

Centre for Distance and Online Education (CDOE)



PROGRAMME PROJECT REPORT (PPR)
FOR
MASTER OF COMPUTER APPLICATION (M.C.A)

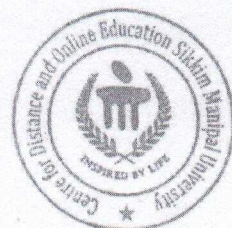
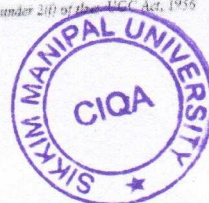
SIKKIM MANIPAL UNIVERSITY
CENTRE FOR DISTANCE AND ONLINE EDUCATION (CDOE)

[Handwritten signature]



SMU SIKKIM
MANIPAL
UNIVERSITY

Established under the Sikkim Manipal University Act, 1975, recognised under 2(f) of the UGC Act, 1956



Dr. [Name]
Asst. Director, CDOE

Sikkim Manipal University

Vision

Global Leadership in Human Development, excellence in Education and Healthcare

Mission

Develop professionals of excellent calibre in the field of Health Sciences, Engineering, Management, Medical Biotechnology, Basic and Social Sciences, with a humane approach, capable of shouldering the responsibility of building the nation and be globally competent.

Centre for Distance and Online Education

Vision

Empowering tomorrow's global learners through accessible, innovative, and inclusive online education.

Mission

To foster a dynamic virtual learning environment that transcends boundaries, leveraging technology and pedagogical excellence to cultivate personalized, lifelong learning experiences for diverse communities worldwide



PROGRAMME PROJECT REPORT
MASTER OF COMPUTER APPLICATION (M.C.A.)
CENTRE FOR DISTANCE AND ONLINE EDUCATION
(CDOE)
SIKKIM MANIPAL UNIVERSITY



Sl. No	Contents	Page No
1.	Programme's Mission and Objectives	3
2.	Relevance of Programme with Sikkim Manipal University Mission and Goals	3
3.	Nature of Prospective Target Group of Learners	4
4.	Appropriateness of programme to be conducted in Online mode to acquire specific skills and competence	4
5.	Instructional Design	
	5.1 Curriculum design	5
	5.2 Programme structure and Detailed syllabus	5
	5.3 Duration of the programme	28
	5.4 Faculty and support staff requirement	28
	5.5 Instructional delivery mechanisms	28
	5.6 Identification of media—print, audio or video, online, computer aided	29
	5.7 Student Support Services	29
6.	Procedure for Admission, Curriculum Transaction and Evaluation	
	6.1 Procedure for Admission	30
	6.2 Curriculum Transactions	31
	6.3 Evaluation	33
	6.4 Practical and Project	36
7.	Requirement of the Laboratory Support and Library Resources	38
8.	Cost Estimate of the Programme and the Provisions	39
9.	Quality Assurance Mechanism and Expected Programme Outcomes	39
10.	Scholarship and Financial Assistance Policy	40
11.	Learner's Support	41



PROGRAMME PROJECT REPORT (PPR)

Introduction

The Programme leading to the award of Master of Computer Applications (MCA) is developed to prepare students to take up a career in the field of IT and Computer Applications. This is a Post Graduate Programme where students are exposed to various areas of Computer Applications including the latest developments in the Industry.

1. Programme's Mission and Objectives

To afford a Quality Postgraduate Degree in the field of computing science and Information Technology through Online Learning mode to impart high quality training to students using the latest tools in computer technology. To upkeep the high standard of education, the academic syllabus is designed keeping in view the latest trends and cutting-edge technologies in the field of Computer Science. The coursework is designed to be flexible and wide-ranging, incorporating cutting edge technology and ensuring that the students have a firm grasp on the core fundamentals of IT, and its applications.

The objectives of the programme are to enable students to:

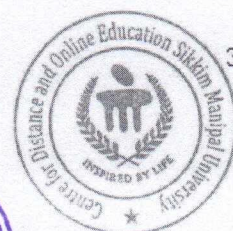
- Work productively as IT professionals both at supportive and leadership roles.
- Advance successfully in their chosen career path utilizing technical abilities, leadership qualities, communication, and interpersonal skills with high regard to legal and ethical responsibilities, in IT domain.
- Build their profession adaptable to the changes in technology, with lifelong learning.

2. Relevance of Programme with Sikkim Manipal University, Mission and Goals

In order to align with the mission and goals of Sikkim Manipal University, the Online MCA Programme is planned to enable students and working professionals gain knowledge in various domains of IT, specialize in a domain of their choice, gain knowledge of not only IT, but also managerial skills including analysis, data based decision making and entrepreneurship, in newer and emerging markets, products and technologies.

Vision

Technologically skilled and equipped leadership in all domains and Higher education for Human Development



Mission

- Enable human resource in facilitating technological knowledge for better governance and administration
- Transforming all to most advanced and upgraded technical knowledge
- Remain ahead in fostering research and cutting edge technology through innovative and interdisciplinary learning

3. Nature of Prospective - Target Group of Learners

It is by now well accepted that an MCA degree is an important tool for professionals to contribute to business in all areas of IT, expand their career options and move up their career ladder, acquire Leadership skills or embark on an entrepreneurial journey.

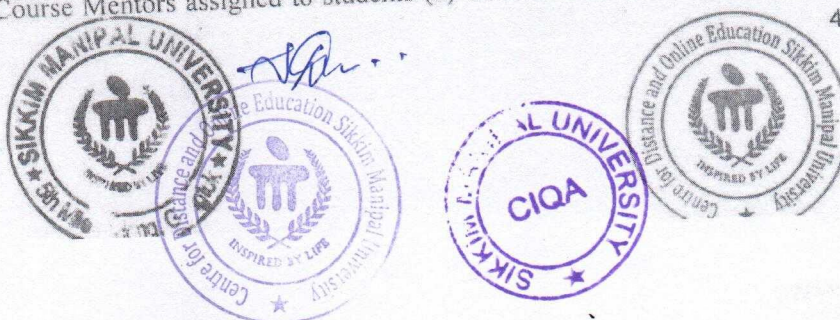
This Online programme has been designed for conventional learners, as well as working professionals and other individuals aspiring to acquire knowledge and associated academic credentials. Considering that all candidates interested in pursuing a degree may not be able to afford the same through a campus mode for reasons of paucity of time or financial constraints, online delivery is a feasible option to enable them to acquire knowledge and skills. Delivery through this mode also contributes towards Gross Enrolment Ratio (GER) of 50% by 2035, as envisaged by the Government of India.

The programme is so designed that the prospective students who may not be able to afford full time, residential MCA are provided with high value learning, anytime, anyplace, at one's own pace.

4. Appropriateness of programme to be conducted in Online mode to acquire specific skills and competence

The courses in the programme are delivered through Self-Learning e-Module which is a modular unit of e-learning material which is inter-alia self-explanatory, self-contained, self-directed at the learner, and amenable to self-evaluation, and enables the learner to acquire the prescribed level of learning in a course of study and includes contents in the form of a combination of the following e-Learning content, and made available through four-quadrant approach namely,

- (a) e-Tutorial - faculty led Audio - Video Lectures, (b) e-Content (combination of PDF/ e-pub) Text Materials, (c) Discussion forum for raising of doubts and clarifying the same on real time basis by the Course Coordinators/Course Mentors assigned to students (d) Self-Assessment Quiz, Test and



Assignments to reinforce learning. Reference books are also mentioned in the syllabus. Latest Edition of Reference books may be referred to.

A robust Learning Management System that keeps track of delivery of e-Learning Programmes, learner's engagement, assessment, results and reporting in one centralized location, is in place. All of the above can be done/delivered by online and other platforms without much loss of fidelity. Hence the MCA programme is suited for Online mode of learning.

5. Instructional Design

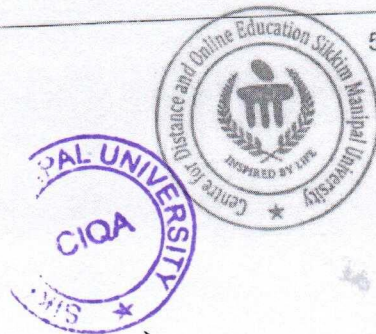
5.1. Curriculum design

Curriculum has been designed by experts in the area of Management and Information Technology and care has been taken to include contemporary topics, as well as topics that also inculcate environmental awareness in students. The curriculum and syllabus are approved by the Board of Studies, Centre for Internal Quality Assurance (CIQA) and University Academic Council which consists of experts from Academia and Industry.

5.2. Programme structure and detailed syllabus

5.2.1 Programme Structure

SUBJECT CODE	SUBJECT NAME	CREDITS	TOTAL CREDITS
SEMESTER I			
OMCA001	FUNDAMENTALS OF COMPUTERS AND IT	-	20
OMCA002	FUNDAMENTALS OF MATHEMATICS	-	
OMCA101	COMPUTATIONAL MATHEMATICS	4	
OMCA102	JAVA PROGRAMMING	4	
OMCA103	OPERATING SYSTEMS	4	
OMCA104	DATABASE MANAGEMENT SYSTEM	4	
OMCA105	DATABASE MANAGEMENT SYSTEM LAB	2	
OMCA106	JAVA PROGRAMMING LAB	2	
SEMESTER II			
OMCA201	PYTHON PROGRAMMING	4	20
OMCA202	COMPUTER ORGANIZATION AND ARCHITECTURE	4	
OMCA203	SOFTWARE ENGINEERING AND UNIFIED MODELLING LANGUAGE	4	
OMCA204	DATA STRUCTURE AND ALGORITHMS	4	
OMCA205	DATA STRUCTURE AND ALGORITHMS LAB	2	
OMCA206	PYTHON PROGRAMMING LAB	2	
SEMESTER III			



OMCA301	.NET FRAMEWORK	4	20
OMCA302	COMPUTER NETWORK	4	
OMCA303	IT LAWS AND PRACTICES	4	
	Elective – I (Select Any One)		
OMCA3041	DATA WAREHOUSING AND DATA MINING	4	
OMCA3042	CLOUD COMPUTING	2	
OMCA305	.NET FRAMEWORK LAB	2	
OMCA306	COMPUTER NETWORK LAB		
SEMESTER IV			
OMCA401	MOBILE APPLICATION DEVELOPMENT	4	20
OMCA402	ANGULAR JS, REACT JS AND VUE JS	4	
	ELECTIVE –II (Select Any one)		
OMCA4031	MACHINE LEARNING	4	
OMCA4032	DISTRIBUTED SYSTEM AND GRID COMPUTING	8	
OMCA405	PROJECT		80
TOTAL CREDITS			

5.2.2 Detailed Syllabus

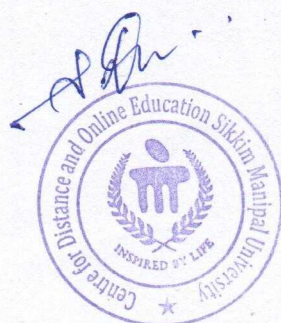
SEMESTER I	
No of units	OMCA001 FUNDAMENTALS OF COMPUTER AND IT; Bridge course
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to discuss the fundamental concepts of computer and IT 2. Enable students to explain the basics of web design, CSS and web publishing 3. Students will be able to identify number system. 4. Students will be able to explain combinational logic circuits and sequential logic circuits 5. Students will able to write, Compile and Debug basic C programs using an IDE and using a workstation.
1	Decimal, Binary, Octal, Hexadecimal Number systems, and number base conversion, complements- Binary Codes, Binary storage, 1's Complements, 2's Complements, Axiomatic definitions of Boolean algebra – Basic theorem and properties, Boolean functions
2	Basic Gates (NOT, AND & OR), canonical and standard forms, other logical operations, and IC digital logic families, Adders, Subtractor, Code Converter etc., Multilevel NAND and NOR circuits, EX-OR and Equivalence functions, combinational logics.
3	Different types of Flip-flops and their triggering (D-flip flops, RS-Flip flops, JK-Flip flops, Master – Slave Flip flops)
4	RAM, ROM, Word, Capacity, Address, Access time, Memory Hierarchy, Magnetic Core Memory, Semiconductor memory (SRAM and DRAM), Magnetic tape storage, magnetic disk storage
5	The evolution of computers, Characteristics of computers, Organization of a Computer, Input Unit, Central Processing Unit, Computer Output Units, CPU Essentials, The BUSES, Processor Modes, Modern CPU concepts, CISC vs. RISC CPUs, Circuit Size and Die Size, Processor Speed, Processor Cooling, System Clocks, Architectural Performance Features
6	Classification of Computers: Supercomputer, Mainframe Computers, Minicomputers, Microcomputer, Distributed Computer System, Parallel Computers





7	Programming Concepts and Fundamentals, Types of Programming Language, Structured vs. Unstructured Programming Language, Procedural vs. Object Oriented Programming Language; Basic Structure of C and C++ Programs, Character Set, Token, Identifiers and keywords, Constants, Variable, Data Types, Variable Declarations.
8	Features of C and its Basic Structure, Simple C programs, Constants, Integer Constants, Real Constants, Character Constants, String Constants, Backslash Character Constants, Concept of an Integer and Variable, Rules for naming Variables and assigning values to variables
9	Data Types and Input/Output Operators, Floating-point Numbers, Converting Integers to Floating-point and vice-versa, the type cast Operator, The type char, Character Input and Output, Formatted input and output, The gets() and puts() functions
10	Control Statements and Decision Making, The goto statement, The if statement, The if-else statement, The conditional expression, The switch statement, The while loop, The do...while loop, The for loop, The nesting of for loops, The break statement and continue statement
11	Introduction to HTML, The set of tags, Block level and text level tags, Images and image maps, Client side server side maps, Anchor elements, Tables, Frames, Forms.
12	Introduction to DHTML, Cascading style sheets, DOM, Scripting, ActiveX pad with embedded active controls.

OMCA101 COMPUTATIONAL MATHEMATICS; Credits: 4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to apply fundamental concepts of computation for mathematics 2. Enable students to implement algorithms. 3. Enable students to analyze algorithms 4. Enable students to design efficient algorithms 5. Enable students to evaluate and choose suitable algorithms
Pre-requisite: Basic concept of matrix algebra, ordinary differential equations, linear algebra.	
1	Number Systems and Errors: The representation of Integers, Fractions, Floating Point Arithmetic, Loss of significance and Error propagation, Computational Method for Error Estimation, Some Mathematical Preliminaries Complex Numbers and their Operations, Number Bases and Conversions
2	Interpolation: Polynomial Forms, Existence and Uniqueness of Interpolating Polynomial, Finite Difference: Operators with properties and application, Newton's polynomial interpolation formulae. Divided difference and Lagrange's interpolation polynomials with Algorithms Spline Interpolation
3	The Solution of Nonlinear Equations: Iterative Methods: Bisection Method, Regula Falsi Method, Secant Method, Newton Raphson method, and their algorithms, Fixed Point iterative method. Convergence of methods. (onen method is sufficient) Brent's Method
4	Solution of System of Linear Equations: The Solution of Linear Systems by Elimination: Gauss Elimination with algorithm Iterative Refinement, Effect of Pivoting
5	The Triangular Factorization: Cholesky, LU decomposition methods with algorithm, Method for Matrix Inversion. The Iterative Method: Jacobi, Gauss-Seidal methods QR Factorization
6	Eigen value Problems: Definition of eigen value, eigen vector, and characteristic equation. Algebraic and geometric multiplicities Finding all eigen values and eigen vectors of a given square matrix, Power Method with algorithm Applications of Eigenvalues and Eigenvectors
7	Approximation: Approximation by Polynomials, Data fitting, Least square approximation by polynomials. Fast Fourier Transformation Chebyshev Polynomials



8	Numerical Differentiation and Integration: Forward difference, backward difference and central difference formula with error terms. Some basic Rules such as Trapezoidal rule, Simpsons' rule, their derivations and algorithms. Gaussian Rule: 5-Point Legendre formula, Laguerre and Hermite Error Analysis for Numerical Integration, Adaptive Quadrature Methods
9	The Solution of Differential Equations: Simple difference equations, Numerical solution of initial value problems in ordinary differential equations by Taylor series method, Picard's method, Euler's method, Modified Euler's method, Runge Kutta fourth order method with Algorithms Stability of Methods, Stiff Differential Equations
10	Multi Step Method: Predictor-Corrector Method, Milne's Predictor and Corrector method. Finite difference method for the solution of boundary value problems in linear ordinary differential equations using central differences Adams-Moulton Method
11	Graphs and Algorithms I: Basic definitions: Graph, incidence and degree, isolated vertex, null graph, weighted graph, subgraphs, networks Applications of graphs, Definitions: Walks, paths, circuits, connected and disconnected graphs, components, Euler graph, Hamiltonian paths and circuits. Traveling salesman problem Graph Isomorphism, Graph Coloring
12	Graphs and Algorithms II: Cut-sets: Definition and properties All cut-sets in a graph Fundamental circuits and cut-sets, Matrices associated with graphs: Matrix representations of simple graphs: Incidence and adjacency matrices. Properties of incidence matrix with example Properties of adjacency matrix with example Distance matrices Graph Traversal Algorithms (DFS, BFS). Shortest Path Algorithms (Dijkstra's, Bellman-Ford)

OMCA102 JAVA PROGRAMMING; Credits: 4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to understand the core concepts of Object-Oriented Programming (OOP). 2. Enable students to implement complex software solutions. 3. Enable students to utilize Java Collections Framework and Java Stream API. 4. Enable students to design user interfaces for Java applications. 5. Enable students to evaluate the efficiency of Java code.
Pre-requisites: Object Oriented Programming.	
1	Introduction to Java: Introduction, History of Java, Features of Java, Java Virtual Machine (JVM), Java Runtime Environment (JRE), Java Development Kit (JDK), Security in Java Java coding standards, Java Concurrency Utilities
2	Java Basics: Keywords, Working of Java Including Comments, Data Types in Java, Primitives Data Types, Abstract/Derived Data Types, Variables in Java, Using Classes in Java, Declaring Methods in Java, Java, Generics, Generic Methods, Generic Classes, Compiling and Executing a Java Program Java Naming Conventions, Basic Object-Oriented Concepts (Class, Object, Inheritance, Polymorphism, Abstraction, Encapsulation)
3	Operators and Control Statements: Operators, Arithmetic Operators, Increment and Decrement Operators, Comparison Operators, Logical Operators, Operator Precedence, Control Flow Statements, Loops, Break and Continue Statement Ternary Operator, Enhanced for loop
4	Arrays and Strings: String Handling, Special String Operations, Character Extraction, String Comparison, Searching Strings, String Modification, StringBuffer String Builder, Java Collections Framework (brief overview)
5	Inheritance, Package, and Interface: Inheritance, Types of Relationships, Access Specifiers, Abstract Class, Packages, Understanding CLASSPATH, Interface, Some Uses of Interfaces, Interfaces vs Abstract Classes, Nested Classes Default and static methods in Interfaces, Module



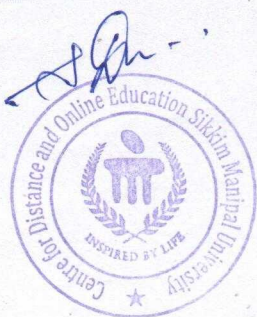
	System in Java
6	Exception Handling: Definition of an Exception, Exception Classes, Common Exceptions, Exception Handling Techniques Try-with-resources, Custom exceptions
7	Streams in Java: Streams Basics, The Abstract Streams, Stream Classes, Readers and Writers, Random Access Files, Serialization, Stream API Java NIO Package, File Handling
8	Collections: Collection Hierarchy, List, Set, Map Interfaces and their Common Implementations, Iterators, Collections Utility Class, Arrays Utility Class, Vectors and Stack, Hashtable, Properties Queue, Deque, Concurrent Collection classes
9	MultiThreading: Process and Java Threads, Creating and Starting a Thread, Key Methods of a Thread, Synchronization, Interthread Communication Executors, Callable and Future
10	Other Features in Java: Assertion, Variable Argument (Varargs), Java Static Import, Autoboxing and Unboxing, Java Enum, Java Annotation, Java Custom Annotation, Junit, Unit Testing and Junit, Writing Test, Junit Assertions Functional Interfaces, Lambda Expressions, Stream API advanced concepts
11	Graphical User Interface: Java Foundation Classes, Java Swing Packages, Swing Component Classes, JavaFX – Architecture, Layout Pane Event Handling, JavaFX Scene Builder, JavaFX Properties and Bindings
12	Java Database Connectivity (JDBC): Java Data Base Connectivity, Database Management, Mechanism for connecting to a backend database (ODBC), Loading the ODBC driver, Brief overview of Hibernate, JPA Connecting to various types of databases, Transaction Management, Spring Data JPA

OMCA103 OPERATING SYSTEMS; Credits:4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to understand fundamental algorithmic problem-solving techniques. 2. Enable students to compare the time and space complexity of different algorithms. 3. Enable students to implement various algorithms in a high-level programming language and assess their practical efficiency. 4. Enable students to design new algorithms or modify existing ones to efficiently solve real-world computing problems. 5. Enable students to evaluate the performance and suitability of different algorithms for given problem scenarios.
1	Introduction: Basic Concepts, Simple Batch Systems, Multi-programmed Batched Systems, Time-Sharing Systems. History of Operating Systems, Types of Operating Systems, Operating System Structures
2	Processes and Threads: Process Concept, Process Scheduling, Operation on Processes, Cooperating Processes, Inter-process Communication Thread Scheduling, Multithreading Models, Process and Thread Management in modern operating systems
3	CPU Scheduling: Scheduling Criteria, Scheduling algorithms, Multiple process scheduling Priority Scheduling, Round Robin Scheduling, Multilevel Queue Scheduling
4	Process Synchronization: The Critical-Section Problem, Basics of Semaphores Mutex, Monitors, Classical Problems of Synchronization
5	Deadlocks: Deadlock Characterization, Methods of Handling Deadlock, Deadlock Prevention.



	Deadlock Avoidance, Deadlock Detection & Recovery from Deadlock Resource Allocation Graph, Deadlock Prevention Strategies
6	Memory Management: Logical versus Physical Address Space, Swapping, Paging Fragmentation, Allocation Methods, Segmentation
7	Virtual Memory: Need for Virtual Memory Technique, Demand Paging, Concept of Page Replacement, Page Replacement Algorithms: FIFO page replacement algorithm, Optimal algorithm, LRU page replacement algorithm, Thrashing Memory Mapping, Demand Segmentation
8	File System Interface and Implementation: Concept of a File, File Access Methods, Directory Structure, Allocation Methods, Free Space Management, Directory Implementation File System Mounting, File Sharing, Protection, Overview of Mass storage structure, Disk Structure: FCFS Scheduling, SSTF Scheduling, SCAN scheduling, C-SCAN Scheduling, LOOK Scheduling.
9	Input – Output Architecture: I/O Structure, I/O Control Strategies, The I/O Address Space Disk Scheduling, Disk Management, Swap Space Management
10	Operating Systems in Distributed Processing: Centralized and Distributed Processing, Network Operating System (NOS) Architecture, Functions of NOS, Global Operating System (GOS), Remote Procedure Call (RPC), Distributed File Management Distributed Coordination, Recovery
11	Security and Protection: Attacks on Security, Computer Worms, Computer Virus, Security Design Principles, Authentication, Protection Mechanism, Encryption, Security in Distributed Environment Firewall Configuration, Intrusion Detection Systems, Secure OS Design
12	Multiprocessor Systems: Multiprocessors, Multiprocessor Classification, Multiprocessor Interconnections, Types of Multiprocessor Operating Systems (MOS), MOS Functions and Requirements, Operating System Design and Implementation Issues Case studies of real-world multiprocessor systems

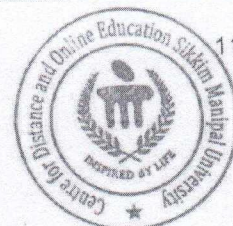
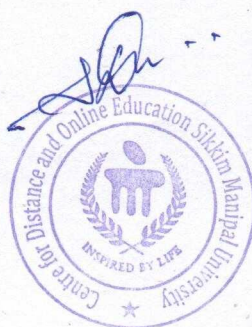
OMCA104 DATABASE MANAGEMENT SYSTEM; Credits:4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to interpret the core principles and applications of different types of databases and database management systems. 2. Enable students to implement sql commands and integrity constraints. 3. Enable students to optimize database queries. 4. Enable students to evaluate appropriate concurrency control methods. 5. Enable students to design robust and efficient database models and applications.
Pre-requisites: Programming Concepts.	
1	Comparison between Different Databases: Introduction, Significance of Databases, Applications of Database System, Personal databases, Two-Tier client/server databases, Multi-tier client/server databases, Enterprise application, Different Types of DBMS, Comparison between Centralised and Distributed Database
2	RDBMS and SQL: Introduction, Relational Query Languages, SQL, Integrity Constraints, Data Definition Statements, Data Manipulation Language, Normalisation and Database Design, Denormalisation Concepts of SQL Joins, SQL Functions and Stored Procedures
3	Query Optimisation: Introduction, Query Execution Algorithm, Heuristics in Query Optimisation, Semantic Query Optimisation, Multi-Query Optimisation and Application, Execution Strategies for SQL Sub Queries, Query Processing for SQL Updates
4	Query Execution: Introduction to Physical-Query-Plan Operators, Scanning tables, One-Pass Algorithms for Database Operations, Two-Pass Algorithms Based on Sorting and Hashing, Index-Based Algorithms, Parallel Algorithms for Relational Operations, Basic Algorithm for Executing



	Query Operations
5	Adaptive Query Processing and Query Evaluation: Introduction, Query Processing Mechanism: Eddy, Need and Uses of Adaptive Query Processing, Robust Query Optimisation through Progressive Optimisation, Query Evaluation Techniques for Large Databases
6	Concurrency Control: Introduction, Enforcing Serialisability by Locks, Concurrency Control by Timestamps, Concurrency Control by Validation, Database Recovery Management
7	Transaction Processing Systems: Introduction to Transaction Processing, Transaction States, Schedules, Serializability, Transaction Recovery Techniques Distributed Transaction Processing, Real-time Transaction Processing
8	Parallel Database Architectures for Parallel Databases: Introduction, Parallel Database, Parallel Query Evaluation, Parallelising Individual Operations, I/O Parallelism, Inter-Query Parallelism, Intra Query Parallelism, Design of Parallel Systems
9	Object Oriented DBMS: Introduction, Object Oriented Paradigm, OODBMS Architectural Approaches, Object Identity, Procedures and Encapsulation, Object Oriented Data Model, Basic OODMS Terminology, Basic Interface and Class Structure, Type Hierarchies and Inheritance Querying and Updating in OODBMS
10	Distributed Databases: Introduction of Distributed Databases, DDBMS architectures, Distributed Data Storage, Advantages and Disadvantages of Data Distribution, Distributed Transaction, Commit Protocols, Concurrency Control, Recovery of Distributed Database, Directory Systems, DDBMS Transparency Features
11	Object Relational and Extended Relational Databases: Introduction, Object Relational Database, Extension Techniques in RDBMS, Standards for OODBMS Products and Applications, Nested Relations and Collections, Storage and Access Methods, Implementation Issues for Extended Type, Comparing RDBMS, OODBMS and ORDBMS
12	Database Application: Introduction, Active Database, Temporal Database, Multimedia Database, Video Database Management Big Data Management, Cloud Databases

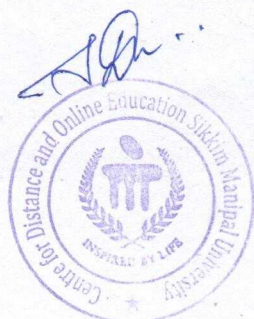
OMCA105 DATABASE MANAGEMENT SYSTEM LAB; Credits:2	
No of units	
Learning Objectives:	<p>1. Enable students to get hands on experience on core concepts of Database Management System</p> <p>2. Enable students to apply SQL for various operations in Database Management system</p>
1	Demonstrate Simple SQL Statements.
2	Demonstrate the use of SQL Built-in Functions.
3	Demonstrate Primary Key, Foreign Key & Normalization with SQL statement.
4	Demonstrate inner join, Left join, right join, full join with SQL statement.
5	Demonstrate the use of View and Union
6	Write a procedure to calculate simple interest, taking principle, rate and year as inputs.
7	Construct a DB for any organization and write a function to determine if an employee salary is out of range.

OMCA106 JAVA PROGRAMMING LAB; Credits:2	
No of units	
Learning Objectives:	<p>1. Enable students to get hands on experience on core concepts of Object-Oriented Programming (OOP).</p>



2. Enable students to code in Java for various applications.	
1	Write a Java program to display employee details (employee name, salary) using class and method invocation
2	Write a java program to demonstrate the use of (i) increment (++) and decrement operator (--) (ii) IF-ELSE Statement (iii) While loop.
3	Write a java program to perform matrix multiplication using 2-dimensional array
4	Write a Java Program to check whether given String is Palindrome or not
5	Write a java program to demonstrate the use of super class and sub class concept.
6	Write a Java program using exception handling methods
7	Write a Java program to create a class and a subclass and demonstrate that a method can override.
8	Write a Java program to demonstrate the use of byte stream.
9	Write a Java program to create and start multiple threads that increment a shared counter variable concurrently.

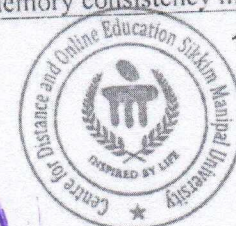
SEMESTER II	
OMCA201 PYTHON PROGRAMMING; Credits: 4	
No of units	
Learning Objectives:	1. Enable students to code in Python programming. 2. Enable students to manipulate data using Python. 3. Enable students to handle exceptions and errors in Python. 4. Enable students to acquire data from web applications via web scraping techniques. 5. Enable students to visualize data effectively using Python libraries.
Pre-requisite: Programming skill.	
1	Introduction and Basics I: Introduction, Features, Lexical Structure, Data Type, Variables, Numbers, Strings, Expression and Operators, Numeric Operations, Conditional Statements, Looping Python Installation, Basics of Python interpreter
2	Introduction and Basics II: Control flow Statements, Introduction to List and Tuple, Accessing List and Tuple, Operations, working with List and Tuple Build-in Function and Methods, Working with dictionaries, properties and Methods Set and Dictionary Data Structures
3	Functions and Objects I: Defining a function, calling a function, Types of functions, Function Arguments, Anonymous functions, Defining Your Own Functions, Scope, Passing Functions to a Function Recursion, Decorators
4	Functions and Objects II: Defining Class, Creating Object, Inheritance, Overloading and Overriding, Data Hiding, Polymorphism Modules and Packages
5	Files: File Objects, File Built-in Function, File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules Working with CSV, Excel files, and JSON data
6	Error and Exceptions: Exceptions in Python, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, Run Time Errors, The Exception Model, Exception Hierarchy, Handling Multiple Exceptions Unit Testing and Debugging
7	Data Wrangling and Data Aggregation: Combining and Merging DataSets, Reshaping and Pivoting, Data Transformation, String Manipulation, Regular Expressions, Data Aggregation, Group wise Operations and Transformations, Pivot, Tables and Cross Tabulations Basics of Data Cleaning, Data preprocessing
8	Web Scraping: Data Acquisition by Scraping web applications - Submitting a form - Fetching web pages, Downloading web pages through form submission, CSS Selectors, Introduction to





	BeautifulSoup, Web scraping using Selenium
9	Libraries in Python: NumPy - Array, Operators, Pandas- Series, Dataframes, Methods - iloc & loc and Use of zip SciPy, Seaborn
10	Pandas in Python: Importing data, Combining dataframes, Handling Strings, Data Manipulations, Groupby Operations & Handling dates Data Cleaning with Pandas
11	Visualization in Python I: Matplotlib package, Plotting Graphs, Controlling Graph, Adding Text, More Graph Types, Getting and setting values, Patches Seaborn for statistical data visualization
12	Visualization in Python II: Plotly and Dash, Bokeh, ggplot, Folium, NetworkX, 3D Visualization, Data visualization best practices Interactive Plots using Plotly

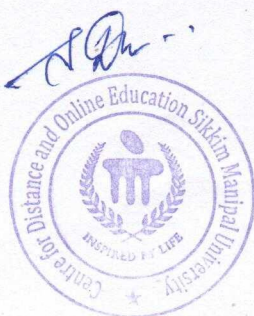
OMCA202 COMPUTER ORGANIZATION AND ARCHITECTURE; Credits: 4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to comprehend the knowledge of computer architecture and operational concepts. 2. Enable students to analyze and evaluate processing techniques. 3. Enable students to equip with a deep understanding of memory organization. 4. Enable students to evaluate the principles of high-speed memories and secondary memory technologies. 5. Enable students to utilize knowledge of peripherals, multiprocessing, and parallel organization.
Pre-requisites: Digital Circuits & Logic Design, PC hardware & peripherals.	
1	Basic structure of computers: Functional units, Basic operational concepts, Bus structures, Memory locations, Encoding of information, Main memory operations, Addressing modes, Program sequencing, Basic Input-Output operations, Memory reference, Register reference, Input-Output reference instructions, Register stack, Memory stack, Subroutines. Understanding of computer architecture, Introduction to microprocessors, Introduction to operating systems
2	The Processing Unit: Fetching a word from memory, Storing a word in memory, Register transfers, Performing an arithmetic or logic operation, Instruction format-Three, Two, One and Zero address instruction, Execution of a complete instruction, Control Unit-hardwired control, Micro programmed control. Introduction to parallel processing, Understanding of processor architectures
3	Arithmetic: Number representation, Fixed point addition and subtraction, Design of fast adders, Multiplication of fixed point numbers, Booth's multiplication, Integer division, Floating point numbers and operations." Comparison of arithmetic operations in different computer architectures, Implementation of arithmetic operations in hardware
4	Memory organization: A review of random and serial access memories, Basic concept of main memory-static and dynamic memory, Memory hierarchies-cache memory, Virtual memory, Logical to physical memory mapping. Modern memory technologies (DDR4, DDR5, HBM), Solid State Drive (SSD) technology
5	Input and output organization: Accessing I/O devices interrupt, Enabling and disabling interrupts, Handling multiple devices, Controlling device requests, Direct memory access, I/O hardware-processor bus, Interface circuits, Standard I/O interfaces. Introduction to USB, HDMI, and other I/O interfaces, I/O in modern operating systems
6	High Speed Memories: Cache Memory, Principles of cache memory, Structure of cache and main memory, Performance using cache memory, Elements of Cache Design, Mapping functions, Replacement algorithms, External Memory, Virtual memory, Memory Management in Operating Systems. Overview of modern cache architectures, Discussion on memory consistency models





7	Secondary Memory: Magnetic Disk and Tape, Digital Audio Tape (DAT), RAID, Optical memory. Understanding of modern storage technologies (NVMe, SSD), File systems overview
8	Processor Organization: Parallelism and Computer Arithmetic, Floating Point in the 8086, Programmers Model of 8086, Max/Min Mode, Register Organisation, Instruction Cycles, Read Write cycles, addressing Modes. Understanding of modern processor architectures (ARM, RISC-V), Power-efficiency considerations in processor design
9	Peripherals: Input Devices, Output Devices. Wireless peripherals, Introduction to Human Computer Interaction (HCI)
10	Multiprocessor Configuration: Multiprocessing, Coupled Multiprocessor, Contention problems in multiprocessor systems, Coprocessor, I/O Processor, Caselet. Discussion on multi-core and many-core architectures, Introduction to GPGPU (General-purpose computing on graphics processing units)
11	Microprogramming: Basic Principles, Computer Clock, Microinstructions and its timing, Control Path, Microcode, Machine Instructions. Overview of microcoding in modern processors, Comparisons between hardwired control and microprogrammed control
12	Parallel Organization: Parallel Organization, Instruction Set Architecture (ISA), Vector Processing Requirements, Characteristics of vector processing, Super Scalar Processors, The emergence and spread of super scalar processors, Specific task of Super scalar processing, Super Scalar Instruction Issue. Introduction to High Performance Computing (HPC), Discussion on thread-level and data-level parallelism

OMCA203 SOFTWARE ENGINEERING AND UNIFIED MODELLING LANGUAGE;	
No of units	Credits: 4
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to explain the basic principles of software engineering. 2. Enable students to apply their knowledge of software design, project management, and computer-aided software engineering. 3. Enable students to analyze various software engineering methodologies. 4. Enable students to evaluate the effectiveness of different project management strategies. 5. Enable students to create software systems using their knowledge of software engineering.
Pre-requisites: Algorithm concept and Database Management System.	
1	Introduction: The software engineering discipline-evaluation and impact, Computer system engineering Recent trends in software engineering, Ethics in software engineering
2	Life Cycle Models: SDLC phases, Feasibility study, Requirements gathering and analysis. Software requirement specification (SRS), Software Design Agile methodologies, DevOps. Continuous Integration and Continuous Deployment
3	Cohesion and coupling: Object oriented vs. function oriented design, user interface design, Characteristics of a good user interface, Basic concepts, Types of user interfaces, Coding, Code review, testing, types of testing, test cases design Design patterns, Clean code practices
4	Software project management: Responsibilities of project manager, Project planning, Metrics for project size estimation, Project estimation techniques, Scheduling, Organization and team structure, Staffing, Risk management, Software configuration management Agile project management, Software project cost estimation
5	Software Reliability and Quality Management: Software reliability, Statistical testing, Software



	quality management system, ISO 9000, SEI capability maturity model, Personal software process (PSP), Six sigma Testing methodologies, Automated testing, Continuous Quality Improvement
6	Computer Aided Software Engineering: Case and its scope, Case environment, Case support in software life cycle, Architecture of a case environment Introduction to software development tools, Integrated Development Environments (IDEs), Version control systems (Git)
7	Software maintenance & reuse: Characteristics of software maintenance, Software reverse engineering, Basics issues in any reuse program Refactoring techniques, Technical debt management
8	Introduction to Object Oriented Paradigm: Objects, Functional method and Object-Oriented method, the key concepts of OOP, Relationship among objects, Fundamentals of Object Oriented approach, various concepts SOLID principles, Dependency Injection, Advanced concepts in OOP
9	Object Oriented Analysis & Object Oriented Design: Rum Baugh et. al.'s object model, Booch Model, Jacobson et. al. methodology, Coad and Yourdon Object Oriented analysis model, Pattern & frameworks Design patterns, Principles of API design, Microservices architecture
10	UML:UML Background, Its evolution and various variants, UML building blocks and modeling views, requirements engineering using various UML tools such as Use Case Modelling, Conceptual modelling, Behavioural modelling Advanced UML diagram types, Model Driven Architecture (MDA)
11	Architecture: System Architecture using various UML tools such as collaboration diagrams, Component diagrams, Design of class diagrams, Activity diagrams, Sequence diagrams, State chart diagrams, Design patterns Software Architecture styles (Layered, Event-driven, Microservices, etc.), Architectural patterns
12	Agile Process: Introduction, Agile manifesto, Scrum, Kanban, Lean, User stories, Sprints, Incremental delivery, Retrospectives Extreme Programming (XP), Agile Estimation techniques, Agile scaling frameworks like SAFe and LeSS

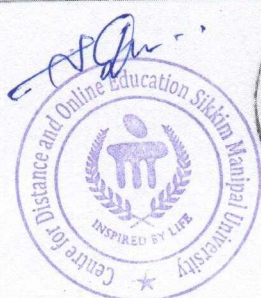
OMCA204 DATA STRUCTURE AND ALGORITHM; Credits: 4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to comprehend the foundational concepts of data structures and algorithms. 2. Enable students to develop the ability to implement different data structures like linked lists, stacks, queues, trees, and graphs. 3. Enable students to evaluate the efficiency of various searching and sorting techniques. 4. Enable students to design efficient solutions using advanced data structures and algorithms. 5. Enable students to recognize the principles of advanced topics.
Pre-requisites: Programming concepts and 'C' /Java programming language.	
1	Structure and Problem Solving, Data Structures, Data Structure Operations, Algorithm: Complexity and Time- Space Tradeoff: Abstract Data Types
2	Mathematical Notation and Functions, Algorithm Notation, Control Structures, Complexity of Algorithm, Rate of Growth: Asymptotic Notations, Recursion
3	Linked List and its representation in memory, Traversing a Linked List, Searching a Linked List, Memory Allocation and Garbage Collection, Insertion into Linked list, Deletion from a Linked list, Types of Linked List: Circular and Doubly Linked Lists
4	Stacks and Queues: Stack, Applications of Stack, Queue: Deque, Priority Queues



5	Definition and Concepts, Binary Tree: Definition and Concepts, Types of Binary Tree, Traversal on Binary Tree, Representation of Binary Tree, Binary Search Tree- Conversion of General Tree to Binary Tree, Sequential and Other Representations of Binary Tree, Operations on BST: Binary Heaps, Heap Operations
6	Definition and Structure of AVL Tree, Operations on AVL Tree, Definition and Structure of B-Tree, Operations on B-Tree, Applications of B-Tree: Red-Black Trees, Splay Trees
7	Basic Concepts about Graphs, Matrix Representation of Graphs, List Structures, Other Representations of Graphs, Algorithms for Graph Traversal, Spanning Trees: Graph Algorithms: Bellman-Ford, Floyd-Warshall
8	Topological Sorting, Weighted Shortest Path – Dijkstra's Algorithm, Minimum Spanning Tree (MST), Introduction to NP Completeness: Network Flow, Bipartite Graphs
9	Introduction, Memory Management, First-fit Storage Allocation, Storage Release, Buddy Systems, Garbage Collection: Dynamic Memory Allocation Techniques, Memory Pool
10	Sequential Searching, Binary Searching, Bubble sort, Merge sort, Selection sort, Heap sort: Quick Sort, Radix Sort
11	External Storage Devices, Introduction to File Organization, Sequential Files, Indexed Sequential Files, Direct Files: B+ Trees, File Compression
12	External Sorting, Sorting on Disks, Generating Extended Initial Runs, External Searching Techniques- External Searching, Introduction to Static Hashing, Organizing direct files with hashing, Collision Resolution Methods, Dynamic Hashing Techniques: Disk Scheduling Algorithms, Data Warehousing

OMCA205 DATA STRUCTURE AND ALGORITHMS LAB; Credits: 2	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Student will be able to design and analyze the time and space efficiency of the data structure. 2. Student will be able to identify the appropriate data structure for given problem. 3. Student will get hands on experience on the applications of data structures.
1	Design and Implement List data structure using i) array ii) singly linked list.
2	Design and Implement stack using i) array ii) singly linked list
3	Design and Implement Queue using i) array ii) singly linked list
4	Design and Implement basic operations(insertion, deletion, search, findmin and findmax) on Binary Search trees.
5	Implementation of Breadth First Search Technique and Depth First Search Techniques.
6	Implementation of Sorting methods i) Bubble Sort ii) Selection Sort iii) Merge Sort
7	Implementation of Sorting methods i) Sequential Searching ii) Binary Searching.

OMCA206 PYTHON PROGRAMMING LAB; Credits: 2	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. To get hands on experience of applying Python code for topics covered in theory subjects. 2. Enable students to code in Python for various usage such as Data Wrangling and Data Aggregation, Web Scraping, Visualization in Python etc.
1	Write a Python program to input an array of n numbers from keyboard and find separately the sum of positive numbers and negative numbers.
2	Write a Python Program to read a word and print the number of letters, vowels and percentage of vowels in the word using dictionary.
3	Write a Python program to create a class representing a shopping cart. Include methods for adding



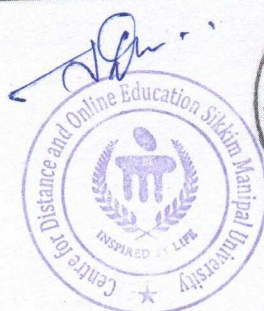


	and removing items, and calculating the total price.
4	Write a Python program that opens a file and handles a FileNotFoundError exception if the file does not exist.
5	Write a Python program to join the two given dataframes along rows and assign all data using Pandas.
6	Write a Python program to download all Images from a given Web Page.
7	Write a NumPy program to find the roots of the following polynomial equation. a) $x^2 - 4x - 12$. b) $x^4 - 11x^3 + 9x^2 + 11x - 10$
8	Write a Python program to draw a horizontal bar chart with Matplotlib for any items sold against its price for a vendor.

SEMESTER III

OMCA301 .NET FRAMEWORK; Credits: 4

No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to apply the architecture and components of the .NET Framework. 2. Enable students to apply the concepts of object-oriented programming in VB.NET and C#. 3. Enable students to analyze different approaches to data access in .NET. 4. Enable students to evaluate different techniques for state management in web applications and design solutions using appropriate methods. 5. Enable students to create multi-threaded applications using the Task Parallel Library.
Pre-requisites: Basic programming skill on C, C++ and Java.	
1	Overview of the .NET Framework: Common Language Runtime (CLR), NuGet, Visual Studio .NET Core, .NET 5 and .NET 6
2	VB.NET Introduction: Syntax, Object-Oriented Programming (OOP) in VB.NET, Windows Forms Data binding in Windows Forms
3	Introduction to C#: Understanding .NET, Overview of C#, Literals, Variables, Data Types, Operators, Expressions, Branching, Looping, Methods, Arrays, Strings, Structures, Enumerations Lambda expressions, Extension methods, LINQ
4	Object Oriented Aspects of C#: Classes, Objects, Inheritance, Polymorphism, Interfaces, Operator Overloading, Delegates, Events, Errors and Exceptions Indexers, Anonymous types, Tuples
5	Data Access using ADO.NET: Characteristics of ADO.NET, Comparing ADO and ADO.NET, Creating a connection, Select command, Using a command with a data reader, Updating data, Accessing, modifying, updating disconnected data, Selecting multiple tables Entity Framework, Dapper, LINQ to SQL
6	Web Based Application Development on .NET: Programming Web Applications with Web Forms, Programming Web Services ASP.NET MVC, ASP.NET Web API, ASP.NET Core
7	State Management: The Problem of state, Viewstate, Query String, Cookie, Session state configuration, Application state Caching, ViewState vs Session vs Application vs Cache
8	Files, Streams and Email: Getting file system information in ASP.NET, Reading, writing files with streams, File Uploading, Sending mail in ASP.NET Secure file upload, Email validation, Sending email with attachments



9	Exception Handling I: Basic Concept, Type of exception, Exception class, Try, catch, final, throw, Checked and unchecked, Creating user defined exception Logging exceptions, Debugging, Tracing
10	Exception Handling II: Throwing exceptions, Handling unhandled exceptions, Exception handling best practices, Exception filters, Exception Policy Global error handling, Custom error pages
11	Thread Management I: Process, Thread, Multithreading, Thread class mutex class, Creating and starting thread, Scheduling a thread, Synchronizing threads, Thread pooling Asynchronous programming, Task-based asynchronous pattern
12	Thread Management II: Task Parallel Library (TPL), Asynchronous Programming, Thread-safe collections, Thread-local storage, Thread Priority, Thread Affinity Async and await, Task continuation, Parallel LINQ

No of units	OMCA302 COMPUTER NETWORK; Credits: 4
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to explain the fundamental concepts of each layer in the OSI and TCP/IP models. 2. Enable students to implement the usage of different network protocols, addressing schemes, and routing algorithms in network configurations. 3. Enable students to analyze network traffic and connectivity issues using appropriate tools and techniques. 4. Enable students to compare the performance of different network designs, protocols, and routing algorithms.
Pre-requisites: Communication Technique and Data Communication.	
1	Introduction: Definition of Computer Network, Data Communication System Model, Network Categories, Network Topologies, Network Reference Models: ISO-OSI Model, TCP/IP Model, Physical layer: Transmission Modes, Transmission Media, Transmission Impairments, Switching Techniques Concepts of Network Security
2	Physical Layer: Analog and Digital Signals, Transmission Impairments, Transmission Media-Guided and Unguided Guided Media: Twisted Pair, Coaxial, and Fiber Optics Cables, Radio Waves, Microwaves, Infra-red. Signal Encoding Techniques
3	Data Link Layer: Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Protocol: Simplex, Stop-and-Wait ARQ, Sliding Window ARQ Protocol: One Bit Sliding Window ARQ, Go-back-N ARQ, Selective Repeat ARQ. Virtual Circuit and Datagram Networks
4	Medium Access Control Sub Layer: Multiple Access Control (MAC) Protocol: ALOHA, Slotted ALOHA, CSMA Protocols. Wireless LANs, Bluetooth
5	Network Layer I: Network Layer Design Issue, Addressing: IP Address, Classless and Class full Addressing, Routing: Definition, Static vs, Dynamic Routing, Routing Table. IPv6
6	Network Layer II: Routing Algorithm: Introduction, Optimality Principle, Shortest Path, Flooding, Distance Vector Routing, Link State Routing, General Principle of Congestion Control: Leaky Bucket Algorithm, Token Bucket Algorithm. NAT
7	Sub Netting and Super Netting: Need of Sub Network, Hierarchical Addressing Scheme, Role of Gateway, Subnet Mask in Packet Transmit. VLANs
8	Network Layer in the Internet: Internet Protocol: IPv4 protocol and IPv6 protocol, IPv4 Addressing: ARP, RARP, DHCP and mobile IP, Internet Routing Protocols: RIP, OSPF, BGP, and ICPS MPLS

Signature



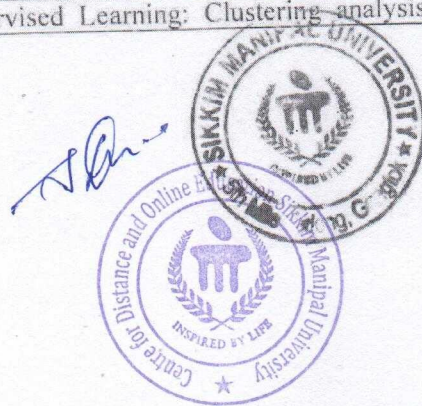
9	Transport Layer I: The Transport Services: Services Provided to Upper Layer, Transport Service Primitives, Element of transport protocols, Congestion Control. SCTP
10	Transport Layer II: Introduction, TCP Service Model, TCP Protocol, TCP Segment Header, TCP Connection Establishment, TCP Connection Release, TCP Connection Management, Transmission Policy, Congestion Control, UDP- Introduction, RPC, RTTP. Quality of Service (QoS)
11	Application Layer I: DNS, Email- SMTP, POP3, World Wide Web (WWW) SMTP
12	Application Layer II: HTTP, Mobile Web, FTP, TELNET, P2P Network. Multimedia Applications

No of units	OMCA303 IT LAWS AND PRACTICES; Credits: 4
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to comprehend and critically analyze the design constraints, requirements, and technologies underlying the creation of mobile applications. 2. Enable students to apply the principles of user interface design and software architecture to develop mobile applications. 3. Enable students to differentiate between native and cross-platform app development and analyze the strengths and weaknesses of each approach. 4. Enable students to implement security measures, testing, and performance optimization techniques. 5. Enable students to organize fully functional mobile application incorporating emerging technologies.
Pre-requisites: Fundamentals of IT and E-commerce.	
1	Introduction to Information Security I: History of Information Security, Characteristics of Information, NSTISSC Security model, Components of Information System, Security Systems Development life cycle, Security professional.
2	Introduction to Information Security II: The Information Technology Act, 2000 (No. 21 of 2000). Intellectual Property, Understand how copyrights, patents, trademarks and other forms of IP differ. Cognizable and non-cognizable offence. Necessity of Arrest without Warrant under IT Act 2000.
3	Need for Security: Business needs, Threats, Attacks. Information Security Risk Management: An Overview of Risk Management, Risk Identification Identifying assets, Threats and vulnerabilities.
4	Risk Analysis and Assessment: Understanding risk analysis, planning for risk analysis, Performing risk analysis and assessment, Risk Control Strategies and selection.
5	Cyber Security and practices: Security problem in computing: definition of secure, Attacks, the meaning of computer security, computer criminals, methods of defence.
6	Planning for Security :Information security policy, Standards and practices, the information security blueprint, security education, and Training and awareness program.
7	Implementing Information Security: Project Management for Information Security, Technical topics, Nontechnical aspects. Disaster Recovery and Risk Monitoring: Disaster recovery planning, Disaster recovery plan , Risk monitoring Application-Level Risks the Fundamental hacking concept.
8	Network Vulnerability: Network service vulnerabilities and attacks memory manipulation attacks, Mitigation process manipulation risks.
9	Security Technology: Introduction Access control, Firewall, protecting Remote connections-



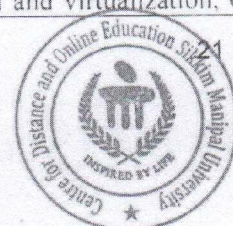
	Remote Access and Virtual Private Networks (VPNs) Intrusion Detection and prevention system: IDPS Terminology, need of IDPS. Types of intrusion detection system.
10	Network Security Assessment and Tools Required: The business benefits, Classifying internet-based attackers, Assessment service definitions, Network security methodology, The cyclic assessment approach, Free and commercial network scanning tools.
11	Understanding Different Internet Laws: Data protection and privacy laws, Intellectual property laws, E-commerce laws. Internet censorship laws, Data Retention Laws, Child Online Protection laws, Online Harassment laws, Net Neutrality laws. Additional in-depth study of case laws related to these laws could be included.
12	Cyber Crime Laws and Forensic Practice: Introduction to cyber crimes, Types of cyber crimes, Prevention measures for cyber crimes, Cyber crime laws, Basics of Cyber Forensics, Tools and Techniques, Steps involved in Cyber Forensic Investigation. Case studies on cyber crime and forensics.

OMCA3041 Elective I: DATA WAREHOUSING AND DATA MINING; Credits: 4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to understand the principles, techniques, and applications of data warehousing. 2. Enable students to apply data mining techniques to extract knowledge from large data sets, recognizing the issues and challenges that may arise in the process. 3. Enable students to evaluate the quality of data and perform necessary pre-processing steps. 4. Enable students to analyze different types of data, including structured, semi-structured, and unstructured data, using appropriate data mining techniques and tools. 5. Enable students to design effective extract, transform, load (ETL) processes.
Pre-requisites: Data structures, Concepts of Algorithms, Fundamentals of probability & statistics and programming skills	
1	Data Warehousing: Introduction, Definition, Multidimensional data transformation, OLAP operations, Warehouse schema, Warehouse Server, Meta data, OLAP engine Application of Data Warehousing in real-world scenarios
2	Data Mining: Introduction, Definition, KDD vs. DM, DBMS vs. DM, DM techniques, Issues and challenges in DM, DM applications Deep Dive into Data Mining Algorithms
3	Data Types: Data objects and attributes types, basic statistical descriptions of data, data visualization mechanisms, measuring data similarity and dissimilarity Advanced Data Visualization Techniques
4	Data Pre-processing: Data Quality, Major tasks of data pre-processing, Data Cleaning, Data Integration, Data Reduction, Data transformation and discretization Applications and Case Studies of Data Pre-processing
5	ETL Basics: ETL Basics, Need for ETL, ETL Process, Operational Considerations Practical aspects and challenges of ETL
6	ETL Requirements: ETL Requirements, ETL Architecture, ETL Data Structures ETL tools and their comparison
7	Unsupervised Learning: Clustering analysis, Types of data in cluster analysis, Partitioning



	algorithms, Hierarchical, Density based, Grid based, Model based algorithms, High dimensional & Categorical data clustering Case Studies and Applications of Unsupervised Learning
8	Supervised Learning: Basic Concepts, Decision tree induction, Tree pruning, Extracting classification rules from decision trees, Decision tree construction algorithms, Decision tree construction with pre-sorting Other Supervised Learning Algorithms like SVM, Naive Bayes, etc.
9	Outlier Detection: Outliers and Outliers analysis, Outlier detection methods, statistical approaches, proximity-based approaches, clustering and classification-based approaches, Outlier detection in high dimensional data Case Studies and Real-world applications of Outlier Detection
10	Unstructured Data Mining: Text mining, Web mining, spatial data mining, Multimedia data mining Dealing with Semi-structured Data
11	Data Mining Trends & Research: Frontiers Mining complex data types, other methodologies of data mining Future Predictions for Data Mining
12	Steps in text mining: Representation of text documents, preprocessing techniques, feature selection, constructing a vector space model, predicting and validating the text classifier.

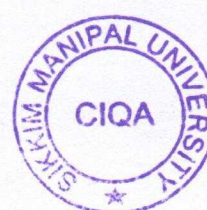
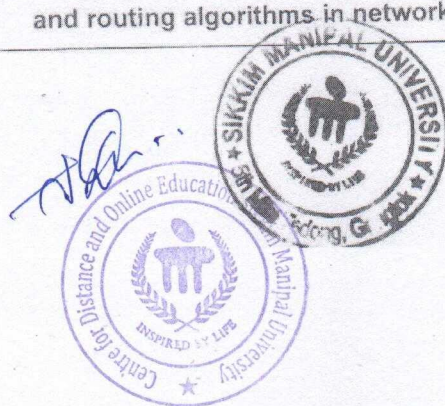
No of units	OMCA3042 Elective I: CLOUD COMPUTING; Credits: 4
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to explain the fundamental principles and models of Cloud Computing including IAAS, PAAS, AND SAAS. 2. Enable students to apply the concepts of virtualization and cloud services in the creation and management of cloud environments. 3. Enable students to evaluate different cloud platforms like Microsoft Azure, Amazon Web Services (AWS), and Google Cloud Platform for suitability in various use-case scenarios. 4. Enable students to analyze security risks in the cloud, demonstrating proficiency in implementing cloud security measures such as Identity and Access Management (IAM), encryption, and vpns. 5. Enable students to develop strategies for business transformation and innovation using cloud technologies.
Pre-requisites: Operating Systems and Computer Network	
1	Introduction to Cloud Computing I: Introduction- Definition, Characteristics, Components, Applications, Pros and cons, Limitations. Need for cloud computing. History/ Evolution of cloud and related technologies- Multi-processing, Distributed computing, Parallel computing to ubiquitous computing.
2	Introduction to Cloud Computing II: What cloud computing really is and what really isn't? Importance of cloud computing in current era- why cloud computing matters? Who should use cloud computing and who shouldn't use it? Types of cloud computing. Major players in cloud computing.
3	Cloud Computing Platforms and Technologies: Migrating into the cloud platform- Issues and deployment considerations. Cloud Reference Model, NIST Cloud model, Exploring cloud service models- IaaS, PaaS, SaaS, IDaaS, STaaS, CaaS, and other service models. Deployment models: Private, Public, Community and Hybrid clouds, Cloud-cube model. Cloud computing platforms- Microsoft Azure, Hadoop, Map-reduce, Amazon Web services (AWS).
4	Cloud Computing Architecture: Service Level Agreement (SLA), Understanding cloud architecture, Service Oriented Architecture (SOA), Cloud abstraction and virtualization, Cloud



	federation, Cloud meshup, Cloud life cycle.
5	Virtualization I: Virtualization- definition, Implementation levels of virtualization, Virtualization architecture and software. Virtual clustering, Virtual Infrastructure, Virtualization of CPU, Memory.
6	Virtualization II: I/O Devices, Application virtualization, Virtualization for data-centre automation. Virtualization and cloud computing, Migrating virtual machines, Pros and cons of virtualization. Para-virtualization, Full-virtualization. Examples- Xen, Microsoft Hyper-V, VMware.
7	Developing Cloud Services I: Web-based application, Pros and cons of cloud service development, Types of cloud service development, cloud development life cycle.
8	Developing Cloud Services II: Discovering cloud services, Development services and tools, Amazon EC2, Amazon Web services (AWS), Google app engine, IBM blue cloud, Microsoft Azure, Hadoop, Map-reduce, Force.com.
9	Cloud Security: Identity and Access Management (IAM), Encryption, Secure Virtual Private Network (VPN), Cloud-based Firewalls, Cloud Security Alliance (CSA), Secure Multi-Tenancy.
10	Cloud Compliance: Cloud Service Provider (CSP) compliance, Data sovereignty, Data Privacy, Security, Auditing and Reporting, Compliance automation, Compliance as Code, Cloud Compliance in Multi-cloud and hybrid cloud.
11	Cloud Native and Cloud-Native Applications: Microservices architecture, Containerization, Kubernetes, Service Mesh, Serverless computing, Cloud-native development, Cloud-native monitoring and observability, Cloud-native security.
12	Cloud Computing Business Transformation and Strategy: The Role of Cloud Computing in Business Transformation, Business Strategies for Cloud Adoption, Cost Analysis and ROI of Cloud Computing, Case Studies of Successful Cloud Transformation, Cloud Computing and Digital Innovation, Strategic considerations for Multi-cloud and Hybrid cloud deployments, Managing Change in Cloud Transformation.

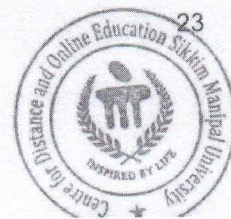
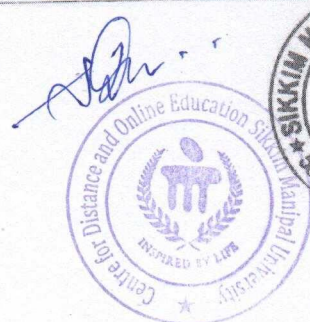
OMCA305 .NET FRAMEWORK LAB; Credits: 2	
No of units	
Learning Objectives:	1. To get hands on experience on C# programming 2. To get used to the .Net framework.
1	Write a C# Sharp program to print the sum of two numbers entered by the user.
2	Write a C# Sharp program to find the largest of three numbers using if else statement.
3	Write a C# Sharp program to find the sum of the first 10 natural numbers using for loop.
4	Write a program in C# Sharp to count the total number of words in a string.
5	Write a C# program that prompts the user to input a numeric integer and throws an exception if the number is less than 0 or greater than 999. Demonstrate the use of Try, catch, final exception.
6	Write a program in C# to create a blank file on the disk and demonstrate to delete it if the file already exist.
7	Write a C# program to illustrate the Binary Operator Overloading.
8	Write a C# program to illustrate the use of Derived and Base Class.

OMCA306 COMPUTER NETWORK LAB; Credits: 2	
No of units	
Learning Objectives:	1. To get hands on experience on the network layer of OSI and TCP/IP models using appropriate tools and techniques. 2. Enable students to simulate different network protocols, addressing schemes, and routing algorithms in network configurations.



3. Enable students to design networks using network simulation tools.	
1	Study and execution of the basic networking commands.
2	Write a socket program for implementation of echo.
3	Write a client-server application for chat using TCP
4	Perform File Transfer in Client & Server Using TCP/IP.
5	Implement Remote Command Execution(RCE).
6	Write a program to implement simple client-server application using UDP
7	Implement Address Resolution Protocol .
8	Write a socket program for implementation of TCP module.

SEMESTER IV	
OMCA401 MOBILE APPLICATION DEVELOPMENT; Credits: 4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to comprehend and critically analyze the design constraints, requirements, and technologies underlying the creation of mobile applications. 2. Enable students to apply the principles of user interface design and software architecture to develop mobile applications. 3. Enable students to differentiate between native and cross-platform app development and analyze the strengths and weaknesses of each approach. 4. Enable students to implement security measures, testing, and performance optimization techniques. 5. Enable students to create a fully functional mobile application incorporating emerging technologies.
Pre-requisites: xml, html, Java/Kotlin	
1	Introduction: Introduction to mobile applications, Embedded systems, Market and business drivers for mobile applications, Publishing and delivery of mobile applications, Requirements gathering and validation for mobile applications
2	Basic Design I: Introduction to basics of embedded systems design, Embedded OS, Design constraints for mobile applications, Architecting mobile applications Understanding of Hardware and Software Requirements
3	Basic Design II: User interfaces for mobile applications, touch events and gestures, Achieving quality constraints, performance, usability, security, availability and modifiability UI/UX Best Practices
4	Advanced Design: Designing applications with multimedia and web access capabilities, Integration with GPS and social media networking applications, Accessing applications hosted in a cloud computing environment, Design patterns for mobile applications Advanced Design Patterns
5	Android Development Environment I: Introduction to Android Environment, Establishing the development environment, Android architecture, Activities and views, Interacting with UI, Persisting data using SQLite Android Development Best Practices
6	Android Development Environment II: Packaging and deployment, Interaction with server side applications, Using Google Maps, GPS and Wifi – Integration with social media applications Android Hardware Access and Permissions
7	Mobile App Development Best Practices: Native app development, Cross-platform app development, Mobile user interface design, Mobile app security, Mobile app testing, Mobile app performance optimization Hybrid App Development



8	Mobile App Business Aspects: Mobile app monetization, Mobile app analytics, Mobile app push notifications Mobile App Marketing and User Retention
9	iOS Development Environment: Introduction to Objective C, iOS features, UI implementation, Touch frameworks, Data persistence using Core Data and SQLite, Location aware applications using Core Location and Map Kit, Integrating calendar and address book with social media application, Using Wifi - iPhone marketplace iOS Development Best Practices
10	Advanced iOS Development: Swift and Objective-C, iOS SDK and Xcode, Cocoa Touch, UIKit and AutoLayout, Core Data, AVFoundation, SwiftUI iOS Hardware Access and Permissions
11	App Deployment and Maintenance: App Store Optimization, App Versioning, Handling User Feedback, App Updates and Maintenance, Ensuring App Security Post-Deployment Understanding of DevOps in Mobile App Development
12	Emerging Technologies in Mobile Apps: Overview of Augmented Reality (AR) and Virtual Reality (VR) in mobile apps, Use of Artificial Intelligence (AI) and Machine Learning (ML) in apps, Blockchain-powered apps, Introduction to IoT in mobile apps Current Trends and Innovations in Mobile App Development

No of units	OMCA402 ANGULAR JS, REACT JS AND VUE JS; Credits: 4
Learning Objectives:	<ol style="list-style-type: none"> 1. Enable students to explain the fundamental concepts, architecture, and functionality of Angular JS, React JS, and Vue JS. 2. Enable students to develop functional components, apply data binding, handle events, and manage state changes across different javascript frameworks. 3. Enable students to distinguish use cases for Angular, React, and Vue, and determine appropriate use cases. 4. Enable students to evaluate and optimize web applications using lifecycle hooks, state management, and two-way data binding. 5. Enable students to create a full-featured web application utilizing learned frameworks (Angular, React, or Vue).
Pre-requisites: HTML basics, Programming concepts.	
1	Introduction to Angular JS: Introduction to Angular, Architecture of Angular Apps, Setting Up the Development Environment, Structure of Angular Projects, Webpack, Angular Version History.
2	Introduction to TypeScript: TypeScript Fundamentals, Declaring Variables, Types, Type Assertions, Arrow Functions, Interfaces, Classes, Objects, Constructors, Access Modifiers, Access Modifiers in Constructor Parameters, Properties, Modules.
3	Angular JS Fundamentals: Building Blocks of Angular Apps, Creating Components, Generating Components Using Angular CLI, Templates, Directives, Services, Dependency Injection, Generating Services Using Angular CLI.
4	React JS Fundamentals I: What is React, Setting Up the Development Environment, Your First React App, Components, Setting Up the Project.
5	React JS Fundamentals II: Your First React Component, Specifying Children, Embedding Expressions, Setting Attributes, Rendering Classes Dynamically, Rendering Lists, Conditional Rendering.
6	React JS Fundamentals III: Handling Events, Binding Event Handlers, Updating the State, What



	Happens When State Changes, Passing Event Arguments, Composing Components, Passing Data to Components, Passing Children.
7	React JS Fundamentals IV: Debugging React Apps, Props vs State, Raising and Handling Events, Updating the State, Single Source of Truth, Removing the Local State, Multiple Components in Sync, Lifting the State Up, Stateless Functional Components, Destructuring Arguments, Lifecycle Hooks, Mounting Phase, Updating Phase, Unmounting Phase.
8	Vue JS Fundamentals I: Introduction, The Vue Instance, Data & Methods, Data Binding, Events, Event Modifiers, Keyboard Events, Two-Way Data Binding.
9	Vue JS Fundamentals II: Computed Properties, Dynamic CSS Classes, Conditionals, Looping with v-for, Multiple Vue Instances, Intro to Components, Refs.
10	Vue JS Fundamentals III: The Vue CLI, Vue Files & The Root Component, Nesting Components, Component CSS, Nesting Components Examples.
11	Vue JS Fundamentals IV: Props, Primitive vs Reference Types, Events, The Event Bus, Lifecycle Hooks, Slots, Dynamic Components, Input Binding, Checkbox Binding, Select Box Binding, HTTP Requests.
12	Vue JS Fundamentals V: GET Requests, Custom Directives, Filters, Custom Search Filter, Registering Things Locally, Mixins, Setting up Routing, Hash vs History (Routing), Adding Router Links, Route Parameters, Posting to Firebase, Retrieving Posts from Firebase.

OMCA4031 Elective II: MACHINE LEARNING; Credits: 4	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. To enable students explain fundamental concepts of machine learning algorithms. 2. To enable students apply the principles of machine learning to design, implement, and evaluate algorithms for real-world datasets. 3. To enable students evaluate different machine learning methods based on their assumptions, applicability, and performance. 4. To enable students create machine learning models for different tasks. 5. To enable students synthesize knowledge from various machine learning techniques.
1	Introduction: Machine learning and its applications, Analysing Supervised and unsupervised learning, Designing a learning system, Issues of machine learning Importance of data in Machine Learning, Ethical considerations in Machine Learning
2	Decision Tree Learning: Decision tree representation, Characteristics of decision trees, Basic decision tree algorithm. Inductive bias in decision tree Decision Trees for Regression
3	Decision Tree II: Avoiding Overfitting the Data, Reduced error pruning, Rule post pruning. Incorporating Continuous-Valued Attributes, Handling Training Examples with Missing Attribute Values, Handling Attributes with Differing Costs Decision Trees in Ensemble Methods
4	Artificial Neural Networks: Biological motivation, Neural network representations, Appropriate problems for neural network learning, Perceptron, Multilayer networks and the backpropagation, Hidden Layer Representation Deep Learning Concepts, Convolutional Neural Networks, Recurrent Neural Networks
5	Bayesian Learning: Bayes Theorem, Concept learning, Maximum likelihood and least-squared error hypotheses, Maximum likelihood hypotheses for predicting probabilities. Minimum description length principle Advanced Bayesian Techniques, Gaussian Mixture Models. Hidden Markov Models
6	Bayesian Learning II: Bayes optimal classifier, Gibbs algorithm, Naïve Bayes Classifier, Bayesian



	belief network, The EM algorithm Bayesian Optimization, Dirichlet Processes
7	Instance-based Learning: k-Nearest neighbour learning, Distance weighted nearest neighbour algorithm, Locally Weighted Regression, Radial Basis Function, Case based reasoning, Lazy vs Eager Learning Advanced Nearest Neighbors Techniques, KD-trees, Ball-trees
8	Genetic Algorithm: Basic Concepts and the algorithm, Genetic operators, Fitness function and selection, Hypothesis space search, Lamarckian Evolution, Baldwin Effect Multi-Objective Genetic Algorithms, Coevolution
9	Learning sets of rules: Sequential covering algorithms, Learn one rule algorithm, Evaluation functions of learn one rule algorithm, Learning first order rules, learning sets of first-order rules: FOIL, Guiding the Search in FOIL, PROLOG Rule learning in the context of other types of learning, Inductive Logic Programming
10	Analytical Learning: Basic concepts, Inductive and Analytical Learning Problems, Learning with perfect domain theories: PROLOG-EB, Properties of explanation-based learning, Explanation-based learning of search control knowledge Reinforcement Learning, Multi-Agent Learning
11	Three Basic Machine Learning Algorithms: Linear Regression, k-Nearest Neighbors (k-NN), k-means, Motivating application: Filtering Spam, Why Linear Regression and k-NN are poor choices for Filtering Spam, Naive Bayes and why it works for Filtering Spam, Data Wrangling: APIs and other tools for scrapping the Web Logistic Regression, Support Vector Machines, Ensemble Methods
12	Recommendation Systems: Algorithmic ingredients of a Recommendation Engine, Dimensionality Reduction, Singular Value Collaborative Filtering, Content-Based Recommendation Systems, Hybrid Recommendation Systems

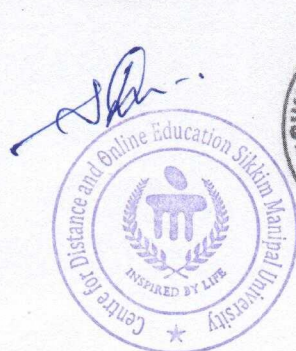
No of units	OMCA4032 Elective II: DISTRIBUTED SYSTEM AND GRID COMPUTING; Credits: 4
Learning Objectives:	<ol style="list-style-type: none"> 1. To enable students explain the fundamental concepts, challenges, and trends in distributed systems. 2. To enable students apply the principles of inter-process communication, remote method invocation, synchronization, and replication to solve problems. 3. To enable students analyze the architecture and protocols of peer-to-peer systems, grid computing, and distributed databases. 4. To enable students evaluate and select appropriate strategies for transaction and concurrency control. 5. To enable students design secure distributed systems by implementing suitable security measure.
Pre-requisite: Networking, Operating systems, Distributed system.	
1	Introduction to the Distributed Systems: Introduction – Examples of Distributed Systems–Trends in Distributed Systems – Focus on resource sharing – Challenges. Case study: World Wide Web
2	Communication in Distributed System: System Model – Inter process Communication, The API for internet protocols, External data representation and Multicast communication. Network virtualization: Overlay networks. Case study: MPI
3	Remote Method Invocation and Objects: Remote Invocation -- Introduction - Request-reply protocols - Remote procedure call - Remote method invocation. Case study: Java RMI - Group communication - Publish-subscribe systems - Message queues - Shared memory approaches -




	Distributed objects - Case study: Enterprise Java Beans -from objects to components
4	Peer to Peer services and File system: Introduction to P2P systems, Overlays, Unstructured P2P systems (Gnutella, Freenet), Structured P2P systems (Distributed, Hash Tables - Chord, Pastry), Integrating unstructured and structured , P2P system
5	Synchronization and Replication: Introduction - Clocks, events and process states - Synchronizing physical clocks- Logical time and logical clocks, Global states, Coordination and Agreement, Introduction, Distributed mutual exclusion, Elections
6	Transactions and Concurrency Control: Transactions ,Nested transactions, Locks, Optimistic concurrency control, Timestamp ordering , Atomic Commit protocols, Distributed deadlocks Case study :Coda
7	Introduction to Grid Computing: Introduction to Grid Computing, Classification of Grids, Introduction to Service Oriented Computing
8	Grid Computing Middleware: Functions/Challenges of a Grid computing middleware Globus: Open source software toolkit used for building Grid systems and applications Vishwa: A reconfigurable P2P middleware for Grid computations
9	Replication: Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data Distributed shared memory, Design and Implementation issues, Consistency models
10	File Systems and HADOOP Case Study: Introduction – File Models – File accessing, sharing and caching – File Replication – Atomic transactions Case Study HADOOP. : Resource and process management – Task assignment approach – Load balancing approach – Load sharing approach
11	Distributed Database Systems: Distributed Database Concepts, Types of Distributed Databases, Data Replication, Data Fragmentation, Distributed Database Architecture, Distributed Database Design, Distributed Query Processing and Optimization, Distributed Transactions Management, Concurrency Control, and Recovery in Distributed Database Systems
12	Distributed Systems Security: Security Challenges in Distributed Systems, Types of Threats and Attacks in Distributed Systems, Cryptography and Encryption Techniques, Authentication and Authorization in Distributed Systems, Secure Communication in Distributed Systems, Intrusion Detection Systems, Blockchain and Distributed Ledger Technology

OMCA405 PROJECT; Credits: 8	
No of units	
Learning Objectives:	<ol style="list-style-type: none"> 1. To make students industry ready 2. To enable students apply the knowledge in handling and completing the project work of their area and interest. 3. To prepare students as Programmers, Systems Analysts, Systems Designers etc. in the field related to computer science and information technology.

Students can take up a project work that are related to the development of software applications in the field of computer science or Information Technology. (Refer to Project manual for more details)



5.3 Duration of the programme

Programme	Level	Duration	Maximum Duration for Completion	Credits
MCA	Master's Degree	2 Years	(2+ 2) years (As per UGC Notification on Specification of Degree, 2014)	80

5.4 Faculty and support staff requirement

Academic Staff	Number available to meet the required delivery norms
Programme Coordinator	1 member
Course Coordinator	1 member
Course Mentor	1 member per batch of 250 students

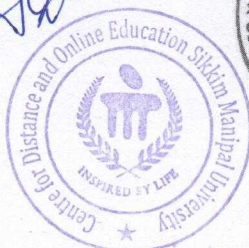
5.5 Instructional delivery mechanisms

The Centre for Distance and Online Education (CDOE), Sikkim Manipal University comprises of faculty members and staff who are well versed in Distance Education and Online delivery.

An Academic calendar depicting dates for all major events during each semester will be prepared by faculty members and shared with students through LMS, at the beginning of each academic session.

Apart from providing content in the form of Self Learning Material, enough e-learning resources in the form of Audio and Video content will be provided to students. Regular engagement of students will be ensured through the following means:

- Conduct of webinars/live lectures/online lectures/Virtual Class
- By encouraging them to participate in mandatory Discussion Forums to stimulate their thinking, and to be able to fearlessly express their views in forums. These discussion forums will be moderated by faculty to provide equal opportunity for everyone to participate, as well as to ensure maintenance of decorum of the forum.
- Through periodic formative assessments

Regular evaluation of content learnt will be provided for, through Self-Assessment Questions within the SLM, as well as quizzes on the LMS. The quizzes can be taken any number of times, so that they reach a stage of being able to answer questions without errors, which reflects their understanding of the concept.

Effort will be made to provide case studies to enhance their analytical ability and make right decisions.

Link to National Portals (SWAYAM/NPTEL) will be provided, as also link to University's digital library portal.

All links to additional reading will be provided in the LMS. Interested students can study beyond the confines of the syllabus.

5.6 Identification of media—print, audio or video, online, computer aided

LMS provides for all audio video content (e-learning material, e-pubs, faculty-led video sessions, virtual classrooms and discussion boards), dashboard of their progress in learning, comparison with their peers in terms of learning, regular notifications regarding upcoming Webinars/virtual classes, Assignments, Discussion Forum participations and Examinations. It also provides an opportunity for raising queries if any, and seeking answers to the same, by chat bot or course mentors.

5.7 Student Support Services

The Student Support services will be facilitated by the Centre for Distance and Online Education (CDOE), Sikkim Manipal University which includes the pre-admission student support services like counselling about the programme including curriculum design, mode of delivery, fee structure and evaluation methods. Post-admission student support services include guiding students towards accessing e-identity card, LMS portal, Academic calendar and academic delivery. Examinations support staff shall answer queries pertaining to the conduct of end-semester examinations, evaluation and issue of certificates.

6. Procedure for Admission, Curriculum Transaction and Evaluation

The purpose of Centre for Distance and Online Education (CDOE), Sikkim Manipal University, is to provide flexible learning opportunities to students to attain qualification, wherever learners are not able to attend the regular classroom teaching. Academic programmes offered for such candidates under Online Learning mode will be conducted by Centre for Distance and Online Education (CDOE).



Sikkim Manipal University, with support of the various University schools. The programmes/courses may be termed Online mode for award of Degree. Eligibility criteria, programme/course structure, curriculum, evaluation criteria and duration of programme shall be approved by Board of Studies and Academic Council which are based on UGC guidelines.

Candidates seeking admissions in any programme offered by Centre for Distance and Online Education (CDOE), Sikkim Manipal University, shall fill up an online application form available on CDOE- SMU website. Before applying, candidates must check eligibility criteria for the programme that they are interested in. Details about Eligibility criteria, programme structure, curriculum, duration, and fee structure are available on the website.

6.1. Procedure for Admission

6.1.1 Minimum Eligibility Criteria for admission

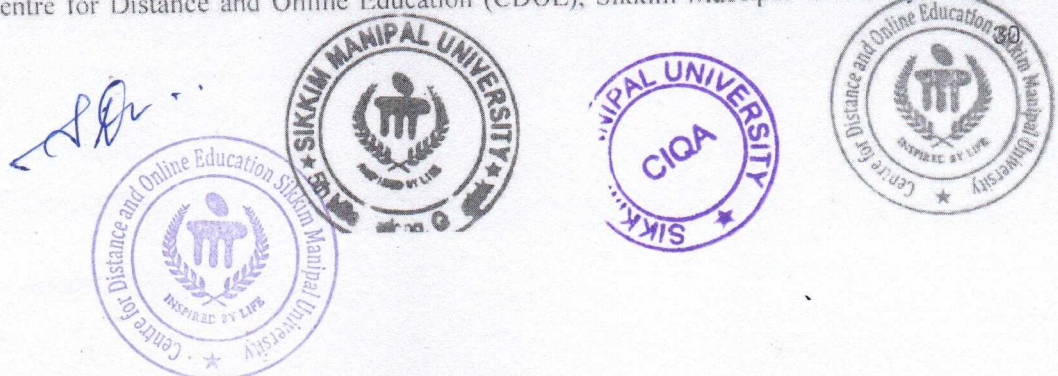
Candidate must have a 10 + 2 + 3 years bachelor degree from recognized University/ Institution or equivalent qualification as recognized by Association of Indian Universities (AIU) or other competent body in Computer Applications/Computer Science/Information Technology with a minimum 50% (45% for Reserved category) marks in aggregate.

Candidates from other streams like Science/Business Administration/ Business management/ Arts & Humanities/Commerce should have completed 10 + 2 + 3 years bachelor degree from recognized University or equivalent qualification as recognized by Association of Indian Universities (AIU) or other competent body with Mathematics at 10+2 level with a minimum 50% (45% for Reserved category) marks aggregate in graduation.

Such candidates need to attend and complete Bridge Course in Fundamentals of Mathematics and Fundamentals of Computer and IT along with their Semester 1 courses

Important Instructions:

- All admissions shall be provisional until and unless candidates meet the eligibility criteria.
- Admission will stand cancelled if a candidate does not meet eligibility criteria, or there is failure to pay programme/course fees.
- Admission will stand cancelled, if candidate does not submit proof of eligibility within stipulated time given by Directorate of Online Education- Sikkim Manipal University.
- Centre for Distance and Online Education (CDOE), Sikkim Manipal University has the



right to make necessary changes from time to time as deemed fit in Eligibility criteria, programme/course structure, curriculum, duration, fee structure and programme announcement dates. All changes will be notified on the website.

- Candidates should carefully read all instructions given in the Programme prospectus before start of application form.

6.1.2. Fee Structure

The fees for the MCA for Domestic student

- Total course fee (Four semesters): INR 98,000
- Each semester: INR 24,500 per semester

The fees for the MCA for International students

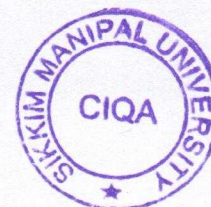
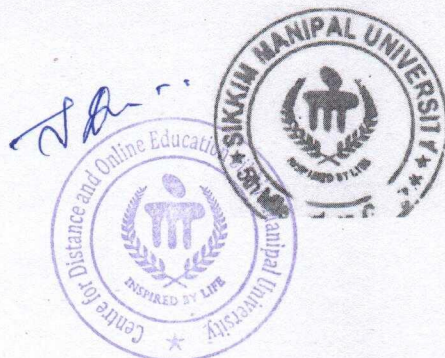
- Total course fee (Four semesters): USD 1220
- Each semester: USD 305

6.2. Curriculum Transactions

6.2.1. Programme Delivery

Centre for Distance and Online Education, Sikkim Manipal University has state-of-the-art mechanism for online mode of Academic delivery to ensure quality education. Faculty members at SMU offer expert guidance and support for holistic development of the students. Faculty members are not mere facilitators of knowledge but they also mentor students to make learning more engaging and maintain high retention level. The programme will be delivered with an aim to provide expertise and ensure that students excel in their domains. The features of programme delivery are:

- Online Mode of Academic Delivery
- Periodic review of Curriculum and Study material
- Live Interactive lectures from faculty / Course coordinators
- Continuous Academic and Technical support
- Guidance from Course coordinators
- Learning and delivery support from Course Mentors



6.2.2. Norms for Delivery of Courses in Online Mode

Credit value of the course	No. of Weeks	No. of Interactive Sessions		Hours of Study Material		Self-Study hours including Assessment etc.	Total Hours of Study (based on 30 hours per credit)
		Synchronous Online Counselling/ Webinars/ Interactive Live Lectures (1 hour per week)	Discussion Forum/ asynchronous Mentoring (2 hours per week)	e-Tutorial in hours	e-Content hours		
2 Credits	6 Weeks	6 hours	12 hours	10	10	22	60
4 Credits	12 Weeks	12 hours	24 hours	20	20	44	120

6.2.3. Learning Management System to support Online mode of Course delivery:

LMS Platform has been built to help learners reach their potential in their chosen programme. It is a secure, reliable learning experience tool that works consistently on Web and Mobile devices. Its simple interface makes it easy for instructors to design courses, create content and grade assignments. It provides a great mobile experience due to the responsive design which is paired with purpose-built native apps. It provides seamless accessibility to ensure all tools are standards-compliant and easy for students to navigate using assistive technologies. It provides 24 X 7 learning experience to facilitate learning as per the pace chosen by learners. Digital portfolio functionality allows students to document and share their learning journey as it happens, on both web and mobile platforms.

6.2.4. Course Design

The Course content is designed as per the SWAYAM guidelines using a 4-quadrant approach as detailed below to facilitate seamless delivery and learning experience:

- Quadrant-I i.e. e-Tutorial, that contains – Faculty led Video and Audio Contents, Simulations, video demonstrations, Virtual Labs
- Quadrant-II i.e. e-Content that contains - Portable Document Format or e-Books or Illustration, video demonstrations, documents as required.
- Quadrant-III i.e. Discussion forums to raise and clarify doubts on real time basis by the Course Coordinator and his team.



d) Quadrant-IV i.e. Self-Assessment, that contains MCQs, Problems, Quizzes, Assignments with solutions and Discussion forum topics.

6.2.5. Academic Calendar

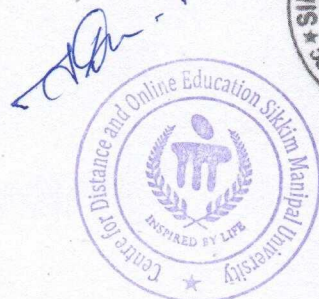
Sl. No.	Event	Academic Calendar
		Timeline
1	Commencement of Academic Session	January/July
2	Assignment Submission	5 th week – 18 th week
3	Assignment Evaluation	6 th week onwards
4	Webinars / Interactive Live Lectures and Discussion Forum for query resolution (Detailed course-wise webinar schedule will be published in student portal)	Webinar/ Interactive Live Lectures: 9 th week onwards upto 22 nd Week Discussion Forum: 9 th week onwards upto 22 nd Week
5	Online Examination Booking System (OEBS) opens for booking exam schedule	Three weeks before the commencement of examination
6	Practical Examinations (Applicable for Programmes with Practical Courses)	28 th Week
7	Admit Card Generation	One week before the commencement of examinations
8	Term End Examinations	23 rd week to 27 th Week
9	Result Declaration of End Term Examination	Within eight weeks of last date of examinations

6.3-Evaluation

The students' learning in a course would be evaluated based on Internal assignments, students' response sheets, and semester end examinations. University adopts rigorous process in development of question papers, question banks, assignments and their moderation, conduct of examinations, evaluation of answer scripts by qualified teachers, and result declaration. The Directorate shall frame the question papers so as to ensure that no part of the syllabus is left out of study by a learner.

The evaluation shall include two types of assessments - continuous or formative assessment in the form of assignments, and summative assessment in the form of end semester examination or term end examination which will be held with technology supported remote proctored examination tool.

However, we shall be considering the guidelines issued by the Regulatory bodies from time to time about the conduct of examinations.



The examinations shall be conducted to assess the knowledge acquired during the study. There shall be two systems of examinations viz., internal and external examinations. In the case of theory courses, the internal evaluation shall be conducted as Continuous Internal Assessment via Student assignments preparation, quizzes. The internal assessment shall comprise of a maximum of 30 marks for each course (One Assignment for a two-credit paper and two assignments for a four-credit paper). The end semester examination shall be three hours duration for each course at the end of each semester.

6.3.1 Assignment Submission

The dates for the submission of the Assignments will be as per the University Academic Calendar and will be put up on web portal.

- The assignment questions for all courses will be uploaded course wise.
 - All assignments will be assessed by the University.
 - The assignments are designed to encourage the students to develop the habit of continuous study of the course/subject throughout the semester.
 - Softcopies of Completed assignments should be uploaded on student portal or answers of assignment questions can be typed in the given space of student portal.
 - Specimen format of the assignment & user manual for the same will be available in student login of student Portal.
 - Ensure that you answer all questions according to the marks allocated (not more than 400 words for a 10-mark question and not more than 200 words for a five-mark question). The total page limit shall not exceed 12 pages of A-4 size. Students are advised to check their Assignments thoroughly before uploading the same, as Only ONE submission will be considered and evaluated. Multiple submissions are not allowed.
 - Content that has been directly copied from any books or the Internet will NOT be accepted.
- Assignments that have been copied and shared among students will be automatically rejected and liable for disqualification.

6.3.2 Term End Examination (TEE)

The TEE will be conducted as an online computer-based examination and remotely proctored. Examination dates along with the detailed instructions will be published well in advance on the student portal.



6.3.3 Question paper pattern

The Term end examination Question Paper for Theory subject will have both Multiple-Choice Questions (MCQs) and Descriptive Questions (DQs). The question and time distribution pattern are as follows.

Time: 3 Hours

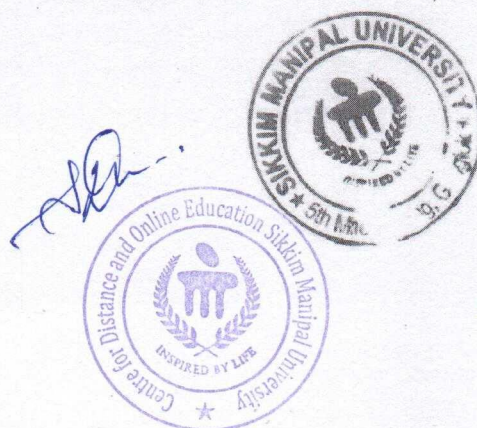
Max. Marks: 70

4 credits course/subject:

Section A (Answer ALL)	Questions x Marks	Marks	Time
Multiple Choice Questions	10 x 2	20	½ hr.
Section B – Answer Four out of Six questions			
Descriptive Questions (Short Answers)	4 x 5	20	1 hr.
Section C – Answer Three out of Four questions			
Descriptive Questions (Long Answers)	3 x 10	30	1½ hr.
Total		70	

Note:

1. The general time distribution for each type of question is as given in the above table.
2. The mark distribution is based upon the complexity involved in the question.
3. The University reserves the right to change the examination pattern from time to time. The change would be intimated to students well in advance.
4. **Reappearance:** There is no provision for improvement by reappearance in the Term-End Examinations or in the Internal Assessment once a student is declared to have passed in a paper.



6.4 Practical

The Student will have to complete Practical subjects in two parts. In the first part, students will have to attend Guided sessions for Guided exercises whereas in the second part students will have to perform the Unguided exercises assigned by the Controller of Examination.

Guided exercises will be of 70 marks and Unguided exercises will be of 30 marks.

The end semester examination (Unguided) of the practical courses will be conducted by the Controller of Examination (CoE) for three hours for 30 marks, which forms the Unguided Exercise(s) (UGE), where one or more problem(s) will be assigned to you.

To complete the lab course successfully, you have to score a combined average of 40% (i.e., totally 40 marks) in both Guided and Unguided parts together and a minimum of 40% in each part (i.e., 28 marks in Guided part and 12 marks in Unguided part).

For details refer to the respective practical manuals.

Scheme of Examination – Practical course/subject

Credits	Duration of University Exam in hour(s)	Internal Assessment Marks	University Exam Marks	Total Marks
2	3	70	30	100

6.5 Project:

The project work provides an opportunity for the students a hands-on experience in developing quality software applications.

The MCA students are encouraged to involve themselves completely on the project work in their final semester. It is advised that students develop their project for solving problems of software industry or any research organization. Doing this will give more exposure to handle real life problems of project development.

The final project report should be submitted to the University prior to the University Examination. On the day of examination students should demonstrate/present and appear for Viva-Voce examination. Project evaluation is done by assessing the project report, presentation and demonstration of the project and Viva.




Marks Distribution for Final Semester Project

Final Semester Project - Mark distribution		Marks in %
IA	Synopsis	30
TEE	Project report (Refer to project manual)	35
	Project presentation.	15
	Viva	20
Total Project Marks in %		100

6.6 Distribution of Marks in Continuous Internal Assessments

The following procedure shall be followed for awarding internal marks for theory courses. Student must submit two assignments for 4-credit (one assignment for 2-credit paper), paper each carrying 30 marks and average of both will be considered as internal assessment marks.

6.6.1 Minimum Passing Mark

The students are considered as passed in a course if they score 40% marks in the Continuous Evaluation (IA) and Term-End Examinations (TEE) individually. If a student fails in any one component (failure to get 40% marks either in IA or TEE), then he/she will be required to re-appear for that component only (IA or TEE as the case may be).

6.6.2 Marks and Grades

Based on the total marks obtained for each course in Internal Assessment and Term End examinations, students will be awarded grade for that course. The following table gives the marks, grade points, letter, grades and classification to indicate the performance of the candidate.

Range of Marks	Grade Points	Letter Grade	Description
≥90 to ≤100	10	A+	Outstanding
≥80 to <90	9	A	Excellent
≥75 to <80	8	B+	Distinction
≥70 to <75	7	B	Very Good
≥60 to <70	6	C+	Good
≥50 to <60	5	C	Average
≥40 to <50	4	D+	Below Average
<40	0	F	Re-appear



ABSENT	0	AAA	ABSENT
--------	---	-----	--------

For a semester:

$$\text{Grade Point Average [GPA]} = \sum_i C_i G_i / \sum_i C_i$$

Grade

$$\text{Grade Point Average} = \frac{\text{Sum of the multiplication of grade points by the credits of the courses}}{\text{Sum of the credits of the courses in a semester}}$$

C_i = Credits earned for the course i in any semester

G_i = Grade Point obtained for course i in any semester.

n refers to the semester in which such courses were credited

For the entire programme:

$$\text{Cumulative Grade Point Average [CGPA]} = \sum_n \sum_i C_{ni} G_{ni} / \sum_n \sum_i C_n$$

$$\text{CGPA} = \frac{\text{Sum of the multiplication of grade points by the credits of the entire programme}}{\text{Sum of the credits of the courses for the entire programme}}$$

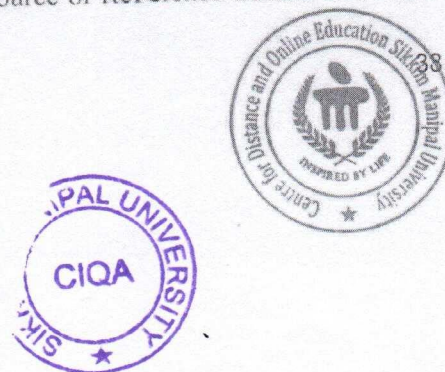
7. Requirement of the Laboratory Support and Library Resources

7.1. Laboratory Support

For practical courses (programming and coding) in syllabus, Learners will have access to lab guide for unguided exercise and online tools to carry out practice of suggested exercises. Video tutorials will be provided for better understanding of concepts and methods to practice. Lab based virtual classrooms in Learning portal will guide students about the laboratory support to the learners in order to carry out practical exercise covered in the programme. There shall be provision of a practical guide made available for learners..

7.2. Library Resources

Centre for Distance and Online Education, Sikkim Manipal University, has excellent Library facility with adequate number of copies of books in relevant titles for the MCA programme. The Central Library of Sikkim Manipal University is also having good source of Reference Books. The books



available at both the libraries are only for reference purpose and lending services. In addition, reference books as prescribed will be procured. Complete e-learning resources for the course would be made available on the Learning Management System for learning along with e-tutorial lectures. Further, expert lectures/workshops/ webinars by industry experts would also be conducted for the students.

8. Cost Estimate of the Programme and the Provisions

The cost estimate of the Programme and provisions for the fund to meet out the expenditure to be incurred in connection with MCA Programme as follows:

Sl. No.	Expenditure Heads	Approx. Amount
1	Programme Development (Single Time Investment)	49,00,000 INR
2	Programme Delivery (Per Year)	6,00,000 INR
3	Programme Maintenance (Per Year)	27,00,000 INR

9. Quality assurance mechanism and expected programme outcomes

The quality of the programme depends on scientific construction of the curriculum, strong- enough syllabus, sincere efforts leading to skillful execution of the course of the study. The ultimate achievement of MCA programme of study may reflect the gaining of knowledge and skill in finance, accounting, marketing and management area. Gaining of knowledge and skills in these fields may help the students to get new job opportunities, upgrading their position not only in employment, but also in the society.

The benchmark qualities of the programme may be reviewed based on the performance of students in their end semester examinations. Also, the feedback from the alumni, students, parents and employers will be received and analysed for further improvement of the quality of the programme.

Centre for Distance and Online Education (CDOE), Sikkim Manipal University has constituted Centre for Internal Quality Assurance (CIQA), which will assist Director, Centre for Distance and Online Education to conduct periodic review and assessments and assist the Directorate to implement necessary quality measures and effectiveness in programme delivery. CIQA is constantly involved in reviewing all materials prepared by CDOE, including syllabus, SLMs and e-learning content. CIQA will be involved in conducting studies to measure effectiveness of methods adopted for learning. As



we proceed further, CIQA will involve in benchmarking quality of academic delivery, and perform various analyses, and guide all stakeholders towards upgrading quality constantly.

Centre for Internal Quality Assurance Committee (CIQAC) chaired by the Vice Chancellor consisting of internal and external experts oversees the functioning of Centre for Internal Quality Assurance and approve the reports generated by Centre for Internal Quality Assurance on the effectiveness of quality assurance systems and processes.

In addition to CIQA, as per the guidelines of National Assessment and Accreditation Council (NAAC), Sikkim Manipal University, has constituted Internal Quality Assurance Cell (IQAC), in which academicians, industry representatives and other stakeholders are nominated as members. The IQAC is a part of the institution's system and works towards realisation of the goals of quality enhancement and sustenance, as quality enhancement is a continuous process. The prime task of the IQAC is to develop a system for conscious, consistent, and catalytic improvement in the overall performance of institutions. The work of the IQAC is the first step towards internalization and institutionalization of quality enhancement initiatives. IQAC's elementary motive is to promote measures for institutional functioning towards quality enhancement through internalization of quality culture and institutionalization of best practices.

The guidelines on quality monitoring mechanism prescribed by the UGC have been adopted by the Centre for Internal Quality Assurance for conducting institutional quality audits, to promote quality assurance and enhance as well as spread best-in-class practices of quality assurance. The university has setup an effective system for collecting feedback from the stakeholders regularly to improve its programmes. The University will conduct self- assessments regularly and use the results to improve its systems, processes etc. and finally quality of programmes.

10. Scholarship and Financial Assistance Policy

1. Divyang —Under the Divyang fee waiver, candidates are offered course fee concession of 20% for the Online Programmes offered by Sikkim Manipal University. Candidates who are certified by the Authorized Doctor may avail themselves of the Divyang fee waiver benefit.
2. Sikkim & Northeast: The fee waiver benefit of 30% on programme fees will be provided for candidates from states of Sikkim, Assam, Arunachal Pradesh, Nagaland, Meghalaya Manipur, Tripura, and Mizoram having valid address proof.
3. Defense Personnel: 20% concession on course fee to the in-service, retired and outgoing defence personnel and their family who will be enrolling for the online Programmes.



4. Manipal Group: Under this all employees* of the Manipal group and their next family members* comprising spouse and limited to 2 dependent children are offered course fee concession of 50% for all the Online Programmes offered by CDOE, Sikkim Manipal University. *Conditions apply."

5. SMU Alumni: Under this fee waiver, candidates are offered course fee concession of 20% for the Online Programmes offered by Sikkim Manipal University.

6. Full Programme Fees: Students who are willing to make an entire programme fee at the time of admission will be provided with up to 10 % fee waiver on the total fees. Not applicable for students opting for the finance options provided.

11. Learners Support:

Students can reach the University and the counsellors through any of the below given modes.

Student Support Services: Important Links and Contact Details:

- Helpdesk E-mail ID: helpdesk@onlinesmu.edu.in (To receive speedy reply Roll Number to be mentioned in all the correspondence)
- Helpdesk Number: +91-7996 799 900
- Student Portal: <https://lumen.u-next.com/smuonline/learner/login>

