, Elliptical Cryptography etc.

A**ctivity:**Bit coin and time stamp server.

UNIT-IV:

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Secure Hash Algorithm (SHA), Message Authentication Codes - Message Authentication Requirements and Functions, HMAC, Digital signatures, RSA Digital Signature Scheme, NIST Digital Signature Schemes (DSA approach)

Outcome: Understand the Authentication functions the manner in which Message Authentication Codes and Hash Functions works.

A**ctivity:**Generate digital signature for a given message.

UNIT-V:

Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS, Security at the Network Layer: IPSec, System Security.

Outcome: Understand existing system security protocols like Kerberos, PGP, SSL and IPSEC.

A**ctivity:**Develop a web page using a protocol HTTPS**.**

Text Books:

1. Cryptography and Network Security, Behrouz A Forouzan, DebdeepMukhopadhyay, (3e) McGraw Hill.
2. Cryptography and Network Security, William Stallings, (7e) Pearson.

Reference Books:

1. Everyday Cryptography, Keith M.Martin, Oxford.
2. Network Security and Cryptography, Bernard Meneges, Cengage Learning.
3. Cryptography and Network Security: AtulKahate, McGraw Hill, 2nd Edition.
4. Information Security, Principles and Practice : Mark Stamp, Wiley India.
5. Principles of Computer Sceurity: WM.Arthur Conklin, Greg White, TMH
6. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.
7. Principles of Information security by Michael E Whitman and Herbert J.Mattord.
8. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.

II MCA I SEMESTER	MCA – VR20	L	T	P	C
CourseCode: 4098202150	**Object Oriented Analysis and Design Through UML** **(Professional Elective-1)**	3	0	0	3

Course Description:

This course is intended to provide an in depth understanding of object oriented approaches to software development, in particular to the analysis and design phases of the software life cycle. Topics include notation, methods, competing methodologies, issues in object oriented development, and recent advancements which complement traditional object-oriented methodologies.

1. Essential and fundamental aspects of object-oriented analysis and design, in terms of "how to use" it for the purpose of specifying and developing software.
2. Explore and analyze different analysis and design models, such OO Models, Structured Analysis and Design Models, etc
3. Understanding the insight and knowledge into analyzing and designing software using different object-oriented modeling techniques
   Understanding the fundamental principles through advanced concepts of analysis and design using UML

Courseoutcomes:

CO'S	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO

CO1	Possess an ability to practically apply knowledge software engineering methods, such as object-oriented analysis and design methods with a clear emphasis on UML	Understand	PO1(3),PO2(3), PO3(3), PO4(2),PO5(2), PO12(1)
CO2	Have a working ability and grasping attitude to design and conduct object-oriented analysis and design experiments using UML, as well as to analyze and evaluate their Models.	Understand	PO1(3),PO2(3), PO3(3), PO4(2),PO6(1), PO7(1), PO12(1)
CO3	Display an ability to identify, formulate and solve software development problems: software requirements, specification (problem space), Software design, and implementation (solution space).	Analysis	PO1(3),PO2(3), PO3(2), PO4(1)
CO4	Show an ability to use the graphical UML representation using tools, such as IBM's Rational Rose or Microsoft's Vision.	Understand/Apply	PO1(3),PO2(3), PO3(3), PO4(2)

1: Weakly correlated 2: Moderately correlated 3: Strongly correlated

UNIT-I:Introduction to UML

The meaning of Object-Orientation, object identity, encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture.

Outcome: Possess an ability to practically apply knowledge software engineering methods, such as object-oriented analysis and design methods with a clear emphasis on UML

Activity: Seminar on Case Study and Class test.

UNIT-II: Basic Structural Modeling & Class & Object Diagrams

Basic Structural Modeling: Classes, relationships, common mechanisms, diagrams, Advanced structural modeling: advanced relationships, interfaces, types & roles, packages, instances.

Class & Object diagrams: Terms, concepts, examples, modeling techniques, class & Object diagrams.

Outcome: Have a working ability and grasping attitude to design and conduct object-oriented analysis and design experiments using UML, as well as to analyze and evaluate their Models.

Activity: Seminar on Case Study and Class test.

UNIT-III:Sequence Diagrams & Collaboration Diagrams

Sequence Diagrams: Terms, concepts, differences between collaboration and sequence diagrams, depicting synchronous messages with/without priority call back mechanism broadcast message.

Collaboration Diagrams: Terms, Concepts, depicting a message, polymorphism in collaboration diagrams, iterated messages, use of self in messages.

Outcome: Have a capacity to analyze and design software systems, components to meet desired needs

A**ctivity:** Seminar on Case Study and Class test.

UNIT-IV: Behavioral Modeling & Advanced Behavioral Modeling

Behavioral Modeling: Interactions, use cases, use case diagrams, activity diagrams.

Advanced Behavioral Modeling: Events and signals, state machines, processes & threads, time and space, state chart diagrams

Outcome: Display an ability to identify, formulate and solve software development problems: software requirements, specification (problem space), Software design, and implementation (solution space).

A**ctivity:** Seminar on Case Study and Class test.

UNIT-V:Architectural Modeling

Terms, concepts, examples, modeling techniques for component diagrams and deployment diagrams.

Outcome: Show an ability to use the graphical UML representation using tools, such as IBM's Rational Rose or Microsoft's Vision.

A**ctivity:** Seminar on Case Study and Class test.

Text Books:

1. The Unified Modeling Language User Guide, Grady Booch, Rambaugh, Ivar Jacobson, PEA.
2. Fundamentals of Object Oriented Design in UML, Meilir Page- Jones, Addison Wesley.

Reference Books:

1. Head First Object Oriented Analysis &Design, Mclaughlin, SPD OReilly,2006
2. Object oriented Analysis& Design Using UML, Mahesh ,PHI
3. The Unified Modeling Language Reference Manual, 2/e, Rambaugh, GradyBooch,etc.,

PEA
4. Object Oriented Analysis & Design, Satzinger, Jackson, Thomson
5. Object Oriented Analysis Design & implementation, Dathan.,Ramnath, University Press
6. Object Oriented Analysis & Design, John Deacon, PEA
7. Fundamentals of Object Oriented Analysis and Design in UML, M Pages-Jones, PEA
8. Object-Oriented Design with UML, Barclay, Savage, Elsevier,2008

II MCA I SEMESTER	MCA – VR20	L	T	P	C
Course Code: 4098202151	Artificial Intelligence (Professional Elective-1)	3	0	0	3

Course Description:

Artificial intelligence (AI) is a research field that studies how to realize the intelligent human behaviors on a computer. The ultimate goal of AI is to make a computer that can learn, plan, and solve problems autonomously. Although AI has been studied for more than half a century, we still cannot make a computer that is as intelligent as a human in all aspects.

Courseoutcomes:

COS	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Identify Methods in AI that may be suited to solving a given problem and Game Playing	Understanding	PO1, PO2
CO2	Analyze the basic issues of different types of knowledge representation techniques to build intelligent system	Analyze	PO2, PO3
CO3	Build Expert systems for real time applications	Create	PO2, PO3, PO5, PO7
CO4	Determination of uncertainty of data using different probability approaches for real time applications	Analyze	PO1, PO2, PO4

UNIT-I:

Introduction to artificial intelligence:

Introduction, history, intelligent systems, foundations of AI, Applications of AI, current trends in AI

Problem solving:

Definition, characteristics of problem, types of Problem solving techniques, General Problem Solver (GPS), Water Jug Problem, Missionaries and Cannibals Problem.

Outcome: Understand need of AI and its applications.

Ac**tivity:**Write a Report on the use of artificial intelligence in every day

UNIT-II:

Search Techniques:

State Space Search**,** Definition, Examples, Exhaustive search techniques: BFS, DFS, IDDFS, Heuristic search techniques: Uniform Cost Search, Best First Search, A* algorithm & Constraint satisfaction Problem

Game playing:

Introduction about game playing, Mini-Max Algorithm, Alpha-Beta pruning algorithm.

Outcome: Identify Methods in AI that may be suited to solving a given problem and Game Playing.

Ac**tivity:** Identify Methods in AI used to solve the tricky puzzles.

UNIT-III:

Logic concepts:

Introduction, propositional logic, natural deduction system, axiomatic system, semantic tableau system in propositional logic, resolution refutation in propositional logic,

Predicate logic:

Introduction, PNX Normal form, Resolution in Predicate Logic

Outcome: Able to understand the uses of logic concepts and predicate logic.

A**ctivity:** Convert Natural Language Sentences To Predicate Logic.

UNIT-IV:

Knowledge representation:

Introduction, approaches to knowledge representation, knowledge representation using semantic network, knowledge representation using frames.

Uncertainty measure:

Introduction to probability theory, Bayesian belief networks, Certainty factor theory, Dempster-Shafer theory.

Outcome:

- Analyze the basic issues of different types of knowledge representation techniques to build intelligent system.
- Determination of uncertainty of data using different probability approaches for real time applications.

A**ctivity:** Consider any data set and classify the test instances using BBN.

UNIT-V:

Expert system and applications:

Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems, application of expert systems, Black board Systems, TMS.

Fuzzy stand fuzzy logic:

Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges

Outcome: Build Expert systems for real time applications.

A**ctivity:** Explain the use of fuzzy logic in the real time scenario.

Text Books:

1. Artificial Intelligence- SarojKaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach , 2nded, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rded, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

Reference Books:

1. Artificial intelligence, structures and Strategies for Complex problem solving, - George F. Lugar, 5th edition, PEA.
2. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer.
3. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier.

II MCA I SEMESTER	MCA – VR20	L	T	P	C
Course Code: 4098202152	Big Data Analytics (Professional Elective-1)	3	0	0	3

Course Description:

This course provides practical foundation level training that enables immediate and effective participation in big data projects. The course provides grounding in basic and advanced methods to big data technology and tools, including MapReduce and Hadoop and its ecosystem.

5. To introduce programming tools PIG & HIVE in Hadoop echo system.

Courseoutcomes:

CO'S	Course outcome	Skill	PO
CO1	Preparing for data summarization, query, and analysis.	Understanding	PO1, PO2
CO2	Applying data modeling techniques to large data sets	Applying	PO1, PO2, PO4
CO3	Creating applications for Big Data analytics	Applying	PO1, PO2, PO4
CO4	Building a complete business data analytic solution	Applying	PO1, PO2, PO4, PO5

UNIT-I:

Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Namenode, Datanode, Secondary Namenode, JobTracker, TaskTracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

Outcome:

- Analyze the distinction between GFS and HDFS
- Demonstrate understanding of different mode of Hadoop Installations.

A**ctivity:**Installation of Hadoop and configuring various XML files

UNIT-II:

Writing MapReduce Programs: A Weather Dataset, Understanding Hadoop API for MapReduce Framework (Old and New), Basic programs of HadoopMapReduce: Driver code, Mapper code, Reducer code, RecordReader, Combiner, Partitioner.

Outcome:

- Analyze the distinction MapReduce execution of old and new versions
- Demonstrate understanding of different code blocks

A**ctivity:**Running of Map Reduce program to forecast weather.

UNIT-III:

Hadoop I/O: The Writable Interface, WritableComparable and comparators, Writable Classes: Writable wrappers for Java primitives, Text, BytesWritable, NullWritable, ObjectWritable and GenericWritable, Writable collections, Implementing a Custom Writable: Implementing a RawComparator for speed, Custom comparators

Outcome:

- Understanding the I/O classes used for HadoopMapReduce concept
- Able to write wrapper classes and Generic class programs

A**ctivity:**Implementation of I/O operations using writable wrappers.

UNIT-IV:

Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin.

Outcome:

- Understand the PIG Architecture and Modes of operations.
- Able to write the PIG scripts

A**ctivity:**Installation of PIG and running pig scripts on different modes

UNIT-V:

Applying Structure to Hadoop Data with Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data.

Outcome:

- Understand the HIVE Architecture and Modes of operations.
- Able to create database on HIVE environment.

A**ctivity:**Installation of HIVE and running the queries on database.

Text Books:

1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly
3. Hadoop in Action by Chuck Lam, MANNING Publ.
4. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss.

Reference Books:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. HadoopMapReduce Cookbook, SrinathPerera, ThilinaGunarathne

II MCA I SEMESTER	MCA – VR20	L	T	P	C
CourseCode: 4098202153	Web designing using PHP (Professional Elective-1)	3	0	0	3

Course Description:

Web designing using php is a powerful open source technologies that allows programmers to generate interactive and dynamic websites. This will also expose students to the basic tools and applications used in Web publishing.

1. To understand the fundamentals of web development.
2. To design static web pages using basic HTML elements.
3. Able to design dynamic web pages using php.
4. To understand the basic concepts of how a database is used in web app.

Courseoutcomes:

CO'S	At the end of the course , the students will have the ability to:	PO's mapped	Strength of mapping
CO1	Create web pages using HTML,JS,CSS	2,3,5	2
CO2	Able to differentiate client side and server side scripting.	2,3,5	3
CO3	Building dynamic web based solutions based on user requirements by using the knowledge of php and MySQL.	2,3,4,5,6,9,10	3
CO4	Adopt design based approaches and frameworks which will make students ready for industry.	3	3

UNIT-I:

Introduction to HTML: Basic Syntax, Standard HTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, HTML5.

CSS: Levels of Style Sheets, Style Specification Formats, Selector Forms, the Box Model, Conflict Resolution.

Outcome: Understand HTML tags to design static web pages

Activity: Design Static website using basic tags and Lists, Tables, Forms.

UNIT-II:

Java Script: The Basic of Java script: Objects, Primitives Operations and Expressions, Screen Output and Keyboard Input, Control Statements, Object Creation and Modification, Arrays, Functions, Constructors, Pattern Matching using Regular Expression.

Ajax A New Approach: Introduction to AJAX, Integrating php and ajax.

Outcome: Describe the basic concepts of Java Scripts to design dynamic web pages

A**ctivity:** Validate the login and registration pages.

UNIT-III:

PHP Programming: Configuration of PHP, Apache Web Server, MySQL, Installing PHP, PHP Structure and Syntax ,Creating PHP pages, Rules of PHP syntax , Integrating HTML with PHP Constants , Variables: Static and Global Variable, data types Conditional Structure and Looping, PHP operators ,Arrays, constructs, functions, Variable Function, string function, MATH functions, Date function , Array Function ,File function

Outcome: Understand the basic concepts of php.

A**ctivity:** Designing basic programs using php.

UNIT-IV:

Form Controls: Reading data using Form Controls, Submitting form values, using $_Get and $_Post Methods, $_REQUEST, Accessing form inputs with Get/Post functions , Combining HTML and PHP codes together on single page, Redirecting the user.

Concepts and Installation of MySQL: MySQL structure and syntax, Types of MySQL tables and Storage engines MySQL commands 5.5 Integration of PHP with MySQL, Connection to the MySQL Database, Creating and Deleting MySQL database using PHP , Updating, Inserting, Deleting records in the MySQL database.

Outcome: Understanding to get data from a page using website.

A**ctivity:** Design a form such that it accepts input from the user and post the data in the database.

UNIT-V:

Working with CodeIgniter: Understanding CodeIgniter. Installation of CodeIgniter, Applications and File Structure, MVC Architecture, error handling, Routes, Creating forms and form validation, Active Records.

Outcome: To understand the working of CodeIgniter framework with basic programming.

Activity: Designing a form Using CodeIgniter.

Text Books:

1. Beginning PHP and MySQL, 4th Edition, W. Jason Gilmore, Apress, 2010
2. Learning PHP, MySQL, JavaScript, CSS & HTML5, Third Edition, Robin Nixon, O'reillyMedia, 2014
3. CodeIgniter 1.7 Jose Argudo Blanco David Upton
4. Web Development with Jquery, Richard York, Wrox.

Reference Books:

1. PHP: The Complete Reference, Steven Holzner, McGraw Hill
2. Teach yourself PHP, MySQL and Apache All in One ,JulieC.Meloni, Pearson Education.

e-Resources:

1. Software: WAMP server / XAMPP server, 'C' Panel, Text Editor
2. http://www.codecademy.com/tracks/web
3. http://www.codeacademy.com/tracks/php
4. http://w3schools.com/PHP

II MCA I SEMESTER	MCA – VR20	L	T	P	C
Course Code: 4098202154	Data Science using Python (Professional Elective-2)	3	0	0	3

Course Description:

Data science is one of the most debated topics in the industries these days.This is course mainly focuses on basics of data science and libraries used in data science for loading data and analyze data using visualization.

Courseoutcomes:

CO'S	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Recognize and understand the significance of data science.	Understanding	PO1,PO3,PO5
CO2	Explain how data is collected, managed and stored for data science.	Understanding, Analyze	PO1,PO2,PO5
CO3	Able to understand the features of Numpy library for analyzing the data. Make use of various file formats in loading and storage of data.	Apply	PO2,PO3,PO5
CO4	Analyze Data using various Visualization techniques.	Apply and Analyze	PO1,PO2,PO3,PO5

UNIT-I:

About Data Science: Computer Science, Data Science, and Real Science, Asking Interesting Questions from Data, Properties of Data, Classification and Regression, Data Science Television: The Quant Shop, Kaggle Challenges. **[10 Hours]**

Learning Outcomes:

- Understand the basics of data science.

Activity: Interpret various types of data objects.

UNIT II:

Data Munging: Languages for Data Science, Collecting Data, Cleaning Data, Crowd sourcing (Text Book 1) Gradient Descent, Getting and working with data (Text book 2)

Learning Outcomes:

- Understand the concepts of Data collection and management.
- Establish sources of data.
- Interpret strategies for data storage and management

Activity: Apply Gradient Descent by taking a case study.

UNIT III:

NumPy Basics: The NumPyNd array: A Multidimensional Array Object, Creating ndarrays, Data Types for ndarrays, Operations between Arrays and Scalars, Basic Indexing and Slicing, Boolean Indexing, Fancy Indexing, Data Processing Using Arrays, Expressing Conditional Logic as Array Operations, Methods for Boolean Arrays , Sorting , Unique.(Text Book 3) **[10 Hours]**

Learning Outcome:

- Able to understand the features of NumPy library for analysing the data.

Activity: Implement a python program using NumPy library.

UNIT-IV:

Getting Started with pandas: Introduction to pandas, Library Architecture, Features, Applications, Data Structures, Series, Data Frame, Index Objects, Essential Functionality Reindexing, Dropping entries from an axis, Indexing, selection, and filtering, Sorting and ranking, Summarizing and Computing Descriptive Statistics, Unique Values, Value Counts, Handling Missing Data, filtering out missing data.(Text Book 3) **[10 Hours]**

Learning Outcome: Summarize the features of pandas library for analyzing the data.
Activity: Design an Application using Pandas

UNIT- V:

Visualizing Data: Exploratory Data Analysis, Developing a Visualization Aesthetic, Chart Types, Great Visualizations Mathematical Models: Philosophies of Modeling, A Taxonomy of Models, Baseline Models, Evaluating Models, Evaluation Environments. (Text Book 1)

Plotting and Visualization: A Brief matplotlib API Primer, Figures and Subplots, Colors, Markers, and Line Styles, Ticks, Labels, and Legends, Annotations and Drawing on a Subplot, Saving Plots to File, Plotting Functions in pandas, Line Plots, Bar Plots, Histograms and Density Plots, Scatter Plots. (Text Book 3) **[12 Hours]**

Learning Outcomes: Understand types of data Visualization techniques. Evaluate models for multiple environments.

Activity: Apply various plotting techniques using matplotlib for any case study.

Text Books:

1. The Data Science Design Manual, Steven S. Skiena, Springer 2017.
2. Data Science from Scratch First Edition, Joel Grus April 2015
3. Wes McKinney, ―Python for Data Analysis,O‘REILLY, ISBN:978-1-449-31979-3,1st edition, October 2012.
4. RachelSchutt&O‘neil, ―Doing Data Science, O‘REILLY, ISBN: 978-1-449-35865-5, 1st edition, October 2013.

Reference Books:

1. Matt Harrison, ―Learning the Pandas Library: Python Tools for Data Munging, Analysis, and Visualization, O'Reilly, 2016.
2. NumPy v1.20 Manual.

II MCA I SEMESTER	MCA – VR20	L	T	P	C
Course Code: **4098202155**	**Principles and Practices of Management (Professional Elective-2)**	3	0	0	3

Course Description:

This course presents the principles, techniques, and concepts needed for managerial analysis and decision-making. It highlights the effective management of planning, organizing, influencing, and controlling related to the internal and external environment and issues of ethics and social responsibility. And aims at ensuring the overall efficiency of the business. Management principles provide guidelines as to how tasks are to be completed for increased efficiency.

At the end of the course, the students will be able to:

- To know the basic management concepts.
- To understand the role and qualities of manager.
- To know Ethical Responsibilities of manager.
- To understand the concept of SWOT analysis of business.
- To able to know the coordination and controlling process.

Courseoutcomes:

CO's	**Course outcome**	**PO Mapped**	**Strength or Mapping**
CO1	Understand the basic functions of management and its role in business.	PO1	3
CO2	To understand the decision making process, SWOT of business organization.	PO1,PO2,PO6	3
CO3	To able to know the directing skills and controlling techniques of business.	PO2, PO4	3
CO4	To understand the coordinating and its importance in the business	PO6, PO7, PO8	3

UNIT-I:

MANAGEMENT: Definition, nature, purpose and scope of management - Functions and Roles of a manager - an overview of planning, organizing and controlling - Is managing a science or art? Ethics in managing and social responsibility of managers - Evolution of management thought. Various approaches to Management - Decision Theory approach. Systems Approach

UNIT-II: PLANNING & DECISION MAKING: Types of plans, steps in planning, and process of planning. Nature of objectives, setting objectives. Concept and process of Managing by Objectives(MBO). Strategic planning process. SWOT analysis, Portfolio matrix, premising and forecasting. Decision Making: Meaning, Importance and steps in Decision Making - Traditional approaches to decision-making.

UNIT-III:
ORGANIZING: Concept of organization, process of organizing, bases of Depart mentation, Authority& power - concept &distinction.Various types of organization structures -Delegation - concept of delegation; elements of delegation - authority, responsibility, accountability.Reasons for failure of delegation & how to make delegation effective. Decentralization - concept, reasons for decentralization. Span of Management - concept, early ideas on span of management.

UNIT-IV:
DIRECTING: Motivation and Motivators: Concept, Theories of Motivation: Hierarchy of Needs, Motivation-Hygiene Expectancy, Equity, Reinforcement, McClelland's needs - Leadership: Meaning, Definition, – Trait Approaches of Leadership – Leadership Behavior and Styles – Contingency Approaches to Leadership – Communication: Meaning, Process, and Importance in Functions of Organization – Barriers in Communication – Effective Communication.

UNIT-V:
CO-ORDINATION AND CONTROL: Concept and importance of coordination; techniques or methods to ensure effective coordination. Control: Concept, planning-control relationship, process of control - setting objectives, establishing standards, measuring performance, correcting deviations. Dimensions or Types of Control: Feed forward control, Concurrent Control (Real Time Information & Control), Feedback Control - Techniques of Control: Brief review of Traditional and Modern Techniques of Control.

Text Books:

1. Stoner, Freeman and Gilbert, "Jr. Management", 6th Edition, Pearson Education, NewDelhi, 2006.
2. Heinz Weihrich, Harold Koontz,"Management A Global Perspective",10th Edition, Tata McGraw Hill, 2007.

Reference Books:

1. Harold Koontz, Heinz Weihrich, A.R.Aryasri, Principles of Management, TMH, 2010.
2. Dilip Kumar Battacharya, Principles of Management, Pearson, 2012.
3. Kumar, Rao, Chhaalill ―Introduction to Management Science‖ CengagePublications,New Delhi
4. V.S.P.Rao, Management Text and Cases, Excel, Second Edition, 2012.
5. K.Anbuvelan, Principles of Management, University SciencePress, 2013.
6. K.Aswathappa ― Organisational Behaviour-Text, Cases andGames‖, Himalaya Publishing House, New Delhi,2008.
7. Steven L Mc Shane, Mary Ann Von Glinow, Radha R Sharma: ―Organisational Behaviour‖, TMH Education, New Delhi,2008
8. Daft, "The New Era of Management", 7th Edition,Thomson New Delhi, 2007.

9. “Schermerhorn: Management”, 8th, Wiley India, 2006.

II MCA I SEMESTER	MCA – VR20	L	T	P	C
Course Code: 4098202156	**Cloud Computing (Professional Elective-2)**	3	0	0	3

Course Description:

This course provides the comprehensive study of key concepts of Cloud & its environment. Building software systems and components that scale to millions of users in modern internet. Cloud concepts capabilities across the various cloud service models including Iaas, Paas, Saas. Developing cloud based software applications on top of cloud platforms. Storage systems and backup strategies for cloud based data.

To learn the basics of Cloud computing
To know the key concepts of Virtualization
To gain knowledge on cloud computing service models
To develop cloud implementation, programming and mobile cloud computing
To learn key components of Amazon web services
To maintain the Cloud backup and solutions.

Courseoutcomes:

CO’s	**Course outcome**	**Cognitive Level as per Bloom’s Taxonomy**	**PO**
CO1	Define cloud computing and memorize the different cloud computing service and deployment models	Understand/ Remember	PO1
CO2	Describe the importance of virtualization along with their technologies for data center automation	Understand	PO1,PO5
CO3	Identify emerging cloud programming paradigms and its software environments	Apply	PO1,PO3,PO5
CO4	Describe the key application of cloud resource management. Design and develop the backup strategies for cloud data based on features.	Analyze	PO1, PO3, PO10

UNIT-I:

Systems modeling, Clustering, and virtualization of Clusters and Data Centers- Scalable Computing over the Internet, Technologies for Network based systems, System models for Distributed and Cloud Computing, Software environments for distributed systems and clouds, Performance, Security And Energy Efficiency. Implementation Levels of Virtualization, Virtualization Structures/ Tools and mechanisms, Virtualization of CPU, Memory and I/O Devices.

Outcome: understand the basics of cloud computing and key concepts of virtualization

A**ctivity:**virtualization tools and mechanisms.

UNIT-II:

Cloud Platform Architecture-Cloud Computing and service Models, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms, Inter Cloud Resource Management, Cloud Security and Trust Management. Service Oriented Architecture, Message Oriented Middleware Existing cloud applications and new opportunities, Map Reduce program model.

Outcome: To gain knowledge on cloud computing service models, architectural design of store clouds

A**ctivity:**Critique the consistency of services deployed from a cloud architecture.

UNIT-III:

Cloud Programming and Software Environments-Features of Cloud and Grid Platforms, Parallel & Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments, Data sources and sinks with an emphasis on MS Azure, Google Cloud.

Outcome:

- Able to develop cloud implementation, programming and mobile cloud computing,
- Able to learn key components of Amazon web services.

A**ctivity:**Compare and contrast the economic benefits delivered by various cloud models based on application requirements, economic constraints and business requirements.

UNIT-IV:

Cloud Resource Management and Scheduling-Policies and Mechanisms for Resource Management Applications of Control Theory to Task Scheduling on a Cloud, Stability of a Two Level Resource Allocation Architecture, Feedback Control Based on Dynamic Thresholds. Coordination of Specialized Autonomic Performance Managers, Resource Bundling, Scheduling Algorithms for Computing Clouds, Fair Queuing, Start Time Fair Queuing, Borrowed Virtual Time, Cloud Scheduling Subject to Deadlines, Scheduling Map Reduce Applications Subject to Deadlines.

Outcome: understand the application of cloud resource management and scheduling algorithms for computing clouds

A**ctivity:**cloud resource management and scheduling algorithms.

UNIT-V:

Storage Systems-Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system. Apache Hadoop, Big Table, Megastore, Amazon Simple Storage Service (S3).

Cloud Security and trust Management- Cloud Security defence strategies, Distributed intrusion/ Anomaly detection. Data and software protection techniques, Reputation guided protection of Data Centres, OS security, security risks

Outcome: Analyze and develop the backup strategies for cloud data based on features

A**ctivity:**Critically analyze case studies to derive the best practice model to apply when developing and deploying cloud based applications

Text Books:

1. Distributed and Cloud Computing, Kai Hwang, Geoffry C. Fox, Jack J. Dongarra MK Elsevier.
2. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.
3. Cloud Computing, A Hands on approach, ArshadeepBahga, Vijay Madisetti, University Press

Reference Books:

1. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH
2. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammaraiselvi, TMH.

CourseCode:	Full Stack Development (MERN) (Professional Elective-2)	L	T	P	C
4098202157		3	0	0	3

Course Description:

MERN Stack is a Java script Stack that is used for easier and faster deployment of full-stack web applications. MERN Stack comprises of 4 technologies. namely: MongoDB, Express, React and Node.js. It is designed to make the development process smoother and easier. Each of these 4 powerful technologies provides an end-to-end framework for the developers to work in and each of these technologies play a big part in the development of web applications.

Course Objectives:

1. To design as web page using front end technologies
2. To develop application with server side scripting tools
3. To develop web application with REST APIs and use of framework to communicate client-server applications.
   To build as responsive web application with managing NOSQL databases

Courseoutcomes:

CO's	Course outcome	Skill	Program Outcomes (POs)
CO1	Demonstrate basic concepts of react, node, express and mongodb technologies.	Understanding	PO5, PO11
CO2	Design front end application using React and Redux libraries.	Applying	PO2, PO4, PO5, PO11
CO3	Develop interactive web applications on server side with NOSQL databases.	Applying	PO2, PO4, PO5, PO7,P O11
CO4	Build responsive web application communicating with RES API and managing data with NOSQL databases.	Understanding/ Applying	PO5, PO11

UNIT-I:Introduction to React

Welcome to React: Obstacles and Roadblocks, React's future, keeping up with the changes, working with the files. The Basics-Introduction, Installation, getting started -hello world program, Lifecycle of Components, Understanding Functional & Class Components Passing Data. **[10 Hours]**

UNIT-II: React Components and Redux

React Props, React state-setting state, Event handling, Designing components state vs props. An Introduction to Redux - Core Concepts, Reducer, Action, Action Creator, Combining Reducers, Store, Data Flow in Redux, Usage with React. **[11 Hours]**

UNIT-III:Programming in Node.js

Node.js Installation –getting started, Control flow, asynchronous pattern callback, Sequential functionality, nested callbacks and exception handling, asynchronous patterns and control flow. Routing Traffic, Serving Files and Middleware: Building a Simple Static File Server from Scratch, Middleware, Routers and Proxies. **[11 Hours]**

UNIT-IV: Expressing REST APIs

REST-HTTP Methods as actions, Express-Routing, Handler Functions, The List API-automatic Server Restart, testing, Create API, Error Handling. **[10 Hours]**

UNIT-V:Mongo DB

Introduction to MongoDB: -Installation-Databases, Data Types, Using MongoDB Shell. Creating, Updating, Deleting and Querying Documents: Inserting, removing, and updating the documents. Scheme Initialization, Reading and writing to Mongodb. **[10 Hours]**

Text Books:

1. Tomasz DylKamilPrzeorski, “Mastering Full-Stack React Web Development”, 2017 Packt Publishing.
2. Vasan Subramanian, “ProMERN Stack”, Apress,2018.

Reference Books:

1. Eddy Wilson IriarteKoroliova ,“MERN-Full stack Development”, Packt Publishing Ltd.,2018
2. ShamaHoque,“Full stack React Projects”, Pack Publishing Ltd.,2018.