

Semester II

Course Code	Course Title	Credits	Lectures /Week
USCS201	Design & Analysis of Algorithms	2	3
<p>About the Course: The course covers the concepts of - (i) calculating complexity of algorithms, (ii) the essential operations like searching, sorting, selection, pattern matching & recursion, and (iii) various algorithmic strategies like greedy, divide-n-conquer, dynamic programming, backtracking and implementations of all these on basic data structures like array, list and stack.</p>			
<p>Course Objectives: The objectives of this course are:</p> <ul style="list-style-type: none"> <input type="checkbox"/> To make students understand the basic principles of algorithm design <input type="checkbox"/> To give idea to students about the theoretical background of the basic data structures <input type="checkbox"/> To familiarize the students with fundamental problem-solving strategies like searching, sorting, selection, recursion and help them to evaluate efficiencies of various algorithms. <input type="checkbox"/> To teach students the important algorithm design paradigms and how they can be used to solve various real world problems. 			
<p>Learning Outcomes: After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <input type="checkbox"/> Students should be able to understand and evaluate efficiency of the programs that they write based on performance of the algorithms used. <input type="checkbox"/> Students should be able to appreciate the use of various data structures as per need <input type="checkbox"/> To select, decide and apply appropriate design principle by understanding the requirements of any real life problems 			
Unit	Topics	No of Lectures	
I	<p>Introduction to algorithms - What is algorithm, analysis of algorithm, Types of complexity, Running time analysis, How to Compare Algorithms, Rate of Growth, Types of Analysis, Asymptotic Notation, Big-O Notation, Omega-Ω Notation, Theta-Θ Notation, Asymptotic Analysis, Performance characteristics of algorithms, Estimating running time / number of steps of executions on paper, Idea of Computability</p> <p>Introduction to Data Structures - What is data structure, types, Introduction to Array(1-d & 2-d), Stack and List data structures, operations on these data structures, advantages disadvantages and applications of these data structures like solving linear equations, Polynomial Representation, Infix-to-Postfix conversion</p>	15	
II	<p>Recursion - What is recursion, Recursion vs Iteration, recursion applications like Factorial of a number, Fibonacci series & their</p>	15	

	<p>comparative analysis with respect to iterative version, Tower of hanoi problem</p> <p>Basic Sorting Techniques - Bubble, Selection and Insertion Sort & their comparative analysis</p> <p>Searching Techniques - Linear Search and its types, Binary Search and their comparative analysis</p> <p>Selection Techniques - Selection by Sorting, Partition-based Selection Algorithm, Finding the Kth Smallest Elements in Sorted Order & their comparative analysis</p> <p>String Algorithms - Pattern matching in strings, Brute Force Method & their comparative analysis</p>	
III	<p>Algorithm Design Techniques - Introduction to various types of classifications/design criteria and design techniques</p> <p>Greedy Technique - Concept, Advantages & Disadvantages, Applications, Implementation using problems like - file merging problem</p> <p>Divide-n-Conquer - Concept, Advantages & Disadvantages, Applications, Implementation using problems like - merge sort, Strassen's Matrix Multiplication</p> <p>Dynamic Programming - Concept, Advantages & Disadvantages, Applications, Implementation using problems like - Fibonacci series, Factorial of a number, Longest Common subsequence</p> <p>Backtracking Programming - Concept, Advantages & Disadvantages, Applications, Implementation using problems like N-Queen Problem</p>	15
<p>Textbooks:</p> <ol style="list-style-type: none"> 1. “Data Structure and Algorithm Using Python”, Rance D. Necaie, Wiley India Edition, 2016. 2. “Data Structures and Algorithms Made Easy”, Narasimha Karumanchi, CareerMonk Publications, 2016. 3. “Introduction to Algorithms”, Thomas H. Cormen, 3rd Edition, PHI. <p>Additional References:</p> <ol style="list-style-type: none"> 1. “Introduction to the Design and Analysis of Algorithms”, Anany Levitin, Pearson, 3rd Edition, 2011. 2. “Design and Analysis of Algorithms”, S. Sridhar, Oxford University Press, 2014. 		

Course Code	Course Title	Credits	Lectures /Week
USCSP201	Design & Analysis of Algorithms – Practical	1	3
1	Programs on 1-d arrays like - sum of elements of array, searching an element in array, finding minimum and maximum element in array, count the number of even and odd numbers in array. For all such programs, also find the time complexity, compare if there are multiple methods		
2	Programs on 2-d arrays like row-sum, column-sum, sum of diagonal elements, addition of two matrices , multiplication of two matrices. For all such programs, also find the time complexity, compare if there are multiple methods		
3	Program to create a list-based stack and perform various stack operations.		
4	Program to perform linear search and binary search on list of elements. Compare the algorithms by calculating time required in milliseconds using readymade libraries.		
5	Programs to sort elements of list by using various algorithms like bubble, selection sort, and insertion sort. Compare the efficiency of algorithms.		
6	Programs to select the N th Max/Min element in a list by using various algorithms. Compare the efficiency of algorithms.		
7	Programs to find a pattern in a given string - general way and brute force technique. Compare the efficiency of algorithms.		
8	Programs on recursion like factorial, fibonacci, tower of hanoi. Compare algorithms to find factorial/fibonacci using iterative and recursive approaches.		
9	Program to implement file merging, coin change problems using Greedy Algorithm and to understand time complexity.		
10	Program to implement merge sort, Straseen’s Matrix Multiplication using D-n-C Algorithm and to understand time complexity.		
11	Program to implement fibonacci series, Longest Common Subsequence using dynamic programming and to understand time complexity. Compare it with the general recursive algorithm.		
12	Program to implement N-Queen Problem, Binary String generation using Backtracking Strategy and to understand time complexity.		

Course Code	Course Title	Credits	Lectures /Week
USCS202	Advanced Python Programming	2	3
<p>About the Course: This course aims to explore and enable learners to master the skills of advanced topics in Python Programming. It helps learners develop advanced skills such as working with databases, matching patterns, implementing threads and exception handling and GUI in Python. It also highlights and why Python is a useful scripting language for all developers.</p>			
<p>Course Objectives:</p> <ul style="list-style-type: none"> <input type="checkbox"/> To learn how to design object-oriented programs with Python classes. <input type="checkbox"/> To learn about reading, writing and implementing other operation on files in Python. <input type="checkbox"/> To implement threading concept and multithreading on Python <input type="checkbox"/> To design GUI Programs and implement database interaction using Python. <input type="checkbox"/> To know about use of regular expression and handling exceptions for writing robust python programs. 			
<p>Learning Outcomes: After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <input type="checkbox"/> Ability to implement OOP concepts in Python including Inheritance and Polymorphism <input type="checkbox"/> Ability to work with files and perform operations on it using Python. <input type="checkbox"/> Ability to implement regular expression and concept of threads for developing efficient program <input type="checkbox"/> Ability to implement exception handling in Python applications for error handling. <input type="checkbox"/> Knowledge of working with databases, designing GUI in Python and implement networking in Python 			
Unit	Topics	No of Lectures	
I	<p>Working with files: Files, opening and closing a file, working with text files containing strings, knowing whether a file exists or not, working with binary files, the „with“ statement, the seek() and tell() methods, random accessing of binary files, zipping and unzipping files, working with directories, running other programs from python program</p> <p>Regular expressions: What is a regular expression?, sequence characters in regular expressions, quantifiers in regular expressions, special characters in regular expressions, using regular expression on files, retrieving information from an html file,</p> <p>Threads in python: Difference between process and thread, types of threads, benefits of threads, creating threads, single tasking and multitasking, thread synchronization, deadlock in threads, daemon threads</p> <p>Date and time in python: Date and time now, combining date and time,</p>	15	

	formatting dates and times, finding durations using “time delta”, comparing two dates, sorting dates, stopping execution temporarily, knowing the time taken by a program, calendar module	
II	<p>Database in python: Using SQL with python, retrieving rows from a table, inserting rows into a table, deleting rows from a table, updating rows in a table, creating database tables through python, Exception handling in databases.</p> <p>Exceptions in python: Errors in a python program, compile & run-time errors, logical error, exceptions-exception handling, types of exceptions, the except block, the assert statement, user-defined exceptions, logging the exceptions</p> <p>Networking: Protocols,server-client architecture, tcp/ip and udp communication</p> <p>Graphical user interface: Creating a GUI in python, Widget classes, Working with Fonts and Colours, working with Frames, Layout manager, Event handling</p>	15
III	<p>OOPs in python: Features of Object Oriented Programming system (oops)-classes and objects, encapsulation, abstraction, inheritance, polymorphism, constructors and destructors</p> <p>Classes and objects: Creating a class, the self-variable, types of variables, namespaces, types of methods, instance methods, class methods, static methods, passing members of one class to another class, inner classes</p> <p>Inheritance and polymorphism: Inheritance in python, types of inheritance- single inheritance, multilevel inheritance, hierarchical inheritance, multiple inheritance, constructors in inheritance, overriding super class constructors and methods, the super() method, method resolution order (mro), polymorphism, duck typing, operator overloading, method overloading, method overriding,</p> <p>Abstract classes and interfaces: Abstract class, abstract method,interfaces in python, abstract classes vs. Interfaces</p>	15
<p>Textbooks:</p> <ol style="list-style-type: none"> 1. Paul Gries , Jennifer Campbell, Jason Montojo, Practical Programming: An Introduction to Computer Science Using Python 3, Pragmatic Bookshelf, 3rd Edition, 2018 2. Programming through Python, M. T Savaliya, R. K. Maurya, G M Magar, Revised Edition, Sybgen Learning India, 2020 <p>Additional References:</p> <ol style="list-style-type: none"> 1. Advanced Python Programming, Dr. Gabriele Lanaro, Quan Nguyen, SakisKasampalis, Packt Publishing, 2019 2. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018 3. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018 4. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017 5. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018 		

Course Code	Course Title	Credits	Lectures /Week
USCSP202	Advanced Python Programming – Practical	1	3
1	Write a program to Python program to implement various file operations.		
2	Write a program to Python program to demonstrate use of regular expression for suitable application.		
3	Write a Program to demonstrate concept of threading and multitasking in Python.		
4	Write a Python Program to work with databases in Python to perform operations such as <ul style="list-style-type: none"> a. Connecting to database b. Creating and dropping tables c. Inserting and updating into tables. 		
5	Write a Python Program to demonstrate different types of exception handling.		
6	Write a GUI Program in Python to design application that demonstrates <ul style="list-style-type: none"> a. Different fonts and colors b. Different Layout Managers c. Event Handling 		
7	Write Python Program to create application which uses date and time in Python.		
8	Write a Python program to create server-client and exchange basic information		
9	Write a program to Python program to implement concepts of OOP such as <ul style="list-style-type: none"> a. Types of Methods b. Inheritance c. Polymorphism 		
10	Write a program to Python program to implement concepts of OOP such as <ul style="list-style-type: none"> a. Abstract methods and classes b. Interfaces 		

Course Code	Course Title	Credits	Lectures /Week
USCS203	Introduction to OOPs using C++	2	3
<p>About the Course: The course aims to introduce a new programming paradigm called Object Oriented Programming. This will be covered using C++ programming language. C++ is a versatile programming language, which supports a variety of programming styles, including procedural, object-oriented, and functional programming. This makes C++ powerful as well as flexible. It can be used to develop software such as operating systems, databases, and compilers.</p>			
<p>Course Objectives:</p>			
<p>Learning Outcomes: After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <input type="checkbox"/> Work with numeric, character and textual data and arrays. <input type="checkbox"/> Understand the importance of OOP approach over procedural language. <input type="checkbox"/> Understand how to model classes and relationships using UML. <input type="checkbox"/> Apply the concepts of OOPS like encapsulation, inheritance and polymorphism. <input type="checkbox"/> Handle basic file operations. 			
Unit	Topics	No of Lectures	
I	<p>Introduction to Programming Concepts: Object oriented programming paradigm, basic concepts of object oriented programming, benefits of object oriented programming, object oriented languages, applications of object oriented programming.</p> <p>Tokens-keywords, identifiers, constants-integer, real, character and string constants, backslash constants, features of C++ and its basic structure, simple C++ program without class, compiling and running C++ program.</p> <p>Data Types, Data Input Output and Operators: Basic data types, variables, rules for naming variables, programming constants, the type cast operator, implicit and explicit type casting, cout and cin statements, operators, precedence of operators.</p> <p>Decision Making, Loops, Arrays and Strings: Conditional statements-if, if..else, switch loops- while, do...while, for, types of arrays and string and string manipulations</p> <p>Unified Modeling Language (UML): Introduction to UML & class diagrams.</p> <p>Classes, Abstraction & Encapsulation: Classes and objects, Dot Operator, data members, member functions, passing data to functions, scope and visibility of variables in function.</p>	15	

II	<p>Constructors and Destructors: Default constructor, parameterized constructor, copy constructor, private constructor, destructors.</p> <p>Working with objects: Accessor - mutator methods, static data and static function, access specifiers, array of objects.</p> <p>Polymorphism - Binding-static binding & overloading, constructor overloading function overloading, operator overloading, overloading unary and binary operators.</p> <p>Modelling Relationships in Class Diagrams: Association, Aggregation-Composition and examples covering these principles</p>	15
III	<p>Inheritance: Defining base class and its derived class, access specifiers, types of inheritance-single, multiple, hierarchical, multilevel, hybrid inheritance, friend function and friend class, constructors in derived classes.</p> <p>Modelling Relationships: Generalization-Specialization and examples covering these principles</p> <p>Run time Polymorphism - Dynamic Binding, Function overriding, virtual function, pure virtual function, virtual base class, abstract class.</p> <p>Pointers: Introduction to pointers, * and & operators, assigning addresses to pointer variables, accessing values using pointers, pointers to objects & this pointer, pointers to derived classes</p> <p>File Handling: File Stream classes, opening and closing file-file opening modes, text file handling, binary file handling.</p> <p>Applying OOP to solve real life applications: To cover case studies like library management, order management etc. to design classes covering all relationships</p>	15
<p>Textbooks:</p> <ol style="list-style-type: none"> Object Oriented Programming with C++, Balagurusamy E., 8th Edition, McGraw Hill Education India. UML & C++: A Practical Guide to Object Oriented Development, Lee/Tepfenhart, Pearson Education, 2nd Edition 2015 <p>Additional References:</p> <ol style="list-style-type: none"> Mastering C++ by Venugopal, Publisher: McGraw-Hill Education, 2017 Let Us C++ by Kanetkar Yashwant, Publisher: BPB Publications, 2020 Object Oriented Analysis and Design by Timothy Budd TMH, 2001 		

Course Code	Course Title	Credits	Lectures /Week
USCSP203	Introduction to OOPs using C++ - Practical	1	3
1	Program to demonstrate use of data members & member functions.		
2	Programs based on branching and looping statements using classes.		
3	Program to demonstrate one and two dimensional arrays using classes		
4	Program to use scope resolution operator. Display the various values of the same variables declared at different scope levels.		
5	Programs to demonstrate various types of constructors and destructors.		
6	Programs to demonstrate use of public, protected & private scope specifiers.		
7	Programs to demonstrate single and multilevel inheritance		
8	Programs to demonstrate multiple inheritance and hierarchical inheritance		
9	Programs to demonstrate inheritance and derived class constructors		
10	Programs to demonstrate friend function, inline function, this pointer		
11	Programs to demonstrate function overloading and overriding.		
12	Programs to demonstrate use of pointers		
13	Programs to demonstrate text and binary file handling		

Course Code	Course Title	Credits	Lectures /Week
USCS204	Database Systems	2	3
<p>About the Course: The course introduces the core principles and techniques required in the design and implementation of database systems. It includes ER Model, Normalization, Relational Model, and Relational Algebra. It also provides students with theoretical knowledge and practical skills of creating and manipulating data with an interactive query language (MySQL). It also provide student knowledge and importance of data protection.</p>			
<p>Course Objectives:</p> <ul style="list-style-type: none"> <input type="checkbox"/> To make students aware fundamentals of database system. <input type="checkbox"/> To give idea how ERD components helpful in database design and implementation. <input type="checkbox"/> To experience the students working with database using MySQL. <input type="checkbox"/> To familiarize the student with normalization, database protection and different DCL Statements. <input type="checkbox"/> To make students aware about importance of protecting data from unauthorized users. <input type="checkbox"/> To make students aware of granting and revoking rights of data manipulation. 			
<p>Learning Outcomes: After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <input type="checkbox"/> To appreciate the importance of database design. <input type="checkbox"/> Analyze database requirements and determine the entities involved in the system and their relationship to one another. <input type="checkbox"/> Write simple queries to MySQL related to String, Maths and Date Functions. <input type="checkbox"/> Create tables and insert/update/delete data, and query data in a relational DBMS using MySQL commands. <input type="checkbox"/> Understand the normalization and its role in the database design process. <input type="checkbox"/> Handle data permissions. <input type="checkbox"/> Create indexes and understands the role of Indexes in optimization search. 			
Unit	Topics	No of Lectures	
I	<p>Introduction to DBMS – Database, DBMS – Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture</p> <p>Data models - Client/Server Architecture, Object Based Logical Model, Record Based Logical Model (relational, hierarchical, network)</p> <p>Entity Relationship Model - Entities, attributes, entity sets, relations, relationship sets, Additional constraints (key constraints, participation constraints, weak entities, aggregation / generalization, Conceptual Design using ER (entities VS attributes, Entity Vs relationship, binary Vs ternary, constraints beyond ER)</p>	15	

	<p>ER to Table- Entity to Table, Relationship to tables with and without key constraints.</p> <p>DDL Statements - Creating Databases, Using Databases, datatypes, Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables</p> <p>DML Statements – Viewing the structure of a table insert, update, delete, Select all columns, specific columns, unique records, conditional select, in clause, between clause, limit, aggregate functions (count, min, max, avg, sum), group by clause, having clause</p>	
II	<p>Relational data model– Domains, attributes, Tuples and Relations, Relational Model Notation, Characteristics of Relations, Relational Constraints - primary key, referential integrity, unique constraint, Null constraint, Check constraint</p> <p>Relational Algebra operations (selection, projection, set operations union, intersection, difference, cross product, Joins –conditional, equi join and natural joins, division)</p> <p>Functions – String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round, truncate) Date Functions (adddate, datediff, day, month, year, hour, min, sec, now, reverse)</p> <p>Joining Tables – inner join, outer join (left outer, right outer, full outer)</p> <p>Subqueries – subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries</p>	15
III	<p>Schema refinement and Normal forms: Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition.</p> <p>Database Protection: Security Issues, Threats to Databases, Security Mechanisms, Role of DBA, Discretionary Access Control, Backing Up and Restoring databases</p> <p>Views (creating, altering dropping, renaming and manipulating views)</p> <p>DCL Statements (creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges), Transaction control commands – Commit, Rollback</p> <p>Index Structures of Files: Introduction, Primary index, Clustering Index, Multilevel indexes</p>	15

Textbooks:

1. “Fundamentals of Database System”, ElmasriRamez, NavatheShamkant, Pearson Education, Seventh edition, 2017
2. “Database Management Systems”, Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition, 2014
3. “Murach's MySQL”, Joel Murach, 3rd Edition, 3rd Edition, 2019

Additional References:

1. “Database System Concepts”, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw Hill, 2017
2. “MySQL: The Complete Reference”, Vikram Vaswani, McGraw Hill, 2017
3. “Learn SQL with MySQL: Retrieve and Manipulate Data Using SQL Commands with Ease”, Ashwin Pajankar, BPB Publications, 2020

Course Code	Course Title	Credits	Lectures /Week
USCSP204	Database Systems – Practical	1	3
1.	Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)		
2.	Perform the following: <ul style="list-style-type: none"> <input type="checkbox"/> Viewing all databases <input type="checkbox"/> Creating a Database <input type="checkbox"/> Viewing all Tables in a Database <input type="checkbox"/> Creating Tables (With and Without Constraints) <input type="checkbox"/> Inserting/Updating/Deleting Records in a Table 		
3.	Perform the following: <ul style="list-style-type: none"> <input type="checkbox"/> Altering a Table <input type="checkbox"/> Dropping/Truncating/Renaming Tables <input type="checkbox"/> Backing up / Restoring a Database 		
4.	Perform the following: <ul style="list-style-type: none"> <input type="checkbox"/> Simple Queries <input type="checkbox"/> Simple Queries with Aggregate functions 		
5.	Queries involving <ul style="list-style-type: none"> <input type="checkbox"/> Date Functions <input type="checkbox"/> String Functions <input type="checkbox"/> Math Functions 		
6.	Join Queries <ul style="list-style-type: none"> <input type="checkbox"/> Inner Join <input type="checkbox"/> Outer Join 		

7.	<p>Subqueries</p> <ul style="list-style-type: none"> <input type="checkbox"/> With IN clause <input type="checkbox"/> With EXISTS clause
8.	<p>Converting ER Model to Relational Model and apply Normalization on database. (Represent entities and relationships in Tabular form, Represent attributes as columns, identifying keys and normalization up to 3rd Normal Form).</p>
9.	<p>Views</p> <ul style="list-style-type: none"> <input type="checkbox"/> Creating Views (with and without check option) <input type="checkbox"/> Dropping views <input type="checkbox"/> Selecting from a view
10.	<p>DCL statements</p> <ul style="list-style-type: none"> <input type="checkbox"/> Granting and revoking permissions <input type="checkbox"/> Saving (Commit) and Undoing (rollback)
11.	<p>Creating Indexes on data tables.</p>

Course Code	Course Title	Credits	Lectures /Week
USCS205	Calculus	2	3
<p>About the Course: Calculus is a branch of mathematics that involves the study of rates of change. In Computer Science, Calculus is used in Machine Learning, Data Mining, Scientific Computing, Image Processing, and creating the graphics and physics engines for video games, including the 3D visuals for simulations.</p>			
<p>Course Objectives:</p> <ul style="list-style-type: none"> <input type="checkbox"/> The primary objective of this course is to introduce the basic tools of Calculus which are helpful in understanding their applications to the real world problems. <input type="checkbox"/> The course is designed to have a grasp of important concepts of Calculus in a scientific way. <input type="checkbox"/> It covers topics from as basic as definition of functions to partial derivatives of functions in a gradual and logical way. <input type="checkbox"/> The learner is expected to solve as many examples as possible to get complete clarity and understanding of the topics covered. 			
<p>Learning Outcomes: After successful completion of this course, learners would be able to:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Develop mathematical skills and enhance thinking power of learners. <input type="checkbox"/> Understand mathematical concepts like limit, continuity, derivative, integration of functions, partial derivatives. <input type="checkbox"/> Appreciate real world applications which use the learned concepts. <input type="checkbox"/> Skill to formulate a problem through Mathematical modelling and simulation. 			
Unit	Topics	No of Lectures	
I	<p>DERIVATIVES AND ITS APPLICATIONS: Review of Basic Concepts: Functions, limit of a function, continuity of a function, derivative function.</p> <p>Derivative In Graphing And Applications: Increase, Decrease, Concavity, Relative Extreme; Graphing Polynomials, Rational Functions, Cusps and Vertical Tangents. Absolute Maxima and Minima, Applied Maximum and Minimum Problems, Newton's Method.</p>	15	
II	<p>INTEGRATION AND ITS APPLICATIONS: Integration: An Overview of the Area Problem, Indefinite Integral, Definition of Area as a Limit; Sigma Notation, Definite Integral, Evaluating Definite Integrals by Substitution, Numerical Integration: Simpson's Rule.</p> <p>Applications of Integration: Area between two curves, Length of a plane curve.</p> <p>Mathematical Modeling with Differential Equations: Modeling with</p>	15	

	Differential Equations, Separation of Variables, Slope Fields, Euler's Method, First-Order Differential Equations and Applications.	
III	<p>PARTIAL DERIVATIVES AND ITS APPLICATIONS:</p> <p>Functions of Several Variables: Functions of two or more variables, Limits and Continuity of functions of two or three variables.</p> <p>Partial Derivatives: Partial Derivatives, Differentiability, Differentials, and Local Linearity, Chain Rule, Implicit Differentiation, Directional Derivatives and Gradients,</p> <p>Applications of Partial Derivatives: Tangent Planes and Normal Vectors, Maxima and Minima of Functions of Two Variables.</p>	15
<p>Textbooks:</p> <p>1. Calculus: Early transcendental (10th Edition): Howard Anton, Irl Bivens, Stephen Davis, John Wiley & sons, 2012.</p> <p>Additional References:</p> <p>1. Calculus and analytic geometry (9th edition): George B Thomas, Ross L Finney, Addison Wesley, 1995</p> <p>2. Calculus: Early Transcendentals (8th Edition): James Stewart, Brooks Cole, 2015.</p> <p>3. Calculus (10th Edition): Ron Larson, Bruce H. Edwards, Cengage Learning, 2013.</p> <p>4. Thomas' Calculus (13th Edition): George B. Thomas, Maurice D. Weir, Joel R. Hass, Pearson, 2014.</p>		

Course Code	Course Title	Credits	Lectures /Week
USCSP205	Calculus – Practical	1	3
1	<p>Review of Basic Concepts –</p> <p>a. Functions of one variable, its domain and range, Operations on functions</p> <p>b. Limits of functions of one variable</p> <p>c. Continuity of functions of one variable</p> <p>d. Derivatives of functions of one variable</p>		
2	<p>Applications of Derivatives I –</p> <p>a. Increasing and Decreasing functions</p> <p>b. Concavity and inflection points</p> <p>c. Relative Extrema</p> <p>d. Absolute Extrema</p>		
3	<p>Applications of Derivatives II –</p> <p>a. Analysis of polynomials</p> <p>b. Graphing rational functions</p> <p>c. Graphs With Vertical Tangents And Cusps</p> <p>d. Newton's method to find approximate solution of an equation</p>		

4	Integration – a. Finding area using rectangle method and antiderivative method b. Indefinite and definite integrals c. Properties of integrals d. Numerical integration using Simpson’s rule.
5	Applications of Integration – a. Area between two curves b. Length of a plane curve
6	Differential Equations – a. Solution of a first order first degree differential equation using variable separable method b. Solution of a first order linear differential equation using integrating factor c. Numerical solution of first-order equations using Euler’s method d. Modeling using differential equation
7	Functions of Several Variables – a. Functions of two or more variables, its domain and range, Operations on functions, level curves b. Limits of functions of two or three variables c. Continuity of functions of two or three variables
8	Partial Derivatives I – a. Partial derivatives of functions, First and Second order partial derivatives, Mixed derivative theorem, Higher order partial derivatives b. Differential for functions of two or three variables c. Local linear approximation for functions of two or three variables
9	Partial Derivatives II – a. Chain rule for functions of two or three variables b. Implicit differentiation c. Directional derivatives and gradient
10	Applications of Partial Derivatives– a. Tangent Planes and Normal Vectors for functions of two or three variables b. Maxima and Minima of Functions of Two Variables
NOTE	Above Practical’s can also to be implemented using Sage Math/ Geogebra.

Course Code	Course Title	Credits	Lectures /Week
USCS206	Statistical Methods	2	3
About the Course:			
This course introduces the key concepts in probability, conditional probabilities and distribution theory, including probability laws, random variables, expectation and variance, functions of random variables and its probability distributions. Emphasis is placed on theoretical understanding combined with problem solving using various statistical inferential techniques.			
Course Objectives:			
<input type="checkbox"/> To make learner aware about basic probability axioms and rules and its application. <input type="checkbox"/> To understand the concept of conditional probability and Independence of events. <input type="checkbox"/> To make learner familiar with discrete and continuous random variables as well as standard discrete and continuous distributions. <input type="checkbox"/> To learn computational skills to implement various statistical inferential approaches.			
Learning Outcomes:			
After successful completion of this course, learners would be able to			
<input type="checkbox"/> Calculate probability, conditional probability and independence. <input type="checkbox"/> Apply the given discrete and continuous distributions whenever necessary. <input type="checkbox"/> Define null hypothesis, alternative hypothesis, level of significance, test statistic and p value. <input type="checkbox"/> Perform Test of Hypothesis as well as calculate confidence interval for a population parameter for single sample and two sample cases. <input type="checkbox"/> Apply non-parametric test whenever necessary. <input type="checkbox"/> Conduct and interpret one-way and two-way ANOVA.			
Unit	Topics	No of Lectures	
I	Probability: Random experiment, sample space, events types and operations of events, Probability definition: classical, axiomatic, Elementary Theorems of probability (without proof). Conditional probability, „Bayes" theorem, independence, Examples on Probability. Random Variables: Concept and definition of a discrete random variable and continuous random variable. Probability mass function, Probability density function and cumulative distribution function of discrete and continuous random variable, Properties of cumulative distribution function.	15	
II	Mathematical Expectation and Variance: Expectation of a function, Variance and S.D of a random variable, properties. Standard Probability distributions: Introduction, properties, examples and applications of each of the following distributions: Binomial distribution, Normal distribution, Chi-square distribution, t distribution, F distribution	15	

III	<p>Hypothesis testing: One sided, Two sided hypothesis, critical region, p-value, tests based on t, Normal and F, confidence intervals.</p> <p>Analysis of Variance: One-way, two-way analysis of variance.</p> <p>Non-parametric tests: Need of non-parametric tests, Sign test, Wilcoxon's signed rank test, run test, Kruskal-Walis tests, Chi square test.</p>	15
<p>Textbooks:</p> <ol style="list-style-type: none"> Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi Goon, A. M., Gupta, M. K. and Dasgupta, B. (1983). Fundamentals of Statistics, Vol. 1, Sixth Revised Edition, The World Press Pvt. Ltd., Calcutta. <p>Additional References:</p> <ol style="list-style-type: none"> Mood, A. M. and Graybill, F. A. and Boes D.C. (1974). Introduction to the Theory of Statistics, Ed. 3, McGraw Hill Book Company. Hoel P. G. (1971). Introduction to Mathematical Statistics, John Wiley and Sons, New York. Hogg, R.V. and Craig R.G. (1989). Introduction to Mathematical Statistics, Ed. MacMillan Publishing Co., New York. Walpole R. E., Myers R. H. and Myers S. L. (1985), Probability and Statistics for Engineers and Scientists Agarwal, B. L. (2003). Programmed Statistics, Second Edition, New Age International Publishers, New Delhi. 		

Course Code	Course Title	Credits	Lectures /Week
USCSP206	Statistical Methods – Practical	1	3
1	<p>Probability-</p> <ol style="list-style-type: none"> Examples based on Probability definition: classical, axiomatic Examples based on elementary Theorems of probability 		
2	<p>Conditional probability and independence-</p> <ol style="list-style-type: none"> Examples based on Conditional probability Examples based on „Bayes” theorem Examples based on independence 		
3	<p>Discrete random variable-</p> <ol style="list-style-type: none"> Probability distribution of discrete random variable Probability mass function 		
4	<p>Continuous random variable-</p> <ol style="list-style-type: none"> Probability distribution of continuous random variable Probability density function 		

5	Mathematical Expectation and Variance- <ol style="list-style-type: none"> Mean of discrete and continuous Probability distribution S.D. and variance of discrete and continuous Probability distribution
6	Standard probability distributions- <ol style="list-style-type: none"> Calculation of probability, mean and variance based on Binomial distribution Calculation of probability based on Normal distribution
7	Large Sample tests based on Normal (Z) - <ol style="list-style-type: none"> Test of significance for proportion (Single proportion $H_0: P = P_0$) Test of significance for difference between two proportions (Double proportion $H_0: P_1 = P_2$) Test of significance for mean (Single mean $H_0: \mu = \mu_0$) Test of significance for difference between two means. (Double mean $H_0: \mu_1 = \mu_2$)
8	Small sample tests based on t and F- <ol style="list-style-type: none"> t-test for significance of single mean, population variance being unknown (Single mean $H_0: \mu = \mu_0$) t-test for significance of the difference between two sample means (Independent samples) t-test for significance of the difference between two sample means (Related samples) F-Test to Compare Two Variances
9	Analysis of variance - <ol style="list-style-type: none"> Perform One-way ANOVA Perform Two-way ANOVA
10	Non-parametric tests- <ol style="list-style-type: none"> Sign test and Wilcoxon Sign rank test Run test Kruskal-Wallis (H) test Chi-square test
Note: Practical no. 6, 7, 8, 9 can also to be implemented using R programming.	

Course Code	Course Title	Credits	Lectures /Week
USCS207	E-Commerce & Digital Marketing	2	3
About the Course:			
This course introduces the fundamental concepts of e-commerce, its types, the various legal and ethical issues of e-commerce and different e-commerce applications. The course also aims to introduce basic principles and types of digital marketing and web and Google analytics			
Course Objectives:			
<ul style="list-style-type: none"> <input type="checkbox"/> To understand increasing significance of E-Commerce and its applications in Business and Various Sectors <input type="checkbox"/> To provide an insight on Digital Marketing activities on various Social Media platforms and its emerging significance in Business <input type="checkbox"/> To understand Latest Trends and Practices in E-Commerce and Digital Marketing, along with its Challenges and Opportunities for an Organization 			
Learning Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <input type="checkbox"/> Understand the core concepts of E-Commerce. <input type="checkbox"/> Understand the various online payment techniques <input type="checkbox"/> Understand the core concepts of digital marketing and the role of digital marketing in business. <input type="checkbox"/> Apply digital marketing strategies to increase sales and growth of business <input type="checkbox"/> Apply digital marketing through different channels and platforms <input type="checkbox"/> Understand the significance of Web Analytics and Google Analytics and apply the same. 			
Unit	Topics	No of Lectures	
I	<p>Introduction to E-Commerce and E- Business: Definition and competing in the digital economy, Impact of E-Commerce on Business Models, Factors Driving e-commerce and e-Business Models, Economics and social impact of e-Business, opportunities and Challenges, e-Commerce vs m- Commerce, Different e-Commerce Models (B2B, B2C, C2B, C2C, B2E), e-Commerce Applications: e-Trading, e-Learning, e-Shopping, Virtual Reality & Consumer Experience, Legal and Ethical issues in e-Commerce.</p> <p>Overview of Electronic Payment systems: Types of Electronic payment schemes (Credit cards, Debit cards, Smartcards, Internet banking), E-checks, E-Cash Concepts and applications of EDI and Limitation</p> <p>Introduction & origin of Digital Marketing: Traditional v/s Digital Marketing. Digital Marketing Strategy, The P-O-E-M Framework, Segmenting & Customizing Messages, The Digital landscape, Digital Advertising Market in India. Skills required in Digital Marketing. Digital Marketing Plan.</p>	15	
II	<p>Social Media Marketing: Meaning, Purpose, types of social media websites, Social Media Engagement, Target audience, Facebook Marketing: Business through Facebook Marketing, Creating Advertising Campaigns,</p>	15	

	<p>Adverts, Facebook Marketing Tools, LinkedIn Marketing: Importance of LinkedIn Marketing, Framing LinkedIn Strategy, Lead Generation through LinkedIn, Content Strategy, Analytics and Targeting, Twitter Marketing: Framing content strategy, Twitter Advertising Campaigns, YouTube Marketing: Video optimization, Promoting on YouTube, Monetization, YouTube Analytics</p> <p>Email Marketing: Types of Emails, Mailing List, Email Marketing tools, Email Deliverability & Email Marketing automation</p> <p>Mobile Marketing: Introduction, Mobile Usage, Mobile Advertising, Mobile Marketing Types, Mobile Marketing Features, Mobile Campaign Development, Mobile Advertising Analytics</p> <p>Content Marketing: Introduction, Content marketing statistics, Types of Content, Types of Blog posts, Content Creation, Content optimization, Content Management & Distribution, Content Marketing Strategy, Content creation tools and apps, Challenges of Content Marketing.</p>	
III	<p>Search Engine Optimization: Meaning, Common SEO techniques, Understanding Search Engines, basics of Keyword search, Google rankings, Link Building, Steps to optimize website, On-page and off-page optimization</p> <p>Search Engine Marketing: Introduction to SEM, Introduction to Ad Words - Google Ad Words, Ad Words fundamentals, Ad Placement, Ad Ranks, Creating Ad Campaigns, Campaign Report Generation, Display marketing, Buying Models: Cost per Click (CPC), Cost per Milli (CPM), Cost per Lead (CPL), Cost per Acquisition (CPA).</p> <p>Web Analytics: Purpose, History, Goals & objectives, Web Analytic tools & Methods. Web Analytics Mistakes and Pitfalls.</p> <p>Google Analytics: Basics of Google Analytics, Installing Google Analytics in website, Parameters of Google Analytics, Reporting and Analysis</p>	15
<p>Textbooks:</p> <ol style="list-style-type: none"> 1. “E-Commerce Strategy, Technologies and Applications”, Whitley, David, Tata McGraw Hill, 2017 2. Digital Marketing, Seema Gupta, McGraw Hill Education, 2nd Edition <p>Additional References:</p> <ol style="list-style-type: none"> 1. E-Commerce by S. Pankaj, A.P.H. Publication, New Delhi 2. Fundamentals of Digital Marketing, Punit Singh Bhatia, Pearson, 2nd Edition 3. “Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation”, Damian Ryan, Calvin Jone. Kogan Page, 4th Edition 		

Evaluation Scheme

I. Internal Evaluation for Theory Courses – 25 Marks

(i) Mid-Term Class Test– 15Marks

- It should be conducted using any **learning management system** such as **Moodle** (Modular object-oriented dynamic learning environment)
- The test should have **15 MCQ's** which should be solved in a time duration of **30 minutes**.

(ii) Assignment/ Case study/ Presentations– 10 Marks

- Assignment / Case Study Report / Presentation can be uploaded on any **learning management system**.

II. External Examination for Theory Courses – 75 Marks

- Duration: **2.5 Hours**
- Theory question paper pattern:

All questions are compulsory.			
Question	Based on	Options	Marks
Q.1	Unit I	<i>Any 4 out of 6</i>	20
Q.2	Unit II	<i>Any 4 out of 6</i>	20
Q.3	Unit III	<i>Any 4 out of 6</i>	20
Q.4	Unit I,II and III	<i>Any 5 out of 6</i>	15

- All questions shall be compulsory with internal choice within the questions.
- Each Question maybe sub-divided into subquestions as a, b, c, d, etc. & the allocation of Marks depends on the weightage of the topic.

III. Practical Examination

- Each core subject carries 50 Marks
40 marks + 05 marks (journal) + 05 marks (viva)
- Duration: **2 Hours** for each practical course.
- Minimum **80% practical** from each core subjects are required to be completed.
- Certified Journal is compulsory for appearing at the time of Practical Exam**
- The final submission and evaluation of **journal in electronic form** using a Learning Management System / Platform can be promoted by college.
