Centre for Distance and Online Education (VTU-CDOE)

Master of Computer Applications

Scheme and Syllabus

Program Outcomes		
Sl. No	**Description**	**POs**
1	Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.	**PO1**
2	Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.	**PO2**
3	Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.	**PO3**
4	Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.	**PO4**
5	Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.	**PO5**
6	Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices.	**PO6**
7	Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.	**PO7**
8	Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.	**PO8**
9	Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.	**PO9**
10	Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant top professional computing practices.	**PO10**

11	Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.	**PO11**
12	Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.	**PO12**

Program Education Objectives (PEOs):

PEO 1: promote from current position to software architecture / administration.

PEO2: develop products using automation

PEO 3: demonstrate high moral professional ethics

PEO 4: exhibit lifelong adoption for change in technology.

SEMESTER-I

Sl. No	Code	Course Name	Type of the Course	Credits
1	OMCA101	Mathematical Foundation for Computer Application	CORE	4
2	OMCA102	Operating System	CORE	4
3	OMCA103	Database Management System	CORE/SKILL	4
4	OMCA104	Programming Using C	SKILL	4
6	OMCA105	C Programming Lab	SKILL	2
7	OMCA106	Database Management Lab	SKILL	2
Total Credits			**20**	

SEMESTER-II

Sl. No	Code	Course Name	Type of the Course	Credits
1	OMCA201	Data Structure and Algorithms	CORE	4
2	OMCA202	Object Oriented Programming Using Python	CORE / SKILL	4
3	OMCA203	Software Engineering	CORE / SKILL	4
4	OMCA204	Computer Networks	SKILL	4
6	OMCA205	Data Structure Lab	SKILL	2
7	OMCA206	Python Lab	SKILL	2
Total Credits			**20**	

SEMESTER-III

Sl. No	Code	Course Name	Credit
1	OMCA301	Web Programming	4
2	OMCA302	Object Oriented Programming Using Java	4
3	OMCA303	Analysis & Design of Algorithm	3
4	OMCA304x	Elective - I	3
5	OMCA305	Web Programming Lab	3
6	OMCA306	Programming using Java Lab	3
TOTAL			20

Elective - I		
Sl. No.	**Course Code**	**Course Name**
1	OMCA304A	Data Analytics Using Python
2	OMCA304B	Introduction to Data Mining
3	OMCA304C	Cryptography and Network Security

SEMESTER-IV

Sl. No	Code	Course Name	Credit
1	OMCA401x	Elective - II	3
2	OMCA402x	Elective - III	3
3	OMCA403	Major Project	14

Elective - II		
Sl. No.	**Course Code**	**Course Name**
1	OMCA401A	Cloud Computing
2	OMCA401B	Big Data Analytics
3	OMCA401C	Cyber Security Governance, Risk & Compliance

Elective - III		
Sl. No.	**Course Code**	**Course Name**
1	OMCA402A	Artificial Intelligence
2	OMCA402B	Block Chain Technology
3	OMCA402C	Machine Learning

Mathematical Foundation for Computer Applications		Semester	I
Course Code	OMCA101	CIE Marks	30
		SEE Marks	70
Credits	4	Total Marks	100
		Exam Hours	3
Examination nature (SEE)	Theory		

Course objectives:

- The Curriculum supports the prerequisites to enhance their Mathematical knowledge towards their understanding mathematical Concepts in the concerned fields.

MODULE-1

MATRICES Definition, Types of Matrices, Addition, Subtraction, Scalar Multiplication and Multiplication of Matrices, Adjoint, Inverse, Eigen values and Eigen Vectors of a Matrix, Caley-Hamilton Theorem (Statement only) Rank of a matrix, Row reduced echelon form and normal form Solution of homogeneous and non homogeneous system of linear equations.

MODULE-2

SETS: Sets, Subsets, Types of Sets, Operation on Sets, Cartesian product, Cardinality of sets and applications.

MODULE-3

RELATIONS AND FUNCTIONS:

RELATIONS: Definition with illustrations, Representation of relations to Zero-one matrix and digraphs.

FUNCTIONS: Definition, Domain and Range of function, Types of functions with illustrations.

MODULE-4

Random variable and probability distribution:

Concept of random variable, discrete probability distributions, continuous probability distributions, Mean, variance and co-variance and co-variance of random variables. Binomial and normal distribution, Exponential and normal distribution with mean and variables and problems

MODULE-5

Graph Theory:

Graphs and Graphs models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Colouring

Course Outcomes:

CO1: Develop basic knowledge of matrices and to solve system of linear equations.
CO2: Understand the basic concepts of sets, functions and relations.
CO3: Understand the concepts of representations of relations and functions.
CO4: Model the given problem by applying the concepts of graph theory.
CO5: Design strategy using gaming theory concepts for the given problem.

Suggested Learning Resources:

Text Books & Reference Books:

1. Discrete Mathematics by Guru Raja Chaar.
2. B.S.Grewal: Higher Engineering Mathematics Khanna Publishers, 43rd Edition.
3. Richard A Johnson and C.B Gupta "Probability and statistics for engineers" Pearson Education
4. Kenneth H Rosen, "Discrete Mathematics and its Applications", McGraw Hill publications, 7th edition.

OPERATING SYSTEMS		Semester	I
Course Code:	OMCA102	CIE+SEE Marks	30 +70=100
Credits	04	Exam Hours	03
Examination type (SEE)		Theory	

Course Objectives:

CLO 1. Explore the need for OS and different types of OS

CLO 2. Explain the different techniques for management of resources

CLO 3. Learn the Use of processor, memory, storage and file system commands

Module-1

Introduction to operating systems [OS]: What operating systems do; Computer System organization; Computer System architecture; Operating System operations; Resource Management; Security and Protection; Virtualization; Distributed system; Computing environments.

Operating System Structures: Operating Services; User and Operating System interface; System calls; System Services; System programs; Operating system design and implementation; Operating System structure; System Building and Booting; Why Applications Are Operating-System Specific?

Process Management: Process concept; Process scheduling; Operations on processes; Inter process communication [IPC]; IPC Systems

Textbook 1: Chapter – 1, 2 and 3

Module-2

Threads and Concurrency: Multicore Programming, Multithreading models; Thread Libraries; Implicit Threading; Threading issues; OS-Threading examples.

CPU Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling; Multi-Processor Scheduling, Real-Time CPU Scheduling; OS CPU scheduling examples and Algorithm Evaluation

Process Synchronization: Background; The critical section problem; Peterson's solution; Hardware Support for Synchronization; Mutex Locks; Semaphores; Monitors; Classical problems of synchronization.

Textbook 1: Chapter – 4, 5, 6 and 7

Module-3
Deadlocks: System model; Deadlock in Multithreaded Applications; Deadlock characterization; Methods for handling deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock detection and Recovery from Deadlock. **Memory Management:** Background; Contiguous memory allocation; Paging; Structure of page table; swapping; Example: Intel 32- and 64-bit Architectures. **Virtual Memory Management:** Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **Textbook 1: Chapter – 7, 8, 9and 10**
Module-4
Mass-Storage Structure: Overview of Mass-Storage Structure; HDD Scheduling; NVM Scheduling; Storage Device Management; Swap-Space Management; Storage Attachment; RAID Structure. **File System:** File concept; Access methods; Directory structure; Protection; File system structure; File system operation; Directory implementation; Allocation methods; Free space management. File system mounting; File sharing. **Textbook 1: Chapter – 11, 12 and 13**
Module-5
Protection: Goals of protection, Principles of protection, Protection Rings; Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability- Based systems. **Case Study: The Linux Operating System:** Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory Management; File systems, Input and output; Inter-process communication; Network Structure; Security **Textbook 1: Chapter – 17 and 20**

Course Outcomes (Course Skill Set)

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

CO 1. Identify the structure of an operating system and its scheduling mechanism.

CO 2. Demonstrate the allocation of resources for a process using scheduling algorithm.

CO 3. Interpret the root causes of deadlock and provide the solution for deadlock elimination

CO 4. Illustrate different memory management concepts and storage structures such as files , directories and functionalities provided in the Linux Operating system.

Suggested Learning Resources:

Textbooks & Reference Books

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 10th edition, Wiley-India, 2018

2. D.M Dhamdhere, Operating Systems: A Concept Based Approach 3rd Ed, McGraw-Hill, 2013.

3. William Stallings Operating Systems: Internals and Design Principles, 9th Edition, Pearson.

4. Andrew S.Tanenbaum, "Modern operating Systems", fourth Edition, PHI Learning Pvt.Ltd., 2008

Database Management System		Semester	I
Course Code	OMCA103	CIE +SEE Marks	30 +70 =100
Credits	04	Exam Hours	3
Examination type (SEE)		Theory	

Course Objectives:

CLO 1. Practice SQL programming through a variety of database problems.
CLO 2. Explore the use of concurrency and transactions in database.
CLO 3. Build database applications for real world problems.

MODULE 1

Introduction: Characteristics of Database approach, Actors on the Scene, Workers behind the scene, Advantages of using DBMS approach, Data models, schemas and instances, Three -schema architecture and data independence, Database languages and interfaces, the database system environment, Centralized and client -server architectures, Classification of Database Management systems, Entity-Relationship Model: Conceptual Database using high level conceptual data models for Database Design, A Sample Database Application, Entity types, Entity sets Attributes and Keys Relationship types, Relationship Sets, Roles and Structural Constraints Weak Entity Types.

MODULE 2

Relational Model Relational Model and Relational Algebra: Relational Model Concepts, Relational Model Constraints and Relational Database Schema Update Operations, Transactions and Dealing with Constraint violations, Unary Relational operations, Relational Algebra Operations from Set Theory, Binary Relational Operations, JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra Relational Database Design Using ER-to Relational Mapping .

MODULE 3

Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic structure of SQL Queries, Additional Basic Operations, Null values, Aggregate Functions, nested Sub queries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

MODULE 4

Database Design: Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce Codd Normal Forms, Stored Procedures and functions, Triggers.

MODULE 5
Transaction Management: Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels. Concurrency Control: Lock Based Protocols, Deadlock Handling. Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm.

Course outcome (Course Skill Set):

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

CO1: Figure out the concepts of database objects, enforce integrity constraints on a database using RDBMS.

CO2: Demonstrate Structured Query Language (SQL) for database manipulation and also the basic of query evaluation.

CO3: Develop application to interact with databases, relational algebra expression,

CO4: Construct an application using tuple and domain relation expression from queries.

Recommended Text and Reference Books:

Text Books:

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.

2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.

3. Abraham Silberschatz, Henry F. Korth and S. Sudarshan"s Database System Concepts 9th EditionTata Mcgraw Hill Education Private Limited-2013

4. Introduction to Database Management System ,Satinder bal Gupta,Aditiya Mittal,2nd Edition,An imprint of Laxmi publications Private Limited-2017

Programming Using C		Semester	I
Course Code	OMCA104	CIE + SEE Marks	30 + 70 = 100
Credits	4	Exam Hours	03
Examination type (SEE)	Theory		

Course objectives:

CLO1 : Explain user-defined data structures like arrays, structures,/unions and pointers in implementing solutions to problems
CLO2: Design and Develop Solutions to problems using modular programming constructs such as functions and procedures.

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain the functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyse information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.

Module-1

Introduction to C: Structure of C Program, Compiling and executing C programs, Variables, Constants, IO statements in C, Operators in C

Module-2

Decision control and Looping statements: Introduction to decision control, Conditional branching statements, iterative statements, nested loops, break and continue statements, goto statement.

Module-3
Arrays: Declaration of arrays, accessing and storing of values in array, Operations on arrays, 2-D arrays, operations on two-dimensional arrays, multidimensional arrays, applications of arrays **Functions**: Introduction using functions, Function definition & declaration, function call, return statement, passing parameters to functions, Passing arrays to functions, scope of variables, storage classes, recursive functions.
Module-4
Strings: Introduction to strings, operations on strings, arrays of strings. **Pointers**: Introduction to pointers, declaring pointer variables, Types of pointers, Passing arguments to functions using pointers.
Module-5
Structure and Union: Introduction, structures and functions, Unions, unions inside structures. **Files**: Introduction to files, Operation of Files.
Course outcome (Course Skill Set) At the end of the course the student will be able to: **CO1**: Illustrate the fundamental programming constructs of C programming language to solve problem. **CO2**: Interpret the Use of functions and arrays in implementing solutions. **CO3**: Demonstrate the use of structures, unions and pointers to solve problems.
Suggested Learning Resources: Books 1. Computer Fundamentals and Programming in C – Reema Thareja, 2nd Edition, Oxford University, 2017. 2. E. Balaguruswamy, Programming in ANSI C, 7th Edition, Tata McGraw-Hill 3. Brian W. Kernighan and Dennis M. Ritchie, The 'C' Programming Language, Prentice Hall of IndiaYashavanth Kanetkar, Let us C, Authentic Guide to C Programming Langauge, bpb publisher, 17th Edition, 2020 4. Yashavanth Kanetkar, Let us C, Authentic Guide to C Programming Langauge, bpb publisher, 17th Edition, 2020
Web links and Video Lectures (e-Resources):
• elearning.vtu.ac.in/econtent/courses/video/BS/15PCD23.html • https://nptel.ac.in/courses/106/105/106105171/ MOOC courses can be adopted for more clarity in und

C Programming Lab		Semester	I
Course Code	OMCA105	CIE + SEE Marks	30 +70 =100
Credits	02	Exam Hours	03
Examination type (SEE)	Practical		

Course objectives:

CLO1 : Exploring an programs using constructs of C programming language
CLO2: Demonstrate the use of IDE, C Compiler, and identify and rectify the syntax and syntactic errors during programming.
CLO3: Learn to Reporting the observations and debug the program.

Laboratory Experiments:

Implement the following programs with WINDOWS / LINUX platform using appropriate C compiler

1. Implement a C program that takes three coefficients (a, b, and c) of a Quadratic equation ($ax^2+bx+c=0$) as input and compute all possible roots, output the roots with appropriate messages.
2. Write a C program to simulate a Simple Calculator using Switch case construct.
3. Develop a C Program to check whether a given number is PALINDROME or NOT. Ex: Num: 1221, Reverse: 1221, It is a Palindrome
4. Design and develop a C program to read a year as an input and find whether it is leap year or not.
5. Develop a C Program to search a Name in a list of names using Binary searching Technique (Use strcmp built-in function).
6. Write a C program that reads N integer numbers and arrange them in ascending order using Bubble Sort.
7. Develop, implement and execute a C program that reads two matrices A (m x n) and B (p x q) and Compute product of matrices A and B. Print both the input matrices and resultant matrix with suitable headings and output should be in matrix format only. Program must check the compatibility of orders of the matrices for multiplication. Report appropriate message in case of incompatibility.
8. Design and develop a C function isprime(num) that accepts an integer argument and returns 1 if the argument is prime, a 0 otherwise. Write a C program that invokes this function to generate prime numbers between the given range.
9. write a recursive C function to find the factorial of a number, n!, defined by fact(n)=1, if n=0. Otherwise fact(n)=n*fact(n-1). Using this function, write a C program to compute the binomial coefficient nCr . Tabulate the results for different values of n and r with suitable messages.
10. Write a C program to copy the contents of one file to another.
11. Write a C program that uses functions and structures to perform the following operations:

a. Reading a complex number
b. Displaying a complex number
c. Addition of two complex numbers
d. Multiplication of two complex numbers
Display the appropriate output.

12. Write a Program in c to swap two number using pointer.

Course outcome (Course Skill Set)

At the end of the course the student will be able to:
CO1: Make use of IDE for programming, identify and correct the syntax and syntactic errors using various programming constructs.
CO2: Demonstrate use of functions, recursive functions, arrays, strings, structures and pointers in problem solving.
CO3: Design and development of C programs to implement different searching and sorting techniques.

Suggested Learning Resources:

Books

1. Computer Fundamentals and Programming in C – Reema Thareja, 2nd Edition, Oxford Univeristy, 2017
2. E. Balaguruswamy, Programming in ANSI C, 7th Edition, Tata McGraw-Hill
3. Brian W. Kernighan and Dennis M. Ritchie, The 'C' Programming Language, Prentice Hall of India
4. Yashavanth Kanetkar, Let us C, Authentic Guide to C Programming Langauge, bpb publisher, 17th Edition, 2020

Web links and Video Lectures (e-Resources):

- elearning.vtu.ac.in/econtent/courses/video/BS/14CPL16.html
- https://nptel.ac.in/courses/106/105/106105171/

Database Management Lab		Semester	I
Course Code	OMCA106	CIE + SIE Marks	30 +70 = 100
Credits	02	Exam Hours	03
Examination type (SEE)	Practical		

Course objectives:

CLO1: Create SQL queries for the small projects.

CLO2: Create database objects that include tables, constraints, indexes, and sequences.

1. Students should be allowed to choose appropriate DBMS software, install it, configure it and start working on it. Create sample tables, execute some queries, use SQLPLUS features, Use PL/SQL features like cursors on sample database. Students should be permitted to practice appropriate User interface creation tool and Report generation tool.

2. A college consists of number of employees working in different departments. In this context, create two tables' employee and department. Employee consists of columns empno, empname, basic, hra, da, deductions, gross, net, date-of-birth. The calculation of hra,da are as per the rules of the college. 1. Create tables department and employee with required constraints. 2. Initially only the few columns (essential) are to be added. Add the remaining columns separately by using appropriate SQL command 3. Basic column should not be null 4. Add constraint that basic should not be less than 5000. 5. Calculate hra,da,gross and net by using PL/SQL program.

3. Students may be divided into batches and the following experiments may be given to them to better understand the DBMS concepts. Students should gather the required information, draw ER diagrams, map them to tables, normalize, create tables, triggers, procedures, execute queries, create user interfaces, and generate reports.

- Student information system
- KSRTC reservation system
- Hostel management
- Library management
- Indian Railways reservation

Course outcomes (Course Skill Set):

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

CO1: Design entity-relationship diagrams to solve given database applications.

CO2: Implement a database schema for a given problem.

CO3: Formulate SQL queries in Oracle for the given problem.

CO4: Design and Develop suitable database and verify for its appropriate normalization for any given problem.

Data Structures and Algorithms		Semester	II
Course Code	OMCA201	CIE + SIE Marks	30 + 70 =100
Credits	04	Exam Hours	03
Examination type (SEE)	Theory		

Course Learning objectives:

CLO1: Explore step by step and develop algorithms to solve real world problems.

CLO2: Evaluate the Expressions like postfix, prefix conversions.

CLO3: Implementing various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.

CLO4: Define various searching & sorting techniques.

CLO5: Compare functions using asymptotic analysis and describe the relative merits of worst-, average-, and best-case analysis.

Module-1	
Classification of Data Structures: Primitive and Non- Primitive, Linear and Nonlinear; Data structure Operations, Stack: Definition, Representation, Operations and Applications: Polish and reverse polish expressions, Infix to postfix conversion, evaluation of postfix expression, infix to prefix, postfix to infix conversion.	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
Module-2	
Recursion - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi. Queue: Definition, Representation, Queue Variants: Circular Queue, Priority Queue, Double Ended Queue; Applications of Queues. Programming Examples.	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
Module-3	
Linked List: Limitations of array implementation, Memory Management: Static (Stack) and Dynamic (Heap) Memory Allocation, Memory management functions. Definition, Representation, Operations: getnode() and Freenode() operations, Types: Singly Linked List. Linked list as a data Structure, Inserting and removing nodes from a list, Linked implementations of stacks, Header nodes, Array implementation of lists.	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation

Module-4	
Trees: Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression, Programming Examples.	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
Module-5	
Graphs: Definitions, Terminologies, Matrix and Adjacency List Representation Of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search. Insertion Sort,. Hash Table organizations, Hashing Functions, Static and Dynamic Hashing.	
Teaching Learning Process	Chalk and talk method / PowerPoint Presentation

Course outcome (Course Skill Set):

At the end of the course the student will be able to:
CO1: Illustrate the different data structures and operations.
CO2: Demonstrate the concept of stack and Queue data structures use
CO3: Infer the concept of Linked list, Trees and Graphs in problem solving
CO 4: Employ various data structures for solving various problems.

Suggested Learning Resources: Reference books:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
2. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning,2014.
3. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
4. Introduction to Algorithms ,Thomas h.Cormen Charles E. Leiserson Ronald L. Rivest Clifford Stein,4th Edition,2022.MIT Press

Web links and Video Lectures (e-Resources):

- https://www.youtube.com/watch?v=BBpAmxU_NQo
- https://www.youtube.com/watch?v=8hly31xKli0
- https://archive.nptel.ac.in/courses/106/106/106106127/

Object Oriented Programming Using Python		Semester	II
Course Code	OMCA202	CIE + SIE Marks	30 + 70 =100
Credits	4	Exam Hours	3
Examination type (SEE)	Theory		

Course objectives:

CLO1: Study the importance of Object Oriented Programming
CLO2: Explore the Object Oriented Programming concepts
CLO3: Explain the concept of Polymorphism, Inheritance
CLO4: Understand the creation of modules ,packages and organization of modules and packages

Teaching-Learning Process (General Instructions)

Programming Exercises and mini project works.

Module-1

Python Basic Concepts and Programming

Parts of Python Programming Language, Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, The type( ) Function and Is Operator, Control Flow Statements, The if Decision Control Flow Statement, The if…else Decision Control Flow Statement, The if…elif…else Decision Control Statement, Nested if Statement, The while Loop, The for Loop, The continue and break Statements, Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, The return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

Module-2

Python Collection Objects, Strings- Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings, Lists-Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods. Sets, Tuples and Dictionaries. Files: reading and writing files

Module-3

Object-oriented Design :Introducing object-oriented ,Objects and classes, Specifying attributes and behaviours : Data describes objects , Behaviours are actions ,Hiding details and creating the public interface: Composition, Inheritance: Inheritance provides abstraction, Multiple inheritance Objects in Python : Creating Python classes , Adding attributes, Making a function work: passing arguments, Initializing the object, self argument

Module-4
Modules and packages: Organizing the modules, Absolute imports, Relative imports, Organizing module contents: Access control, Third-party libraries, Basic inheritance, Extending built-ins, Overriding and super, Multiple inheritance, The diamond problem, Different sets of arguments,
Module-5
Polymorphism , Abstract base classes, Using an abstract base class , Creating an abstract base class Exceptions: Raising exceptions, The effects of an exception , Handling exceptions, The exception hierarchy, Defining our own exceptions
Course outcome (Course Skill Set) At the end of the course the student will be able to: **CO1**: Demonstrate proficiency in handling loops and creation of functions **CO2**: Illustrate the methods to create and manipulate lists, tuples and dictionaries . CO3: Design and Develop programs for string processing and file organization. **CO4**: Interpret the concept of OOP as used in Python
Suggested Learning Resources: **Books** 1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd edition,Updated for Python 3, Shroff/O'Reilly Publishers, 2016 2. Python 3 Object Oriented Programming, 2nd Edition, Unleash the power of Python 3 Objects by Dusty Phillips , PACKT Publishing. 3. Python Object–Oriented Programming :Build robust and maintainable Object-oriented python applications and libraries, Steven F. Lott, Dusty Philips,4th Edition, Packt Publishing Limited; 2021 4. Python the complete reference ,Martin C. Brown,4th Edition, McGraw Hill Education ,2018
Web links and Video Lectures (e-Resources):
• http://greenteapress.com/wp/thinkpython/

Software Engineering		**Semester**	**II**
Course Code	**OMCA203**	**CIE Marks**	**30**
		SEE Marks	**70**
Credits	**4**	**Total Marks**	**100**
		Exam Hours	**3**
Examination nature (SEE)	**Theory**		

Course objectives:

- Use modern tool to create dynamic diagrams to represent the design for the given problem.
- Draw class diagram, analyse the different types of association that exists as per the given problem and represent them using UML notations.
- Analyse the given system to identify actors, use cases to design use case diagrams for the given problem using RSA/open source tool.
- Design the static/dynamic models to meet application requirements of the given system and generate code (skeleton) using the modern tool.

MODULE-1

Introduction: Software Products and Software process, Process models: Waterfall modal, Evolutionary Development, Bohemia's Spiral model, Overview of risk management, Process Visibility, Professional responsibility. Computer based System Engineering: Systems and their environment, System Procurement, System Engineering Process, System architecture modelling. Human Factors, System reliability Engineering.

MODULE-2

Requirements and Specification: The requirement Engineering Process, The Software requirement document, Validation of Evolution of requirements, Viewpoint – oriented & method based analysis, system contexts, Social 7 organizational factors. Data flow, Semantic, Objects, models, Requirement Specification, Non functional requirement.

MODULE-3

Software Prototyping: Prototyping in software process, Prototyping techniques, User interface prototyping. Software Design: Design Process, Design Strategies, Design Quality, System Structuring control models, Modular decomposition, Domain Specific architecture.

MODULE-4
Object Oriented& function oriented design: Objects, object Classes and inheritance Object identification, An object oriented design example, Concurrent Objects, Data flow design Structural decomposition, Detailed Design, A Comparison of design Strategies. User interface design: Design Principles, User System interaction, Information Presentation, User Guidance, Interface Evaluation.
MODULE-5
Software Verification and Validation : The testing Process , Test Planning & Strategies, Black Box , Structural, interface testing , Program inspections , Mathematically based verification, Static analysis tools, Clean room software development. Management Issues: Project management, Quality management, Software cost estimation, Software maintenance.
Course outcomes: **CO 1: Describe** a software system, component, or process to meet desired needs within realistic constraints. **CO 2**: Compare professional and ethical responsibility' **CO 3:** Apply the techniques, skills, and modern engineering tools necessary for engineering practice, design, implement, verify, validate, implement, and maintain software systems or parts of software systems **Suggested Learning Resources:** **Books** 1. Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012. 2. Michael Blaha, James Rumbaugh: Object Oriented Modelling and Design with UML,2nd Edition, Pearson Education,2005. 3. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill. 4. Stephan R. Schach, "Object oriented software engineering", Tata McGrawHill,2008

Computer Networks		Semester	II
Course Code	OMCA204	CIE Marks	30
		SEE Marks	70
Credits	4	Total Marks	100
		Exam Hours	3
Examination nature (SEE)	Theory		

Course objectives:

- Comprehend the transmission technique of digital data between two or more computers and a computer network that allows computers to exchange data.
- Explain with the basics of data communication and various types of computer networks;
- Demonstrate Medium Access Control protocols for reliable and noisy channels.
- Expose wireless and wired LANs.

MODULE-1

Definition and concept of networking transmission modes. Transmission media, Internet working, Connecting devices, Adapters. Routers, evolution of Network Technology, Standards and protocols, Introduction to Analog signals, Digital signal, Modulation and Demodulation, OSI Reference Model-Layered structure, function of each layer, protocol used

MODULE-2

Switching-Message. Packet, and Circuit Switching, Multiplexing – FDM, TDM WDM, SONNET, Cellular network, satellite network, IEEE 802 STANDARDSCSMA/CD, TOKEN BUS, TOKEN RING, FDDI. Routing algorithms — Distance Vector routing, Link state routing, TCP/IP- Overview. Architectures, functions of each layers and protocol, IP addressing, subnet and subnet mask, IP addressing-classes, IPV4 IPV6.

MODULE-3

Bootstrap protocol, DHCP, mobile IP, DNS, Telnet, SMTP HTTP. SNMP, FTP. ATM network, ATM Architecture, BISND reference model. ATM applications, Data link control — Line discipline, Flow control, Error control. Encryption — Convention Encryption, Conventional Encryption Model, Steganography, Classical Encryption Techniques, Simplified DES. Block Cipher Design Principles. Block Cipher Modes of Operation.

MODULE-4
Cryptography, Public key encryption and hash functions ,public key cryptography, principles of public key cryptosystems, The RSA algorithm, Message Authentication and Hash functions, Authentication Requirements, Authentication Functions, Message Authentication Codes, MAC Algorithm, Hash Function algorithms, Secure Hash Algorithm, IP Security
MODULE-5
Network Security at various layers, Secure-HTTP. SSL, PSP, authentication Header, Key distribution protocols. Digital Signature, Digital Certificate, Security protocols, Levels of security. Virus and Worms related threats. Malicious programs, firewall. Design principles, Wifi, Bluetooth, Infrared.
Course Outcomes: CO 1: List the various components of data communication and transmission modes CO 2: Describe the fundamentals of digital communication and switching. CO 3: Explain data link layer protocols and network security at various layes. Suggested Learning Resources: Books 1. Data Communication‖ , 4 th Edition, Behrouz A. Forouzen, Tata McGraw Hill Education,2006 2. Computer Networks‖ , 5 th Edition, Andrew S. Tanenbaum, Pearson, 2011 3. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007. 4. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.

Data Structures Laboratory		Semester	II
Course Code	**OMCA205**	**CIE + SIE Marks**	**30 + 70 = 100**
Credits	**2**	**Exam Hours**	**03**
Examination type (SEE)	**Practical**		

Course objectives:

CLO1: Explain the Evaluation of Expressions like postfix, prefix conversions.

CLO2: Implementing various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.

Sl. No	Experiments
1	Implement a Program in C for converting an Infix Expression to Postfix Expression.
2	Design, develop, and execute a program in C to evaluate a valid postfix expression using stack. Assume that the postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide).
3	Design, develop, and execute a program in C to simulate the working of a queue of integers using an array. Provide the following operations: a. Insert b. Delete c. Display
4	Write a C program to simulate the working of a singly linked list providing the following operations: a. Display & Insert b. Delete from the beginning/end c. Delete a given element
5	Write a C program to Implement the following searching techniques a. Linear Search b. Binary Search.
6	Write a C program to implement the following sorting algorithms using user defined functions: a. Bubble sort (Ascending order) b. Selection sort (Descending order).
7	Write a C program to implement the Binary Search Tree operations.
8	Write a C program to demonstrate the Binary Tree Traversals - Inorder, postorder, preorder

Course outcomes (Course Skill Set):

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

CO1: Design and Develop Data structure techniques for evaluating the given expression.

CO2: Demonstrate various sorting / searching techniques and validate input/output for the given problem.

CO3: Design data structures to show the operations on Stacks, Queues, Circular Queues, Linked Lists, and Trees.

CO4: Implement the suitable algorithm to find whether the given graph is connected or not and illustrate the performance of the technique implemented.

Python Lab		Semester	II
Course Code	OMCA206	CIE + SIE Marks	30 + 70 =100
Credits	02	Exam Hours	03
Examination type (SEE)		Practical	

Course Learning Objectives:

CLO1: To be able to introduce core programming basics and program design with functions using Python programming language.

CLO2: To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.

CLO3: To understand the high-performance programs designed to strengthen the practical expertise.

1. Write a program to sum all the elements from n1 to n2 where n1 and n2 are positive integers
2. Input an array of n numbers and find separately the sum of positive numbers and negative numbers.
3. Write a program to search an element using linear search
4. Write a program to search an element using binary search.
5. Write a program to simulate stack.
6. Using a stack evaluate an arithmetic expression.
7. Write a program to multiply two matrices
8. Write a program to find the roots of a quadratic equation
9. Write a program to Insert a number in a sorted array.
10.Write a Python Program to check whether the given string is palindrome or not using built in string manipulation methods.
11. Write a Python Program to read a word and prints the number of letters, vowels and percentage of vowels in the word using dictionary
12. Write a Python Program to check a given sentence is a pangram or not using function/Module.

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

CO1: Demonstrate proficiency in handling loops and creation of functions.

CO2: Illustrate the methods to create and manipulate lists, tuples and dictionaries in Python Programme.

CO3: Design and Develop programs for string processing and file organization and use the concept of OOP as used in Python.

Web Programming		Semester	III
Course Code	OMCA301	CIE Marks	30
		SEE Marks	70
Credits	4	Total Marks	100
		Exam Hours	3
Examination nature (SEE)		Theory	

Course objectives:

- Explain advanced features of the web programming.
- Define the characteristics of HTML,XHTML,Java script,XML.
- Explore the basic principles of Web programming
- Enhance problem solving and programming skills in web programming with extensive programming projects.

MODULE-1

Fundamentals of Web: Internet, WWW, Web Browsers, and Web Servers, URLs, MIME, HTTP, Security, The Web Programmers Toolbox. XHTML: Origins and evolution of HTML and XHTML, Basic syntax, Standard XHTML document structure, Basic text markup, Images, Hypertext Links, Lists, Tables.

MODULE-2

HTML and XHTML: Forms, Frames in HTML and XHTML, Syntactic differences between HTML and XHTML. CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The Box model, Background images, The and tags, Conflict resolution.

MODULE-3

Java Script: Overview of JavaScript; Object orientation and JavaScript; General syntactic characteristics; Primitives, Operations, and expressions; Screen output and keyboard input; Control statements; Object creation and Modification; Arrays; Functions; Constructor; Pattern matching using expressions; Errors in scripts; Examples

MODULE-4

Java Script and HTML Documents: The JavaScript execution environment; The Document Object Model; Element access in JavaScript; Events and event handling; Handling events from the Body elements, Button elements, Text box and Password elements; The DOM 2 event model; The navigator object; DOM tree traversal and modification.

MODULE-5

Dynamic Documents with JavaScript: Introduction to dynamic documents; Positioning elements; Moving elements; Element visibility; Changing colors and fonts; Dynamic content; Stacking elements; Locating the mouse cursor; Reacting to a mouse click; Slow movement of elements; Dragging and dropping elements. XML: Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS; XSLT style sheets; XML Processors; Web services.

Course Outcomes:

CO 1: Discover HTML and CSS syntax and semantics to build web pages.

CO 2: Demonstrate format tables and forms using HTML and CSS

CO 3: Construct Client-Side Scripts using JavaScript and Server-Side Scripts using PHP to generate and display the contents dynamically.

Suggested Learning Resources:

Books

1. Robert W Sebesta, "Programming the World Wide Web", 4th Edition, Pearson Education, 2008.

2. Web Programming By Chris Bates , Wiley Publications

3. HTML5 Black Book by Dreamtech

4. Angular JS By Krishna Rungta

Object Oriented Programming Using Java		Semester	III
Course Code	OMCA302	CIE Marks	30
		SEE Marks	70
Credits	4	Total Marks	100
		Exam Hours	3
Examination nature (SEE)		Theory	

Course objectives:

- Understand the Java program structure, data types and statements.
- Learn the concepts of class, objects and methods using JDK tools.
- Explore concepts of inheritance, overloading and multi-threaded programming
- Explore the use

MODULE-1

History and features of java, C++ Vs java, how java works, JAVA Program Structure, Java Virtual Machine concepts, java platform overview, Primitive data types, variables and constants, operators, expression statement- branching, looping and jumping, labeled statements.

MODULE-2

Classes, objects and methods: defining a class, creating object, adding variables and methods, Constructor Instances, field and methods initialization by constructors, Types of constructor, memory allocation and garbage collection, access methods Arrays, String and String buffer classes.

MODULE-3

Inheritance, Super class Subclass, basic types, using super keyword, abstract and final classes, method overriding, dynamics method dispatch. Method overloading, Interface, Thread, Thread Life cycle, Multithreading examples, Synchronized threading, Priorities of thread.

MODULE-4

Exception handling: fundamental, exception types, uncaught exception, throws, throw, try-catch, finally, built in exception, creating your own exception, Packages, Built in Packages, Creating your own Package, input/output-basics streams, Byte and character streams.

MODULE-5

Applet programming-Local and Remote Applets, Applet Vs Application, creating and executing java applets, inserting applets in a web page, java security, passing parameter to applets, Aligning the Display, HTML Tags & Applet Tag, Getting Input from User.

Course Outcomes:

CO 1: Explore the object-oriented concepts and JAVA.

CO 2: Demonstrate programs to solve real world problems in Java.

CO 3: Construct simple GUI interfaces for a computer program to interact with users

Suggested Learning Resources:

Books

1. Programming with Java,6th Edition, E.Balaguruswamy, McGraw-Hill, 2019
2. Internet and Java Programming, 1 st Edition, Prabhu, R. Krishnamurthy, New Age International,2013
3. Java Fundamentals, A comprehensive Introduction by Herbert Schildt, Dale Skrien.Tata McGraw Hill Edition 2013. (Chapters:1,2,3,4,5,6, 7,8,9,10,11,12, 13,15,22,23,24, 25,26)
4. Java6 Programming,Black Book,KoGenT, Dreamtech Press,2012.

Analysis & Design of Algorithm		Semester	III
Course Code	OMCA303	CIE Marks	30
		SEE Marks	70
Credits	4	Total Marks	100
		Exam Hours	3
Examination nature (SEE)		Theory	

Course objectives:

- Explain various computational problem solving techniques.
- Apply appropriate method to solve a given problem.
- Describe various methods of algorithm analysis.

MODULE-1

Introduction: Algorithms, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental Data Structures. Fundamentals of the Analysis of Algorithm Efficiency: The Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-recursive and Recursive Algorithms, Empirical Analysis of Algorithms

MODULE-2

Brute Force Method: Selection Sort and Bubble Sort, Sequential Search, Brute-Force String Matching, Exhaustive Search, Depth-First Search and Breadth-First Search. Decrease and Conquer: Insertion Sort, Topological Sorting, Algorithms for Generating Combinatorial Objects, Decrease by-a-Constant-Factor Algorithms.

MODULE-3

Divide and Conquer: Merge Sort, Quick Sort, Binary Tree Traversals and Related Properties, Stassen's Matrix Multiplication. Space and Time Tradeoffs: Sorting by Counting, Input Enhancement in String Matching, Hashing. Dynamic programming: Binomial Coefficient, Principle of Optimality, Optimal Binary Search Trees, Knapsack Problem and Memory Functions, Warshall's and Floyd's Algorithms.

MODULE-4

Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees. Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP and NP Complete Problems.

MODULE-5

Coping with the Limitations of Algorithm Power: Back Tracking: n Queens problem, Hamiltonian Circuit Problem, Subset-Sum Problem. Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesman Problem.

Course Outcomes:

CO 1: Describe computational solution to well known problems like searching, sorting etc.

CO 2: Identify the computational complexity of different algorithms.

CO 3: Explain an algorithm using appropriate design strategies for problem solving

Suggested Learning Resources:

Books

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, Pearson, 2012
2. Horowitz, Sahni, Rajasekaran, "Fundamentals of Computer Algorithms", 2/e, Universities Press, 2007.
3. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.
4. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press

Data Analytics Using Python		Semester	III
Course Code	OMCA304A	CIE Marks	30
		SEE Marks	70
Credits	4	Total Marks	100
		Exam Hours	3
Examination nature (SEE)		Theory	

Course objectives:

- The objective of this course is to provide comprehensive knowledge of python programming paradigms required for Data Analytics.

MODULE-1

Revisiting Python: Strings- String Slicing and Joining, String Methods, Lists-Creating Lists, Indexing and Slicing in Lists, List Methods. Sets, Tuples and Dictionaries. Files: reading and writing files.
Loading from CSV files, Accessing SQL databases.

MODULE-2

USING NUMPY: Basics of NumPy-Computation on NumPy-Aggregations-Computation on Arrays Comparisons, Masks and Boolean Arrays-Fancy Indexing-Sorting Arrays-Structured Data: NumPy's Structured Array.

MODULE-3

DATA MANIPULATION WITH PANDAS: Introduction to Pandas Objects - Data indexing and Selection - Operating on Data in Pandas Handling Missing Data - Hierarchical Indexing - Combining Data Sets - Aggregation and Grouping - Pivot Tables.

MODULE-4

Web Scraping And Numerical Analysis Data Acquisition by Scraping web applications – Submitting a form - Fetching web pages – Downloading web pages through form submission – CSS Selectors.

MODULE-5

VISUALIZATION AND MATPLOTLIB Basic functions of matplotlib - Simple Line Plot, Scatter Plot - Density and Contour Plots Histograms, Binnings and Density - Customizing Plot Legends, Colour Bars – Three Dimensional Plotting in Matplotlib.

Textbooks / References:

[1] Jake VanderPlas, Python Data Science Handbook - Essential Tools for Working with Data, O'Reily Media Inc., 2016.
[2] Zhang.Y, An Introduction to Python and Computer Programming, Springer Publications, 2016.

References :

[1] Joel Grus , Data Science from Scratch First Principles with Python, O'Reilly Media, 2016.
[2] T.R.Padmanabhan, Programming with Python, Springer Publications, 2016.

Cos	Description
CO1	Demonstrate the use of built-in objects of Python
CO2	Demonstrate significant experience with python program development environment
CO3	Implement numerical programming, data handling and visualization through NumPy, Pandas and MatplotLib modules.

Course Outcomes

CO-PO Mapping:

PO/ PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	1		2		2	2	2			1
CO2	3	3	2	1	1		2	2	1	1	1	2
CO3	3	3	2	1	1	1	2	2	1	1	1	2
CO4	3	3	3	1	1	2	2	2	1	1	1	2
CO5	3	3	2	1	1	2	2	2	1	1	1	2

Course	Introduction to Data Mining	Semester	III
Course Code	OMCA304B	CIE Marks	30
Credits	04	SIE Marks	70
Examination nature (SEE)	Theory	Total Marks	100
		Exam Hours	03

Course objectives:

- Define multi-dimensional data models.
- Explain rules related to association, classification and clustering analysis.
- Compare and contrast between different classification and clustering algorithms

MODULE 1

Data warehousing and OLAP Data Warehouse basic concepts, Data Warehouse Modeling, Data Cube and OLAP : Characteristics of OLAP systems, Multidimensional view and Data cube, Data Cube Implementations, Data Cube operations, Implementation of OLAP and overview on OLAP Software.

MODULE 2

Data Mining and its Applications Introduction, What is Data Mining, Motivating Challenges, Data Mining Tasks, Which technologies are used for data mining, Kinds of pattern that can be mined, Data Mining Applications, Data Preprocessing, Data cleaning, data integration, data reduction and data transformation.

MODULE 3

Association Analysis: Basic Concepts and Algorithms Frequent Item set Generation, Rule Generation, Compact Representation of Frequent Item sets, Alternative methods for generating Frequent Item sets, FP Growth Algorithm, Evaluation of Association Patterns

MODULE 4

Classification : Methods, Improving accuracy of classification Basics, General approach to solve classification problem, Decision Trees, Rule Based Classifiers, Nearest Neighbor Classifiers. Bayesian Classifiers, Estimating Predictive accuracy of classification methods, Improving accuracy of classification methods, Evaluation criteria for classification methods, Multiclass Problem.

MODULE 5

Clustering Techniques Overview, Features of cluster analysis, Types of Data and Computing Distance, Types of Cluster Analysis Methods, Partition Methods, Hierarchical Methods, Density Based Methods, Quality and Validity of Cluster Analysis

Course outcomes:

CO 1: Identify data mining problems and implement the data warehouse
CO 2: Explain association rules for a given data pattern.
CO 3: Simulate between classification and clustering solution

Recommended Text and Reference Books:

1. Jiawei Han and MichelineKamber: Data Mining - Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publisher, 2006.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison- Wesley, 2005.
3. Arun K Pujari: Data Mining Techniques University Press, 2nd Edition, 2009.
4. G. K. Gupta: Introduction to Data Mining with Case Studies, 3rd Edition, PHI, New Delhi, 2009.
5. Alex Berson and Stephen J.Smith: Data Warehousing, Data Mining, and OLAP Computing McGrawHill Publisher, 1997.

Course	Cryptography and Network Security	Semester	III
Course Code	OMCA304C	CIE Marks	30
Credits	04	SIE Marks	70
Examination nature (SEE)	Theory	Total Marks	100
		Exam Hours	03

MODULE 1 :

Introduction to Cryptography

Introduction: OSI Security Architecture, Security Attacks, Security Services, Security Mechanism, model for Network Security.

Classical Encryption Technique: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques.

MODULE 2 :

Data Encryption and advanced encryption techniques

Block Ciphers, Data Encryption Standard and Advanced Encryption Standard

Block Cipher Principles, The Data Encryption Standard, Block Cipher Design Principles and Modes of operation, Evaluation Criteria for AES, AES Cipher-Encryption and Decryption, Data Structure, Encryption Round.

Public Key Cryptography and Key Management

Principles of Public Key Cryptosystem, RSA algorithm, Key management, Diffie Hellman Key exchange.

Module-3:

CRYPTOGRAPHY techniques

Message Authentication and Hash Function: Authentication Requirement, Authentication Functions, Message Authentication Code, Hash Functions, Digital Signatures, Digital Signature Standard.

Authentication Applications: Kerberos, X.509 Authentication Service

MODULE 4:

E-MAIL AND IP SECURITY

Electronic Mail Security: Pretty Good Privacy (PGP), S/MIME

IP Security: IP Security Overview;IP Security Architecture; Authentication Header; Encapsulating SecurityPayload; Combining Security Associations; Key Management

MODULE 5 :

WEB AND SYSTEM SECURITY

Web Security: Web security Considerations; Secure Socket layer (SSL) and Transport layer Security (TLS); Secure Electronic Transaction (SET).

System Security: Intruders, Intrusion Detection, Firewall Design PrinciplesCharacteristics, Types of Firewall and Firewall Configuration.

Text Books:

1. William Stallings, "Cryptography and Network Security – Principles and Practices", 4th Edition, Pearson Education, 2009. (Chapters: 1, 2.1-2.3, 3.1,3.2,3.5, 5.1,5.2, 6.2, 9.1,9.2, 10.1,10.2, 11.1-11.4, 13.1, 13.3, 14.1, 4.2, 15.1, 15.2, 16.1-16.6, 17.1-17.3, 18.1, 18.2, 20.1; Exclude the topic not mentioned in the syllabus)

Reference Books:

1. Behrouz A. Forouzan and DebdeepMukhopadhyay: "Cryptography and Network Security", 2nd Edition, Tata McGraw-Hill, 2010.
2. AtulKahate, "Cryptography and Network Security" 2nd Edition TMH.

Course Outcomes

Cos	Description
CO1	Identify common network security vulnerabilities/attacks
CO2	Understand the foundations of Cryptography and network security.
CO3	Understand encryption and decryption of messages using block ciphers
CO4	Demonstrate detailed knowledge of the role of encryption to protect data.
CO5	Analyze Network Security Practice And System Security.

CO-PO Mapping:

PO/PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	X						X					
CO2		X			X							
CO3			X		X							

Course	Web Lab	Semester	III
Course Code	OMCA305	CIE Marks	30
		SIE Marks	70
Credits	02	Total Marks	100
		Exam Hours	03
Examination nature (SEE)		Lab	

1. Create a Web page by making use of the following tags : Headers, Linking and Images.
2. Create a Web page that will have the following: Frames, Unordered Lists, Nested and ordered Lists
3. Create a Web page Layout with Tables and all its attributes
4. Create a Web page that will have Application form (Forms) , make use of Image Maps and Tags
5. Create an External Style Sheet that defines the style for the following tag : H1, H2, Body , P, Li .
6. Create an Internal Style Sheet that defines a style for Positioning elements & setting the background (color / image)
7. Create a Style Sheets that defines the style with class method , Id method , make use of DIV and Span TAG
8. Write a JavaScript program to Demonstrate the use of Variable , message box , and loops
9. Write a JavaScript Program to demonstrate Functions (predefined / user defined)
10. Write a JavaScript program to demonstrate Event Handling
11. Object Creation and modification in JavaScript
12. Write a PHP program to demonstrate GET and POST method of passing the data between pages

Course Outcomes:

CO 1: Illustrate HTML and CSS syntax and semantics to build web pages.
CO 2: Demonstrate format tables and forms using HTML and CSS
CO 3: Experiment Client-Side Scripts using JavaScript and Server-Side Scripts using PHP to generate and display the contents dynamically.

Course	Java Programming Lab	Semester	III
Course Code	OMCA306	CIE Marks	30
Credits	02	SIE Marks	70

1. a) Write a JAVA program to demonstrate Constructor Overloading and Method Overloading.
   b) Write a JAVA program to implement Inner class and demonstrate its Access protection

2. Write a program in Java for String handling which performs the following:

i) Checks the capacity of StringBuffer objects.

ii) Reverses the contents of a string given on console and converts the resultant string in upper case.

iii) Reads a string from console and appends it to the resultant string of (ii).

3. a). Write a JAVA program to demonstrate Inheritance. b). Simple Program on Java for the implementation of Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.

4. Write a JAVA program which has

i) A Class called Account that creates account with 500Rs minimum balance, a deposit()method to deposit amount, a withdraw() method to withdraw amount and also throws LessBalanceException if an account holder tries to withdraw money which makes the balance become less than 500Rs.

ii) A Class called LessBalanceException which returns the statement that says withdraw amount (Rs) is not valid.

iii) A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a LessBalanceException take appropriate action for the same.

5. Write a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.

6. Complete the following: 1. Create a package named shape. 2.Create some classes in the package representing some common shapes like Square, Triangle, and Circle. 3. Import and compile these classes in other program.
7. Write a JAVA program to create an enumeration Day of Week with seven values SUNDAY through SATURDAY. Add a method isWorkday( ) to the DayofWeek class that returns true if the value on which it is called is MONDAY through FRIDAY. For example, the call DayOfWeek.SUNDAY.isWorkDay ( ) returns false.
8. Write a JAVA program to print a chessboard pattern
9. Write a JAVA program which uses FileInputStream / FileOutPutStream Classes
10. Write JAVA programs which demonstrates utilities of LinkedList Class.
11. Write a JAVA program which uses Datagram Socket for Client Server Communication.
12. Write a JAVA applet program, which handles keyboard event.

Course	Cloud Computing	Semester	IV
Course Code	OMCA401A	CIE +SEE Marks	30 + 70 = 100
Credits	04	Exam Hours	03
Examination type (SEE)		Theory	

Course Objectives:

CLO 1. Explain the fundamentals of cloud computing

CLO 2. Illustrate the cloud application programming and aneka platform

CLO 3. Contrast different cloud platforms used in industry.

MODULE 1 :

Introduction to Cloud Computing: Eras of computing, The vision of Cloud Computing, Defining a cloud, A closer look, Cloud computing reference model, Historical developments: Distributed systems, Virtualization, Web 2.0; Service oriented computing; Utility oriented computing.

MODULE 2 :

Architectures for parallel and distributed computing: Parallel Vs Distributed computing, Elements of distributed computing, Technologies for distributed computing.

MODULE 3:

Virtualization: Introduction, Characteristics of virtualized environments, Taxonomy of virtualization techniques, Virtualization and cloud computing, Pros and cons of virtualization, Technology examples: Xen: Para virtualization, VmWare: Full virtualization, Microsoft Hyper – V.

MODULE 4:

Cloud computing architecture: Introduction, Cloud reference model: Architecture, IaaS, PaaS, SaaS, Types of Clouds: Public, Private, Hybrid and Community clouds, Economics of the cloud, Open challenges

MODULE 5 :

Cloud Platforms in Industry : Amazon web services; Google AppEngine; Microsoft Azure; Cloud Applications. Scientific applications: Healthcare; Biology; Geo-Science, Business and Consumer applications: ARM & ERP; Productivity; Social networking

Textbooks :

1. RjkumarBuyya, Christian Vecchiola, and ThamaraiSelci, Mastering Cloud Computing, Tata McGraw Hill, New Delhi, India, 2013

Reference Books

1. Cloud Computing for Dummies by Judith Hurwitz, R.Bloor, M. Kanfman, F.Halper (Wiley India Edition)
2. Cloud Computing: A Practical Approach by J.Vette, Toby J. Vette, Robert Elsenpeter (Tata McGraw Hill)

Course Outcomes

CO1 Demonstrate the fundamental and core concepts of cloud computing
CO2 Compare between parallel and distributed computing
CO3 Investigate the system virtualization and outline its role in enabling the cloud computing system model
CO4 Compare different deployment and service models of cloud to develop different variety of applications

CO-PO Mapping:

PO/ PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	X											
CO2	X	X			X					X		
CO3	X	X										
CO4	X	X	X									

Course	Big Data Analytics	Semester	IV
Course Code	OMCA401B	CIE +SEE Marks	30 + 70 = 100
Credits	04	Exam Hours	03
Examination Type (SEE)		Theory	

Course Objectives:

CLO 1. To provide an overview of an exciting growing field of big data analytics.

CLO 2. To introduce the tools required to manage and analyze big data like Hadoop, NoSQL, and Map-Reduce.

CLO 3. To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability.

CLO 4. To enable students to have skills that will help them to solve complex real-world problems for decision support

MODULE 1 :

Introduction to Big Data: What is big data, why big data, the convergence of key trends, unstructured data, industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data, credit risk management, big data and algorithmic trading, big data and healthcare, big data in medicine, advertising and big data, big data technologies, introduction to Hadoop, open-source technologies, cloud and big data, mobile business intelligence, Crowd-sourcing analytics, inter and trans firewall analytics.

MODULE 2 :

No SQL: Introduction to NoSQL, aggregate data models, aggregates, key-value and document data models, relationships, graph databases, schema less databases, materialized views, distribution models, sharding, masterslave replication, peer-peer replication, sharding and replication, consistency, relaxing consistency, version stamps, map-reduce, partitioning and combining, composing map-reduce calculations.

Module-3:

Hadoop: Data format, analyzing data with Hadoop, scaling out, Hadoop streaming, Hadoop pipes, design of Hadoop distributed file system (HDFS), HDFS concepts, Java interface, data flow, Hadoop I/O, data integrity, compression, serialization, Avro, file-based data structures.

MODULE 4:

MapReduce: MapReduce workflows, unit tests with MRUnit, test data and local tests, anatomy of MapReduce job run, classic Map-reduce, YARN, failures in classic Map-reduce and YARN, job scheduling, shuffle and sort, task execution, MapReduce types, input formats, output formats.

MODULE 5 :

Recent Trends in Big Data Analytics: HBase, data model and implementations, HBase clients, HBase examples, praxis. Cassandra, Cassandra data model, Cassandra examples, Cassandra clients, Hadoop integration, Hive, data types and file formats, HiveQL data definition, HiveQL data manipulation, HiveQL queries.

Textbooks / References:

1. Big Data Analytics, Introduction to Hadoop, Spark, and Machine-Learning, Raj Kamal, Preeti Saxena, McGraw Hill, 2018.
2. Big Data, Big Analytics: Emerging Business intelligence and Analytic Trends for Today's Business, Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, John Wiley & Sons, 2013.

Reference Books:

1. Business Intelligence and Analytic Trends for Today's Businesses", Wiley, 2013
2. Hadoop: The Definitive Guide, Tom White, Third Edition, O'Reilly, 2012.
3. Hadoop Operations, Eric Sammer, O'Reilly, 2012.
4. Programming Hive, E. Capriolo, D. Wampler, and J. Rutherglen, O'Reilly, 2012.
5. HBase: The Definitive Guide, Lars George, O'Reilly, 2011.
6. Cassandra: The Definitive Guide, Eben Hewitt, O'Reilly, 2010.
7. Programming Pig, Alan Gates, O'Reilly, 2011.

E-Books:

1.http://index-of.co.uk/Big-DataTechnologies/Data%20Science%20and%20Big%20Data%20Analytics.pdf

Course Outcomes

Cos	Description
CO1	Describe big data and use cases from selected business domains.
CO2	Explain NoSQL big data management.
CO3	Install, configure, and run Hadoop and HDFS.
CO4	Perform map-reduce analytics using Hadoop.
CO5	Use Hadoop related tools such as HBase, Cassandra, and Hive for big data Analytics, and understanding the recent trends in Big Data analytics.

CO-PO Mapping:

PO/PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1		1			1			1	1	1
CO2	2	1	2	1	1		1					
CO3	1	2	3	1	1							
CO4	1	1	3	1	1							
CO5	2	2	3	1	1							

Course	Cyber security Governance, Risk and Compliance	Semester	IV
Course Code	OMCA401C	CIE + SEE Marks	30 + 70 = 100
Credits	04	Exam Hours	03
Examination type (SEE)		Theory	

Course Objectives:

CLO 1. The students will learn the principles of cybersecurity governance, risk, and compliance

CLO 2. They will understand the tools methods, including vulnerability management, threat detection, metrics, and evaluations of organizations.

CLO 3. Students will study the NIST framework and learn organizational roles within a company

MODULE 1

Basics of Cyber security governance, Principles of cyber-security governance, Assessment of cyber security maturity, Theories of governance – introduction, Governance – definitions and typologies.

Governance of security operations, Tools, methods, and processes, Vulnerability management

MODULE 2

Threat management, Endpoint management, Intrusion detection and prevention (IDPS), Security incident management.

Security metrics and governance, Measurement of governance: Metrics – concepts, Application security metrics, Network security metrics

MODULE 3

Security incident metrics, Vulnerability metrics, Service level objectives/agreement (SLO / SLA), NIST metrics.

Security analytics and governance, Basics of security analytics, Threat intelligence and governance, Data-driven security governance

MODULE 4

Impact of cognitive security on security governance.

Compliance and governance, Industry-specific security compliance, Cyber security governance – Republic of India, NIST mandates for compliance, Security reporting basics, CISO – role and organization structure, HIPAA, COBITZ compliance.

MODULE 5

Cyber Security Risk:, Information security risk management framework and methodologies, Risk Management Process, Framework, and Life Cycle, Identifying and modeling information security risks, Qualitative and quantitative risk assessment methods, Articulating information security risks as business consequences

Textbooks / References:

1. Information Security Governance: A Practical Development and Implementation Approach, Wiley publications 2009.
2. Information Security Governance, S.H. Solms, Rossouw Solms, Springer Science & Business Media.
3. Internet governance in an age of cyber insecurity,2010, Council on Foreign Relations Press.
4. Cyber justice: human rights and good governance for the internet, 2017, Springer.
5. Cyber Risk Management: Prioritize Threats, Identify Vulnerabilities and Apply Controls 1st Edition, Kogan Publishers, 2019.

Course Outcomes

Cos	Description
CO1	Understand the different methods to assess cybersecurity maturity
CO2	Understand the vulnerability management techniques and threat management methodologies.
CO3	Understand the governance metrics (Application security, vulnerability, and network security).
CO4	Know the relation between security analytics and security governance.
CO5	Understand the NIST compliance for security mandate.

CO-PO Mapping:

PO/ PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2											
CO2	3	1										
CO3	3											
CO4	3	1			1	1						
CO5	2											

Course	Artificial Intelligence	Semester	IV
Course Code:	OMCA402A	CIE+SEE Marks	30 +70=100
Credits	04	Exam Hours	03
Examination type (SEE)		Theory	

Course Objectives:

CLO 1. Illustrate the reasoning on Uncertain Knowledge

CLO 2. Explore the explanation-based learning in solving AI problems

CLO 3. To explore advanced career opportunities

CLO 4. Demonstrate the applications of soft computing and Evolutionary Computing algorithms

Module-1

Artificial Intelligence – Basics, The AI Problems – The Underlying Assumption – What is an AI technique – Criteria for Success. Problems, Problem Spaces and Search – Defining Problem as a State Space Search – Production Systems – Problem Characteristics – Production System Characteristics – Issues in the design of Search Programs.

Module-2

Heuristic Search Techniques - Generate – and – Test – Hill Climbing – Best-First Search – Problem Reduction – Constraint Satisfaction - Means - Ends Analysis. Knowledge Representation issues – Representations and Mapping - Approaches to knowledge Representation

Module-3

Issues in knowledge Representation – The Frame Problem. Case study based on search algorithms.

Using Predicate Logic – Representing simple facts in Logic – Representing Instance and Isa Relationship – Computable Functions and Predicates – Resolution – Natural Deduction. Representing Knowledge Using Rules – Procedural versus Declarative knowledge.

Module-4

Logic Programming – Forward versus Backward Reasoning – Matching – Control Knowledge. Case study based on reasoning

Reasoning under Uncertainty – Introduction to Non-monotonic Reasoning – Augmenting a Problem Solver – Implementation: Depth - First Search, Fuzzy Logic.

Module-5

Game Playing - The Minimax Search Procedure – Adding Alpha-Beta Cut-offs. Applications of artificial intelligence- Case study on social networks using neural networks, DNA sequencing using AI techniques.

Textbooks / References:

1. Artificial Intelligence (Second Edition) – Elaine Rich, Kevin knight (Tata McGraw- Hill)
2. A Guide to Expert Systems – Donald A. Waterman (Addison-Wesley)
3. Principles of Artificial Intelligence – Nils J. Nilsson (Narosa Publishing House)
4. Introduction to Artificial Intelligence – Eugene Charnaik, Drew McDermott (Pearson Education Asia)

Course Outcomes

Cos	Description
CO1	To be aware of the basics of AI and its need along with the issuesin designing search problems.
CO2	Understand and apply various search algorithms in real world problems.
CO3	To get a thorough idea about the fundamentals of knowledge representation, inference and theorem proving.
CO4	Express and comprehend the working knowledge of reasoning in the presence of incomplete and/or uncertain information.
CO5	To gain the aptitude to apply knowledge representation and reasoning to real-world problems

CO-PO Mapping:

PO/ PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	1					2	1	1	2
CO2	2	1	1			1		2	2	1	1	2
CO3	3	2	1	1					2	1		2
CO4	1	2	1	1		1	1	2	1	1	1	2
CO5	2	1	1	1			1	2	1	1	1	2

Course	Block Chain Technology	Semester	IV
Course Code	OMCA402B	CIE + SIE Marks	30 + 70 = 100
Credits	04	Exam Hours	03
Examination type (SEE)		Theory	

Course Objectives:

CLO 1. Demonstrate the basics of Block chain concepts using modern tools/technologies.

CLO 2. Illustrate the role of block chain applications in different domains including cyber security

CLO 3. Evaluate the usage of Block chain implementation/features for the given problem.

CLO 4 : Exemplify the usage of bitcoins and its impact on the economy.

CLO 5 : Analyze the application of specific block chain architecture for a given problem

MODULE 1 :

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

MODULE 2 :

Block chain: Architecture, versions, variants, use cases, Life use cases of block chain, Block chain vs shared Database, Introduction to crypto currencies, Types, Applications.

Module-3:

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

MODULE 4:

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

MODULE 5 :

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

Books

- Beginning Block chain: A Beginner's Guide to Building Blockchain Solutions by ArshdeepBikramaditya Signal,
- Gautam Dhameja (Priyansu Sekhar Panda., A Press.) 2018
- Block chain Applications: A Hands-On Approach by Bahga, Vijay Madisetti ,2017
- Block chain by Melanie Swan, OReilly 2015

Reference Books

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

Web links and Video Lectures (e-Resources):

- https://youtu.be/mzPoUjQC4WU

Course Outcomes

Cos	Description
CO1	Demonstrate the basics of Block chain concepts using modern tools/technologies.
CO2	Analyze the role of block chain applications in different domains including cyber security.
CO3	Evaluate the usage of Block chain implementation/features for the given problem.
CO4	Exemplify the usage of bitcoins and its impact on the economy
CO5	Analyze the application of specific block chain architecture for a given problem

CO-PO Mapping:

PO/ PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2	X									X		
CO3			X					X				
CO4	X								X			
CO5			X									

Course	Machine Learning	Semester	IV
Course Code	OMCA402C	CIE + SIE Marks	30 + 70 = 100
Credits	04	Exam Hours	03
Examination type (SEE)		Theory	

Course objectives:

1. Explain the concept of supervised, unsupervised and semi-supervised learning.
2. Develop algorithms to learn linear and non- linear models using software.
3. Perform creative work in the field ML to solve given problem.

MODULE 1 :

Introduction to Machine learning: Supervised learning, Unsupervised learning, some basic concepts in machine learning, Review of probability, The log-sum-exp trick, Feature selection using mutual information, Linear Regression

MODULE 2 :

Computational Learning theory- Sample complexity, ε- exhausted version space, PAC learning, agnostic learner, VC dimensions, Sample complexity. Bayesian Learning, curse of dimensionality, over fitting. Parametric Estimators - estimator bias and variance, active learning

Module-3:

Dimensionality reduction, Clustering – choosing the number of clusters, Spectral clustering, Evaluating cluster quality. Margin and generalization (EM) algorithm, EM, regularization.

MODULE 4:

Non-parametric methods – KNN Linear discrimination - Support vector machine (SVM) and kernels, Classification errors, regularization, logistic regression.

MODULE 5 :

Model selection, Model selection criteria, Description length, feature selection, Combining classifiers, Bagging, boosting, Random Forest. Markov models, Hidden Markov models (HMMs), Bayesian networks, Learning Bayesian networks, Probabilistic inference, Current problems in machine learning.

Suggested Learning Resources: Books

1. Kevin P. Murphey, —Machine Learning, a probabilistic perspective‖, The MIT Press, 2012.
2. Tom Mitchael, —Machine Learning‖, McGraw Hill, 1997.
3. Ethem Alpaydin,‖Introduction to Machine learning‖, PHI learning, MIT Press, 2010, 2nd edition
4. John D. Killeher, Brian Mac, Namee, AoiFE D'Arcy, Fundamental of Machine Learning for Predictive Data Analytics, 2015 MITpress
5. Alex Smola and SVN. Viswanathan, —Introduction to Machine Learning‖, Cambridge University Press, 2008.

Course	Major Project	Semester	IV
Course Code	OMCA403	CIE Marks	30
Credits	12	SIE Marks	70

Course Objectives:

- To support independent learning.
- To guide to select and utilize adequate information from varied resources maintaining ethics.
- To guide to organize the work in the appropriate manner and present information (acknowledging the sources) clearly.
- To develop interactive, communication, organization, time management, and presentation skills.
- To impart flexibility and adaptability.
- To inspire independent and team working.
- To expand intellectual capacity, credibility, judgement, intuition.
- To adhere to punctuality, setting and meeting deadlines.
- To instill responsibilities to oneself and others.
- To train students to present the topic of project work in a seminar without any fear, face audience confidently, enhance communication skill, involve in group discussion to present and exchange ideas.

Project Work Phase :

- Students in consultation with the guide/s shall carry out literature survey/ visit industries to finalize the topic of the Project. Subsequently, the students shall collect the material required for the selected project, prepare synopsis and narrate the methodology to carry out the project work.
- Each student of the project batch shall involve in carrying out the project work jointly in constant consultation with internal guide, co-guide, and external guide and prepare the project report as per the norms avoiding plagiarism.
- Follow the Software Development life cycle
- Data Collection ,Planning
- Design the Test cases
- Validation and verification of attained results
- Significance of parameters w.r.t scientific quantified data.
- Publish the project work in reputed Journal

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

- Present the project and be able to defend it.
- Make links across different areas of knowledge and to generate, develop and evaluate ideas and information so as to apply these skills to the project task.
- Habituated to critical thinking and use problem solving skills
- Communicate effectively and to present ideas clearly and coherently in both the written and oral forms.
- Work in a team to achieve common goal.
- Learn on their own, reflect on their learning and take appropriate actions to improve it.

Master of Computer Applications (MCA) / MCA in Artificial Intelligence and Data Science/ MCA in Cyber Security and Cloud Computing

ASSESSMENT GUIDELINES (BOTH CIE AND SEE)

The weightage of Continuous Internal Evaluation (CIE) is 30% and for Semester End Exam (SEE) is 70%. The minimum passing marks for the CIE is 50% of the maximum marks. Minimum passing marks in SEE is 50% of the maximum marks of SEE. A student shall be deemed to have satisfied the academic requirements (passed) and earned the credits allotted to each course if the student secures not less than 50% in the sum total of the CIE and SEE taken together.

Continuous Internal Evaluation:

The CIE will be for 30 Marks. A candidate shall obtain not less than 50% of the maximum marks prescribed for the CIE. CIE Marks will be based on 30 objective type questions **(MCQ's, Fill in the blanks, one word answer, etc.) from all the Modules. Equal weightage should be given to all the modules.**

Semester End Examination:

The SEE question paper will be set for 70 marks & will have three sections

- **Section-A** consists of 20 objective type questions carrying 1 mark each. All questions are compulsory
- **Section-B** consists of 8 questions carrying 10 marks each. The students will have to answer 5 complete questions

PROGRAMME CO-ORDINATOR
COMPUTER APPLICATIONS
Visvesvaraya Technological University
Centre for Distance and Online Education
MYSURU-570 029

Visvesvaraya Technological University, Belagavi

Centre for Distance and Online Education (CDOE), Mysuru

VTU-CDOE MYSURU

Master of Computer Applications (MCA) /
MCA in Artificial Intelligence and Data Science/
MCA in Cyber Security and Cloud Computing

PROJECT WORK GUIDELINES

Project Work	Semester	IV	
Course Code	OMCA403 / OMCI403 / OMCC403	CIE Marks	30
Teaching Hours/Week (L: P: SDA)	0:4:0	SEE Marks	70
Credits	12	Exam Hours	03

Objective

To expose the students to understand the working of the organization/company/ industry and take up an in-depth study of an issue/problem in the area of specialization.

General guidelines

- Each candidate shall carry out the project work independently as per Scheme of Teaching and Evaluations under the guidance of one of the faculty members of the Department.
- If the project is of inter-disciplinary nature, a co-guide shall be allotted by the University from the other concerned department.
- The topic and title of the dissertation shall be chosen by the candidate in consultation with the guide and co-guide, if any, before the commencement of fourth semester.
- The subject and topic of the dissertation shall be from the major field of studies of the candidate. Modification of only the title but not the field of work may be permitted at the time of final submission of dissertation report during fourth semester.
- The Project Work and Dissertation preparation could be carried out by the students either in their work place/ institution/ any industry/ R&D labs/ business organizations.
- The candidate shall submit a soft copy of the dissertation work to the University.
- The soft copy shall contain the entire Dissertation on the project work in monolithic form as a PDF file (not separate chapters).

PROGRAMME CO-ORDINATOR
COMPUTER APPLICATIONS
Visvesvaraya Technological University
Centre for Distance and Online Education

- The Guide, after satisfying himself/herself on the suitability of the dissertation and checking the report for completeness and shall upload the Dissertation along with the name, University Seat Number, address, mobile number of the candidate etc., as prescribed in the form available on online Dissertation evaluation portal.
- Once the Guide uploads the dissertation, the same shall be linked for plagiarism check. The allowable plagiarism index shall be less than or equal to 25%. If the check indicates a plagiarism index greater than 25%, he/she shall, resubmit the dissertation to the Registrar (Evaluation)/Regional Centre/ Head Office, VTU along with the penal fees.
- By keeping the business trend in the present scenario, university has given an option to the students to select the research problem either from business organization or they can carry out the project on freelance basis subject to the approval of department committee. It is the total responsibility of the internal guide to monitor the freelance project.
- In case, business problem selected from a Company, no two students of an institute shall work on the same problem in the same organization.
- The student shall seek the guidance of the internal guide on a continuous basis, and the guide shall give a certificate to the effect that the candidate has worked satisfactorily under his/her guidance.
- On completion of the project work, student shall prepare a report with the following format.
  i. The Project report shall be prepared using word processor viz. MS Word with New Times Roman, 12 font size and shall be in the A4 size 1" margin on all the sides (1.5 inch on left side) and 1.5 line spacing. The Project report shall not exceed 100 pages.
  ii. The report shall have a title sheet with the title of the project, guide details and month & year of admission.
  iii. A certificate by the guide, Programme Coordinator and the Director indicating the bonafide performance of the project by the student to be enclosed.
  iv. An undertaking by the student to the effect that the work is independently carried out by him/her.
  v. The certificate from the organization if applicable (if its Freelance project, certificate is not required and internal guide can issue a certificate for successful completion).

Project Report Evaluation:

- Internal evaluation will be done by the internal guide.
- External valuation shall be done by faculty members of PG centers of VTU and VTU affiliated institutes with minimum of 10 years experience.
- **Viva-Voce / Presentation:** A viva-voce examination shall be conducted online where a student is expected to give a presentation of his/ her work.
- Minimum passing marks of the Project work is 50% in each of the components such as Internal Marks, report evaluation and viva-voce examination.

PROJECT STRUCTURE

Abstract

Acknowledgement

Table of Contents

Table of Figures

CHAPTER 1: INTRODUCTION

1.1 Background

1.2 Objectives

1.3 Purpose, Scope, and Applicability

1.3.1 Purpose

1.3.2 Scope

1.3.3 Applicability

1.4 Achievements

1.5 Organisation of Report

CHAPTER 2: Literature Survey

CHAPTER 3: System Requirements & Specifications

3.1 Functional and Non functional Requirements

3.2 Software and Hardware Tools

3.3 Software Requirements Specification

CHAPTER 4: SYSTEM DESIGN

4.1 Basic Modules

4.2 Data Design

4.2.1 Schema Design

4.2.2 Data Integrity and Constraints

4.3 Procedural Design

4.3.1 Logic Diagrams

4.3.2 Data Structures

4.3.3 Algorithms Design

4.4 User interface design

4.5 Security Issues

4.6 Test Cases Design

CHAPTER 5: IMPLEMENTATION

5.1 Implementation Approaches

5.2 Coding Details and Code Efficiency

5.2.1 Code Efficiency

5.3 Testing Approach

5.3.1 Unit Testing

5.3.2 Integrated Testing

5.4 Modifications and Improvements

CHAPTER 6: TESTING

6.1 Test Reports

6.2 User Documentation

CHAPTER 7: CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusion

7.2 Limitations of the System

7.3 Future Scope of the Project REFERENCES

GLOSSARY APPENDIX A APPENDIX B

Abstract

This should be one/two short paragraphs (400 words), summarising the project work. It is important that this is not just a re-statement of the original project outline. A suggested flow is background, project aims and main achievements.

NOTE: From the abstract, a reader should be able to ascertain if the project is of interest to them and, it should present results of which they may wish to know more details.

Chapter 1: Introduction

The introduction has several parts as given below:

Background:

A description of the background and context of the project and its relation to work already done in the area. Summarise existing work in the area concerned with your project work.

Objectives:

Concise statement of the aims and objectives of the project. Define exactly what you are going to do in the project; the objectives should be about 30 /40 words.

Purpose, Scope and Applicability:

The description of Purpose, Scope, and Applicability are given below:

- **Purpose:**

Description of the topic of your project that answers questions on why you are doing this project. How your project could improve the system its significance and theoretical framework.

- **Scope:**

A brief overview of the methodology, assumptions and limitations.

You should answer the question: What are the main issues you are covering in your project? What are the main functions of your project?

- **Applicability:**

You should explain the direct and indirect applications of your work. Briefly discuss how this project will serve the computer world and people.

Achievements:

Explain what knowledge you achieved after the completion of your work. What contributions has your project made to the chosen area?

Goals achieved - describes the degree to which the findings support the original objectives laid out by the project. The goals may be partially or fully achieved, or exceeded.

Organisation of Report:

Summarising the remaining chapters of the project report, in effect, giving the reader an overview of what is to come in the project report.

Chapter 2: SURVEY OF TECHNOLOGIES

In this chapter

- You should demonstrate your awareness and understanding of Available Technologies related to the topic of your project.
- You should give the detail of all the related technologies that are necessary to complete your project.
- You should describe the technologies available in your chosen area and present a comparative study of all those Available Technologies.
- Explain why you selected the one technology for the completion of the objectives of your project.

Chapter 3: REQUIREMENTS AND ANALYSIS

3.1 Problem Definition:

Formulate/define the problem on which you are working in the project.

Provide details of the overall problem and then divide the problem in to sub- problems. Define each sub-problem clearly.

3.2 Proposed Solution:

Define briefly the methodology/technology you are proposing to solve the problem on which you are working in the project.

3.3 Planning and Scheduling:

Planning and scheduling is a complicated part of software development. Planning, for our purposes, can be thought of as determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules, and known as constraints, which, control when certain tasks can or cannot happen. Scheduling can be thought of as determining whether adequate resources are available to carry out the plan. You should show the Gantt chart and Program Evaluation Review Technique (PERT).

3.4 Software and Hardware Tools used:

Define the details of all the software and hardware needed for the development and implementation of your project.

- Hardware Requirement: In this section, the equipment, graphics card, numeric co-processor, mouse, disk capacity, RAM capacity etc. necessary to run the software must be noted.
- Software Tools used: In this section, the operating system, the compiler, testing tools, linker, and the libraries etc. necessary to compile, link and install the software must be listed.

3.5 Preliminary Product Description:

Identify the requirements and objectives of the new system. Define the functions and operation of the application/system you are developing as your project.

3.6 Conceptual Models:

You should understand the problem domain and produce a model of the system, which describes operations that can be performed on the system, and the allowable sequences of those operations. Conceptual Models could consist of complete Data Flow Diagrams, ER diagrams, Object-oriented diagrams, System Flowcharts etc.

3.7 Software Requirements Specification:

- In this phase you should define the requirements of the system, INDEPENDENT of how these requirements will be accomplished.
- The Requirements Specification describes the things in the system and the actions that can be done on these things.
- Identify the operation and problems of the existing system.

i. USER REQUIREMENTS

ii. SYSTEM REQUIREMENTS

- FUNCTIONAL REQUIREMENTS
- NON-FUNCTIONAL REQUIREMENTS
- DOMAIN REQUIREMENTS

Chapter 4: SYSTEM DESIGN

Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudocode and other documentation.

Basic Modules:

You should follow the divide and conquer theory, so divide the overall problem into more manageable parts and develop each part or module separately. When all modules are ready, you should integrate all the modules into one system. In this phase, you should briefly describe all the modules and the functionality of these modules.

Data Design:

Data design will consist of how you organise, managing and manipulate the data.

- Schema Design: Define the structure and explanation of schemas used in your project.
- Data Integrity and Constraints: Define and explain all the validity checks and constraints you are providing to maintain data integrity.

Procedural Design:

Procedural design is a systematic way for developing algorithms or procedurals.

- **Logic Diagrams:**

Define the systematically flow of procedure that improves its comprehension and helps the programmer during implementation. e.g., Control Flow Chart, Process Diagrams etc.

- **Data Structures:**

Create and define the data structure used in your procedures.

- **Algorithms Design:**

With proper explanations of input data, output data, logic of processes, design and explain the working of algorithms.

User Interface Design:

- Define user, task, environment analysis and how you intend to map those requirements in order to develop a "User Interface".
- Describe the EXTERNAL and INTERNAL components and the architecture of your user interface.
- Show some rough pictorial views of the user interface and its components.

Security Issues:

Discuss Real-time considerations and Security issues related to your project and explain how you intend avoiding those security problems. What are your security policy plans and architecture?

Test Cases Design:

Define test cases, which will provide easy detection of errors and mistakes with in a minimum period of time and with the least effort. Explain the different conditions in which you wish to ensure the correct working of your software.

Chapter 5: IMPLEMENTATION AND TESTING

Implementation Approaches:

Define the plan of implementation, and the standards you have used in the implementation.

Coding Details and Code Efficiency:

Students not need include full source code, instead, include only the important codes (algorithms, applets code, forms code etc). The program code should contain comments needed for explaining the work a piece of code does. Comments may be needed to explain why it does it, or, why it does a particular way.

You can explain the function of the code with a shot of the output screen of that program code.

- **Code Efficiency:** You should explain how your code is efficient and how you have handled code optimisation.

Testing Approach: Testing should be according to the scheme presented in the system design chapter and should follow some suitable model – e.g., category partition, state machine-based. Both functional testing and user-acceptance testing are appropriate. Explain your approach of testing.

- **Unit Testing:**

Unit testing deals with testing a unit or module as a whole. This would test the interaction of many functions but, do confine the test within one module.

- **Integrated Testing:**

Brings all the modules together into a special testing environment, then checks for errors, bugs and interoperability. It deals with tests for the entire application. Application limits and features are tested here.

Modifications and Improvements:

Once you finish the testing you are bound to be faced with bugs, errors and you will need to modify your source code to improve the system. Define what modification you implemented in the system and how it improved your system.

Chapter 6: RESULTS AND DISCUSSION

Test Reports:

Explain the test results and reports based on your test cases, which should show that your software is capable of facing any problematic situation and that it works fine in different conditions. Take the different sample inputs and show the outputs.

User Documentation:

Define the working of the software; explain its different functions, components with screen shots. The user document should provide all the details of your product in such a way that any user reading the manual, is able to understand the working and functionality of the document.

Chapter 7: CONCLUSION:

The conclusions can be summarised in a fairly short chapter (2 or 3 pages). This chapter brings together many of the points that you would have made in the other chapters.

Limitations of the System:

Explain the limitations you encountered during the testing of your software that you were not able to modify. List the criticisms you accepted during the demonstrations of your software.

Future Scope of the Project:

It describes two things: firstly, new areas of investigation prompted by developments in this project, and secondly, parts of the current works that were not completed due to time constraints and/or problems encountered.

REFERENCES

It is very important that you acknowledge the work of others that you have used or adapted in your own work, or that provides the essential background or context to your project. The use of references is the standard way to do this. Please follow the given standard for the references for books, journals, and online material.

GLOSSARY

If you use any acronyms, abbreviations, symbols, or uncommon terms in the project report then their meaning should be explained where they first occur. If you go on to use any of them extensively then it is helpful to list them in this section and define the meaning.

APPENDICES

These may be provided to include further details of results, mathematical derivations, certain illustrative parts of the program code (e.g., class interfaces), user documentation etc.

Rubrics for Project Work

Sl. No.	Evaluation Type	Particulars	Marks
1	CIE	Internal Assessment by the Guide- Based on the Presentations by Students	30
2	SEE	Report Evaluation by the Guide & External Examiner. Average of the marks awarded by the two Examiners shall be the final evaluation marks for the Dissertation	35
3	SEE	Viva-Voce Examination to be conducted by the Guide and an External examiner from the Industry/ Institute (Joint Evaluation)	35
Total			**100**

Rubrics for Project Evaluation and Viva voce Examination

A. Internal Assessment by the Guide- Based on three Presentations by Students

Sl. No.	Aspects	Marks
1	Three Presentations	5
2	Introduction and Methodology	5
3	Industry and Company Profile	5
4	Theoretical background of study	5
5	Data analysis and interpretation	5
6	Summary of findings, suggestions and conclusion	5
Total		**30**

B. Report Evaluation by the Guide & External Examiner

Sl. No.	Aspects	Marks
1	Introduction & Relevance of the project	5
2	Conceptual background and literature review	5
3	Research design	5
4	Analysis and interpretation	10
5	Summary of findings, suggestions and conclusion	10
Total		**35**

C. Viva-Voce Examination to be conducted by the Guide and an External examiner from the Industry/ Institute (Joint Evaluation)

Sl. No.	Aspects	Marks
1	Presentation and Communication Skills	5
2	Subject knowledge	5
3	Objectives of the study and Methodology	5
4	Analysis using statistical tools and statistical packages	10
5	Findings and appropriate suggestions	10
Total		**35**

Activity Chart to be followed during Project Work

Activity	Remarks
Identifying the organization and Problem identification	Student individually identifies an organization or identifies problem for his/her study, according to his/her interest.
Problem statement & Research Design	His/ Her interests are discussed with project guides. Discussion with Internal Guide to decide on suitable design for the research
Synopsis Preparation	Preparation of Synopsis* & formulating the objectives
Presentation of Synopsis	The student will present the synopsis with the detailed execution plan to the Internal Guide and Programme Coordinator who will review and may: a. Approve b. Approve with modification or c. Reject for fresh synopsis

Approval Status	The approval status is submitted to Programme Coordinator who will officially give concurrence for the execution of the Project
Understanding Structure, Culture and functions of the organization / Identifying of business problem from the Industry through the literature study	Student should understand products / services and the problems of the organization
Preparation of Research design and Research instrument for data collection	Discussion with the guide for finalization of research design and instrument in his/her domain and present the same to the guide. **(First Presentation)**
Data collection	Date collected to be edited, coded, tabulated and presented to the guide for suggestions for analysis. **(Second Presentation)**
Analysis and finalization of report	Students must use appropriate and latest statistical tools and techniques for analyzing the data. **(Third Presentation)**
Submission of Report	Final Report should be submitted to the University before one week of the commencement of theory examination.

***Synopsis of 3-4 pages to be submitted to the Programme Coordinator through the Guide**

Page 1	Title, Contact Address of student- with details of Internal and External Guide (if applicable)
Pages 2-4	Short introduction with objectives and summary (300 words), Review of Articles / Literature about the topic with source of information.

Formats for Project Report

- Format of Cover Page
- Format of certificate by Company/Institution or from both
- Format of Declaration Page
- Format of Contents
- Format of List of Tables and Charts
- Format of Bibliography

Visvesvaraya Technological University, Belagavi

Centre for Distance and Online Education (CDOE), Mysuru

(Title of the Project Work)

Submitted by

(Student Name)

(USN)

Submitted to

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAVI

In partial fulfillment of the requirements for the award of the degree of

[MASTER OF COMPUTER APPLICATION / MASTER OF COMPUTER APPLICATION IN AI & DS / MASTER OF COMPUTER APPLICATION IN CY & CC]

Under the guidance of

INTERNAL GUIDE	EXTERNAL GUIDE
(Name & Designation)	(Name & Designation)

(Institute Logo)

Department of Computer Applications

VTU's Centre for Distance and Online Education

Mysuru

(Month & Year of submission)

Visvesvaraya Technological University, Belagavi

Centre for Distance and Online Education (CDOE), Mysuru

VTU-CDOE MYSURU

CERTIFICATE

This is to certify that (Name of the Student) bearing USN (xxxx), is a bonafide student of [Master Of Computer Application / Master Of Computer Application In AI & DS / Master Of Computer Application In CS & CC] course of the Institute (Batch), affiliated to Visvesvaraya Technological University, Belgavi. Project Report on "(Title of Report)"is prepared by him/her under the guidance of (Name of the Guide), in partial fulfilment of the requirements for the award of the degree of [Master Of Computer Application / Master Of Computer Application In AI & DS / Master Of Computer Application In CS & CC] of Visvesvaraya Technological University, Belagavi, Karnataka.

Signature of Internal Guide **Signature of PC** **Signature of Director**

VTU-CDOE MYSURU

DECLARATION

I, (Student Name), hereby declare that the Project report entitled "(Title)" with reference to (Organization with place) prepared by me under the guidance of (Guide Name), faculty of Computer Application Department, (Institute name) and external assistance by (External Guide Name, Designation and Organization). I also declare that this Project work is towards the partial fulfilment of the university Regulations for the award of degree of [Master Of Computer Application / Master Of Computer Application In AI & DS / Master Of Computer Application In CS & CC] by Visvesvaraya Technological University, Belagavi. I have undergone a summer project for a period of Twelve weeks. I further declare that this Project is based on the original study undertaken by me and has not been submitted for the award of any degree/diploma from any other University / Institution.

Place: **Signature of the Student**

Date:

PROGRAMME CO-ORDINATOR
COMPUTER APPLICATIONS
Visvesvaraya Technological University
Centre for Distance and Online Education
MYSURU-570 029