



# Janardan Bhagat Shikshan Prasarak Sanstha's CHANGU KANA THAKUR ARTS, COMMERCE & SCIENCE COLLEGE, NEW PANVEL (AUTONOMOUS)

Re-accredited 'A+' Grade by NAAC 'College with Potential for Excellence' Status Awarded by UGC 'Best College Award' by University of Mumbai

> Program: B.Sc Revised Syllabus of F.Y.B.Sc. Computer Science Choice Based Credit & Grading System (60:40) w.e.f. Academic Year 2022-23

## Preamble

The rise of Information and Communication Technology (ICT) has profoundly affected modern society. Increasing applications of computers in almost all areas of human endeavor has led to vibrant industries with concurrent rapid change in technology. As the computing field advances at a rapid pace, the students must possess a solid foundation that allows and encourages them to maintain relevant skills as the field evolves. Specific languages and technology platforms change over time. Thus, students must continue to learn and adapt their skills throughout their careers. To develop this ability, students will be exposed to multiple programming languages, tools, paradigms and technologies as well as the fundamental underlying principles throughout this programme. The programme offers required courses such as programming languages, data structures, computer architecture and organization, algorithms, database systems, operating systems, and software engineering; as well as specialized courses in artificial intelligence, computer-based communication networks, distributed computing, information security, graphics, human-computer interaction, multimedia, scientific computing, web technology, and other current topics in computer science. The core philosophy of this programme is to –

- Form strong foundations of Computer Science
- Nurture programming, analytical & design skills for real-world problems.
- Introduce emerging trends to the students in a gradual way.
- Groom the students for the challenges of ICT industry

The students these days not only aspire for a career in the industry but also look for research opportunities. The main aim of this programme is to deliver a modern curriculum that will equip graduates with strong theoretical and practical backgrounds to enable them to excel in the workplace and to be lifelong learners. Not only does it prepare the students for a career in the Software industry, it also motivates them towards further studies and research opportunities. Graduating students, can thus take up postgraduate programmes in CS leading to research as well as R&D, can be employable at IT industries, or can adopt a business management career in the first year i.e., for semester I & II, basic foundation of important skills required for software development is laid.

The syllabus proposes to have four core subjects of Computer science and two core courses of Mathematics-Statistics. All core subjects are proposed to have theory as well as practical tracks. While the Computer Science courses will form fundamental skills for solving computational problems, the Mathematics & Statistics course will inculcate research-oriented acumen. Ability Enhancement Courses on Soft Skill Development will ensure an overall and holistic development of the students. The syllabus design for further semesters encompasses more advanced and specialized courses of Computer Science. We sincerely believe that any student taking this programme will get a very strong foundation and exposure to basics, advanced and emerging trends of the subject. We hope that the students" community and teachers" fraternity will appreciate the treatment given to the courses in the syllabus.

We wholeheartedly thank all experts who shared their valuable feedback and suggestions in order to improvise the contents; we have sincerely attempted to incorporate each of them. We further thank the members of the Board of Studies for their confidence in us.

# **Scheme of Examination**

#### A) Internal Assessment: 40 %

Sr. No.	Particular	Marks
01	One periodical class test / online examination to be conducted in the given semester	20 Marks
02	Continuous Assessment	20 Marks

Question Paper Pattern for Continuous Assessment (Total Marks 20 to be converted in 10 marks)

Marks	Group Project*/ Individual Project	Presentation and write- up	Practical Skills	Open book test	Quiz
5	Hypothesis/Topic of the project	Presentation skill	Demonstration of skill	High order thinking questions	Quiz on application of subject in
5	Actual laboratory work/Field work	Knowledge	Viva	(HOTS)	real life
5	Result/output	Quality of ppt	Report		
5	Dissertation/Report	Writing skill	Problem solving ability		

#### B) Semester End Examination: 60 %

60 Marks

• Duration: The examination shall be of 2 hours duration.

#### Theory question paper pattern

- 1. There shall be four questions each of 15 marks.
- 2. All questions shall be compulsory with internal options.
- 3. Question may be subdivided into sub-questions a, b, c... and the allocation of marks depends on the weightage of the unit.

#### > Passing Standard

The learners to pass a course shall have to obtain a minimum of 40% marks in aggregate for each course where the course consists of Internal Assessment and Semester End Examination. The learners shall obtain minimum of 40% marks (i.e. 16 out of 40) in the Internal Assessment and 40% marks in Semester End Examination (i.e. 24 Out of 60) separately, to pass the course and minimum of Grade D, wherever applicable, to pass a particular semester. A learner will be said to have passed the course if the learner passes the Internal Assessment and Semester End Examination together.

#### 40 Marks

I. **Practical Examination :** - 300 (50 marks x 6 core papers)

II.	Each	core	subject	carries	:-	50	Marks
-----	------	------	---------	---------	----	----	-------

Sr. No.	Particulars of External	Marks
1	Laboratory Work	40
2	Journal	05
3	Viva	05
	TOTAL	50

Minimum 75 % practical from each core subjects are required to be completed and written in the journal. (Certified Journal is compulsory for appearing at the time of Practical Exam) ------

Semester - I	[Under CBCS Scheme]
--------------	---------------------

			1			1	
Course	Course Type	Course code	Hrs./ week	Internal assessment	Semester - end examination	Total	Credits
Computer Organization and Design	Core	UCS1COD	3	40	60	100	2
Programming with Python I	Core	UCS1PP1	3	40	60	100	2
Programming with C	Core	UCS1PWC	3	40	60	100	2
Database Management Systems – I	Core	UCS1DM1	3	40	60	100	2
Discrete Mathematics	Core	UCS1DMA	3	40	60	100	2
Descriptive Statistics	Core	UCS1DST	3	40	60	100	2
Soft Skill Development	Ability enhancement	UCS1SSD	3	40	60	100	2
Environmental Science		USC1EVS	2	40	60	100	2
Practical of UCS1COD and UCS1PP1	Core	UCS1PR1	6		100	100	2
Practical of UCS1PWC and UCS1DM1	Core	UCS1PR2	6		100	100	2
Practical of UCS1DMA and UCS1DST	Core	UCS1PR3	6		100	100	2

Semester - II [Under CBCS Scheme]							
Course	Course Type	Course code	Hrs./ week	Internal assessment	Semester - end examination	Total	Credits
Object Oriented Programming using C++	Core	UCS2OOP	3	40	60	100	2
Programming with Python II	Core	UCS2PP2	3	40	60	100	2
LINUX	Core	UCS2LIN	3	40	60	100	2
Data Structure	Core	UCS2DST	3	40	60	100	2
Calculus	Core	UCS2CAL	3	40	60	100	2
Statistical Methods	Core	UCS2STM	3	40	60	100	2
Digital Marketing	Ability enhancement	UCS2DIM	3	40	60	100	2
Spoken English		USC2SPC	2	40	60	100	2
Practical of UCS2OOP and UCS2PP2	Core	UCS2PR1	6		100	100	2
Practical of UCS2LIN and UCS2DST	Core	UCS2PR2	6		100	100	2
Practical of UCS2CAL and UCS2STM	Core	UCS2PR3	6		100	100	2

#### Semester I

## **Computer Organization and Design**

Course Description			
Semester	Ι		
Course Name	Computer Organization and Design		
Course Code	UCS1COD		
Credit	2		
Hours	45		

# **Course Objectives**

- 1. To understand the basic structure and organization of computers
- 2. To understand the structure and operation of modern processors and their instruction sets
- 3. To understand the working of microcontroller

#### **Course Outcomes**

- 1. Explain the underlying principles of computers
- 2. Analyze the Instruction set architecture
- 3. Analyze the role of various hardware components of processor
- 4. Analyze how data is transferred between various peripheral devices in the computer

Course Code	Course Title	Credits
UCS1COD	Computer Organization and Design	02
Unit I	<ul> <li>Computer Abstractions and Technology: Basic structure and operation of a computer, functional units and their interaction. Representation of numbers and characters.</li> <li>Logic circuits and functions: Combinational circuits and functions: Basic logic gates and functions, truth tables; logic circuits and functions. Minimization with Karnaugh maps. Synthesis of logic functions with and-or-not gates, Nand gates, nor gates. Fan-in and fan-out requirements; tri state buffers. Half adder, full adder, ripple carry adder. (Flip flops) Gated S-R and D latches, edge-triggered D latch. Shift registers and registers. Decoders, multiplexers. Sequential circuits and functions: State diagram and state table; finite state machines and their synthesis.</li> </ul>	15 L

Unit I	I	<ul> <li>Microcontroller: Introduction to microcontroller, Difference between microcontroller and microprocessor Types of microcontrollers, Memory, Instruction set, Applications 8051 microcontroller Architecture</li> <li>Instruction set architectures: Memory organization, addressing and operations; word size, big-endian and little-endian arrangements. Instructions, sequencing. Instruction sets for RISC and CISC (examples Altera NIOS II and Free scale ColdFire). Operand addressing modes; pointers; indexing for arrays. Machine language, assembly language, assembler</li> </ul>	15 L		
		directives. Function calls, processor runtime stack, stack frame. Types of machine instructions: arithmetic, logic, shift, etc. Instruction sets, RISC and CISC examples.			
τ	J <b>nit III</b>	<b>Basic Processor Unit:</b> Main components of a processor: registers and register files, ALU, control unit, instruction fetch unit, and interfaces to instruction and data memories. Datapath. Instruction fetch and execute; executing arithmetic/logic, memory access and branch instructions; hardwired and micro programmed control for RISC and CISC. Basic I/O: Accessing I/O devices, data transfers between processor and I/O devices. Interrupts and exceptions: interrupt requests and processing.	15 L		
Textb 1) 2) Addit 1) 2)	Dooks: Carl Ha Micropp tional F Pattersco R P Jair	umacher et al., Computer Organization and Embedded Systems, 6 ed., McGraw-Hill 20 rocessors and Microcontrollers: Architecture, Kant Krishna <b>References:</b> on and Hennessy, Computer Organization and Design, Morgan Kaufmann, ARM Edition, 201 n, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd., 4th Edition, 201	12 on, 2011 0		
Sr. No.		Practicals of Computer Organization and Design			
1	Study and verify the truth table of various logic gates (NOT, AND, OR, NAND, NOR, EX-OR, and EX-NOR).				
2	Simplify given Boolean expression and realize it.				
3	Design and verify a half/full adder.				
4	Design and verify half/full subtractor.				
5	Design a 4-bit magnitude comparator using combinational circuits.				
6	Design	and verify the operation of flip-flops using logic gates.			
7	Verify t	the operation of a counter.			

8	Verify the operation of a 4-bit shift register.
9	Using SPIM, write and test an adding machine program that repeatedly reads in integers and adds them

	into a running sum. The program should stop when it gets an input that is 0, printing out the sum at that point.
10	Perform arithmetic operations based on 8051 microcontrollers using KEIL simulator.

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	Computer Abstraction and technology	15h	1	1	1
2	Instruction set architecture	15h	2	1	2
3	Basic processor unit	15h	3	1	2

# **Programming with Python-I**

Course Description			
Semester I			
Course Name	Programming with Python-I		
<b>Course Code</b>	UCS1PP1		
Credit	2		
Hours	45		

#### **Course Objectives:**

- 1. Master the fundamentals of writing Python scripts
- 2. Learn core Python scripting elements such as variables and flow control structures
- 3. Discover how to work with lists and sequence data

#### **Course Outcomes:**

- 1. Understand the pros and cons of scripting languages vs. classical programming languages
- 2. Understand Python programming basics and paradigm
- 3. Apply loops, control statements, and string manipulations
- 4. Illustrate the use of lists, tuples & dictionaries for representing compound data

Course Code UCS1PP1	Course Title Programming with Python-1			
Unit I	<b>Overview of Python</b> : History; Features of Python, Execution of a Python Program, Python Interpreter, Comparison of Python with C and Java, Installing Python, Writing and Executing First Python Program, Getting Help, IDLE			
	<b>Data Types, Variables and Other Basic Elements:</b> Comments, Docstrings, Data types- Numeric Data type, Compound Data Type, Boolean Data type, Dictionary, Sets, Mapping, Basic Elements of Python, Variables			
	<b>Input and Output Operations:</b> Input Function, Output Statements, The print () function, The print("string") function, The print (variables list) function, The print(object) function, The print (formatted string) function, Command Line Arguments.			
	<b>Operators</b> : Arithmetic operators, Assignment operators, Unary minus operator, Relational operators, Logical operators, Bitwise operators, Membership operators, Identity operators, Precedence of Operators, Associativity of Operators			
	<b>Control Statements</b> : The if statement, the if else Statement, the "ifelif else" Statement, Loop Statement- while loop, for loop, Infinite loop, Nested loop, the else suite, break statement, continue statement, pass statement, assert statement, return statement			
Unit II	<b>Arrays:</b> Creating Arrays, Indexing and Slicing of Arrays, Basic Array Operations, Arrays Processing, Mathematical Operations on Array, Aliasing Arrays, Slicing and Indexing in NumPy Arrays, Basic slicing, Advanced Indexing, Dimensions of Arrays, Attributes of an Array, The ndim Attribute, The shape Attribute, The size Attribute, The item size Attribute	15 L		
	<b>Functions</b> : Function definition and call, Returning Results, Returning Multiple Values from a Function, Built-in Functions, Difference between a Function and a Method, Pass Value by Object Reference, Parameters and Arguments, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Arbitrary Arguments, Recursive Functions, Anonymous or Lambda Functions, Using Lambda with the filter() Function, Using Lambda with the reduce() Function Modules:			
	<b>OOPS</b> : What is Object Oriented Programming, what is Procedural Programming, Difference between object-oriented Programming and Procedural Programming, Python OOP's Concept- Object, Class, Encapsulation, Inheritance, Polymorphism, Data Abstraction			

Unit III	<b>Strings</b> : Creating Strings, Functions of Strings, Working with Strings, Length of a String, Repeating and Concatenating Strings, Checking Membership, Comparing Strings, Removing Spaces, Finding Substrings, Counting Substrings, Immutability, Splitting and Joining Strings, Changing Case, Checking Starting and Ending of a String, Sorting Strings, searching in the Strings, Testing Methods, Formatting Strings, Finding the Number of Characters and Words, Inserting Substrings into a String	15 L
	List and Tuples: Lists, List Functions and Methods, List Operations, List Slices, Nested Lists, Tuples, Functions in Tuple	
	<b>Dictionaries:</b> Creating a Dictionary, Operators in Dictionary, Dictionary Methods, Using for Loop with Dictionaries, Operations on Dictionaries, Converting Lists into Dictionary, Converting Strings into Dictionary, Passing Dictionaries to Functions, Sorting the Elements of a Dictionary using Lambda, Ordered Dictionaries	

#### **Textbooks:**

- 1. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries, Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition, 2014
- 2. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016.

#### **Additional References:**

1. Programming through Python, M. T Savaliya, R. K. Maurya& G M Magar, Sybgen Learning India, 2020

Sr. No.	Practicals of Programming with Python-1
1	Script and interactive modes; defining a function in the two modes; executing a script; interactively executing a statement list (semicolon-separated sequence of simple statements)
2	Programs using built-in and user-defined functions.
3	Programs based on conditional constructs, the for statement and the range function.
4	Programs related to string manipulation.
5	Write a Python Program for demonstration of an Array, and Adding an element in an array.
6	Programs based on the while statement with break and continue.
7	Programs using built in functions from the time, math and random modules.
8	Programs related to dictionaries.
9	Programs using list comprehensions and anonymous functions.
10	Programs using the built-in methods of the string, list and dictionary classes.
11	Programs using OOP

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	Introduction to Python-1	15h	1	2	1
2	Array, Function, OOPS	15h	2	2	2
3	Strings, List, Tuples, Dictionary	15h	4	2	3

# **Programming With C**

Course Description			
Semester	Ι		
Course Name	Programming with C		
Course Code	UCS1PWC		
Credit	2		
Hours	45		

#### **Course Objectives:**

- 1. Provide basic knowledge of concepts and programming language.
- 2. Describe implementation of operators, data types and loops.
- 3. Illustrate the array, pointer and file handling techniques.
- 4. Use of string functions and evaluation of string operations.

#### **Course Outcome:**

- 1. Explain the basic programming concepts and broad view of programming language.
- 2. Apply programming concepts such as operators, primitive data types, and loops.
- 3. Illustrate the use of an array, pointer, and file handling techniques.
- 4. Demonstrate the use of strings and string handling functions.

Course	Course Title	Credits
Code:	Programming with C	02
UCSIPWC		

Unit I	Programming Paradigms: Use of Algorithms/Flow Charts for problem solving	15 L
	<b>Structure of C program</b> : Header and body, Use of comments. Interpreters' vs compilers, Python vs C. Compilation of a program.	
	Formatted I/O: print(), scan ().	
	<b>Data</b> : Variables, Constants, data types like: int, float char, double and void, short and long size qualifiers, signed and unsigned qualifiers. Compare with datatypes in Python. Compare static typing in C vs dynamic typing in Python	
	Variables: Declaring variables, scope of the variables according to block, hierarchy	

	of data types. Compare explicit declarations in C with implicit declarations in Python.	
	<b>Types of operators</b> : Arithmetic, relational, logical, compound assignment, increment and decrement, conditional or ternary, bitwise and comma operators. Precedence and order of evaluation, statements and Expressions. Automatic and explicit type conversion.	
	<b>Iterations</b> : Control statements for decision making: (i) Branching: if statement, else. if statement, (does the writer mean if-else or nested ifs) switch statement. (ii) Looping: while loop, doWhile, for loop. (iii) Jump statements: break, continue and goto	
	<b>Arrays:</b> (One and two dimensional), declaring array variables, initialization of arrays, accessing array elements. Compare array types of C with list and tuple types of Pythons.	15 L
Unit II	<b>Data Input and Output functions</b> : Character I/O format: getch(), getch(), getchar(), getc(), gets(), putchar(), putc(), puts().	
	<b>Manipulating Strings</b> : Declaring and initializing String variables, Character and string handling functions. Compare with Python strings.	
	<b>Functions</b> : Function declaration, function definition, Global and local variables, return statement, calling a function by passing values.	
	Recursion: Definition, Recursive functions.	
Unit III	<b>Pointer:</b> Fundamentals, Pointer variables, Referencing and de-referencing, Pointer Arithmetic, Using Pointers with Arrays, Using Pointers with Strings, Array of Pointers, Pointers as function arguments, Functions returning pointers.	15 L
	<b>Dynamic Memory Allocation</b> : malloc (), calloc (), realloc (), free () and sizeof operator. Compare with automatic garbage collection in Python.	
	<b>Structure</b> : Declaration of structure, reading and assignment of structure variables, Array of structures, arrays within structures, structures within structures. Compare C structures with Python tuples.	
	Unions: Defining and working with unions.	
	<b>File handling</b> : Different types of files like text and binary, Different types of functions: fopen (), fclose (), fgetc (), fputc (), fgets (), fputs (), fscanf (), fprintf (), getw (), putw (), fread (), fwrite (), fseek ().	
	Textbooks:	
	1. Programming in ANSI C (Third Edition): E Balagurusamy, TMH	
	Additional References:	
	<ol> <li>Pradip Dey, Manas Ghosh, "Programming in C", second edition, Oxford University Press</li> <li>Yashavant P. Kanetkar. "Let Us C", BPB Publications</li> </ol>	

Sr. No.	Practicals of Programming with C
1	Programs to understand the basic data types and I/O.
2	Programs on Operators and Expressions
3	Programs on decision statements
4	Programs on looping.
5	Programs on arrays.
6	Programs on functions.
7	Programs on structures and unions.
8	Programs on pointers
9	Programs on string manipulations.
10	Programs on basic file operations.

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	Introduction to basic concepts of C	15h	1	1	1
2	Array Implementation	15h	2	1	2
3	Pointer Implementation	15h	3	1	1

#### **Database Management Systems-I**

Course Description				
Semester	Ι			
Course Name	Database Management Systems-I			
Course Code	UCS1DM1			
Credit	2			
Hours	45			

#### **Course Objectives:**

1. The objective of this course is to introduce the concept of the DBMS with respect to the relational model, to specify the functional and data requirements for a typical database application and to understand creation, manipulation and querying of data in databases

#### **Course Outcome:**

After completing the course, students will be able to

1. Evaluate business information problem and the requirements of a problem in terms of data

- 2. Design the database schema with the use of appropriate data types for storage of data in database
- 3. Create, manipulate, query and back up the databases
- 4. Analyze various security mechanisms required for database protection

Course Code UCS1DM1	Course Title Database Management System-I		
Unit I	<ul> <li>Introduction to DBMS: Database, DBMS–Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture</li> <li>Data models: Client/Server Architecture, Object Based Logical Model, Record Based Logical Model (relational, hierarchical, network)</li> <li>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)</li> <li>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</li> </ul>	15 L	
Unit II	<ul> <li>ER to Table: Entity to Table, Relationship to tables with and without key constraints.</li> <li>Schema refinement and Normal forms: Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition.</li> <li>Relational Algebra: operations (selection, projection, set operations union, intersection, difference, cross product, Joins –conditional, equi join and natural joins, division)</li> <li>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, Backing Up and Restoring databases</li> </ul>	15 L	
Unit III	<ul> <li>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</li> <li>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 (addate, datediff, day, month, year, hour, min, sec, now, reverse) Joining Tables: inner join, outer join (left outer, right outer, full outer)</li> <li>Subqueries: subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries</li> <li>Database Protection: Security Issues, Threats to Databases, Security Mechanisms, Role of DBA, Discretionary Access Control Views: (creating, altering dropping, renaming and manipulating views)</li> </ul>	15 L	

	DCL Statements: (creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges)			
Textbooks:				
1) Ramez Elmasri & Shamkant B.Navathe, Fundamentals of Database Systems, Pearson Education, Sixth				
Edition, 2010 2) Remelvishnern, Cabrica Database Management Systems, McCraw Hill, 2007				
2) Ramakrishnam, Genrke, Database Management Systems, McGraw-Hill, 2007 3) Ioel Murach, Murach's MySOL, Murach, 2012				
Additional References:				
1) Robert Sheldor	, Geoff Moes, Begning MySQL, Wrox Press, 2005.			

Sr. No.	Practicals of Database Management System
1	For given scenario • Draw an E-R diagram and convert entities and relationships to tables.
2	Write relational algebra queries on the tables created in Practical-1.
3	<ul> <li>Perform the following:</li> <li>Viewing all databases</li> <li>Creating a Database</li> <li>Viewing all Tables in a Database</li> <li>Creating Tables (With and Without Constraints)</li> <li>Inserting/Updating/Deleting Records in a Table</li> <li>Saving (Commit) and Undoing (rollback)</li> </ul>
4	Perform the following: • Altering a Table • Dropping/Truncating/Renaming Tables • Backing up / Restoring a Database
5	<ul> <li>Perform the following:</li> <li>Simple Queries</li> <li>Simple Queries with Aggregate functions</li> <li>Queries with Aggregate functions (group by and having clause)</li> </ul>
6	Queries involving <ul> <li>Date Functions</li> <li>String Functions</li> <li>Math Functions</li> </ul>
7	Join Queries • Inner Join • Outer Join
8	Subqueries • With IN clause • With EXISTS clause
9	<ul> <li>Views</li> <li>Creating Views (with and without check option)</li> <li>Dropping views</li> <li>Selecting from a view</li> </ul>

10	DCL statements
	• Granting and revoking permissions

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	Introduction to DBMS	15h	1	1	2
2	Schema refinement and normal forms	15h	2	2	3
3	Database Protection	15h	4	3	4

#### **Discrete Mathematics**

Course Description			
Semester	Ι		
Course Name	Discrete Mathematics		
Course Code	UCS1DMA		
Credit	2		
Hours	45		

#### **Course Objectives:**

- 1. The purpose of the course is to familiarize the prospective learners with mathematical structures that are fundamentally discrete.
- 2. This course introduces sets and functions, forming and solving recurrence relations and different counting principles. These concepts are useful to study or describe objects or problems in computer algorithms and programming languages.

#### **Course Outcome:**

After completing the course, Student will be able to

- 1. Explain function and relation
- 2. Solve the problems on recurrence relation
- 3. Solve the problems by using the permutation, combination and counting principle
- 4. Examine the properties of graph, application of graph and tree

Course Code: UCS1DMA	Course Title Discrete Mathematics	Credits 02
UNIT I	<b>Functions:</b> Definition of function. Domain, co domain and the range of a function. Direct and inverse images. Injective, surjective and bijective functions. Composite and inverse functions.	15L
	<b>Relations</b> : Definition and examples. Properties of relations, Partial Ordering sets, Linear Ordering Hasse Daigrams, Maximum and Minimum elements, Lattices	
	Recurrence Relations: Definition of recurrence relations, formulating recurrence	

	relations, solving recurrence relations- Back tracking method, Linear homogeneous recurrence relations with constant coefficients. Solving linear homogeneous recurrence relations with constant coefficients of degree two when characteristic equation has distinct roots and only one root, Particular solutions of nonlinear homogeneous recurrence relation, Solution of recurrence relation by the method of generation functions, Applications- Formulate and solve recurrence relation for Fibonacci numbers, Tower of Hanoi, Intersection of lines in a plane, Sorting Algorithms			
Unit II	<ul> <li>Counting Principles Permutations and Combinations: Partition and Distribution of objects, Permutation with distinct and indistinct objects, Binomial numbers, Combination with identities: Pascal Identity, Vandermonde's Identity, Pascal triangle, Binomial theorem, Combination with indistinct objects.</li> <li>Counting Principles: Sum and Product Rules, Two-way counting, Tree diagram for</li> </ul>	15 L		
	solving counting problems, Pigeonhole Principle (without proof); Simple examples, Inclusion Exclusion Principle (Sieve formula) (Without proof).			
Unit III	<b>Graphs</b> : Definition and elementary results, Adjacency matrix, path matrix, representing relations using diagraphs, Warshall's algorithm- shortest path, Linked representation of a graph, Operations on graph with algorithms – searching in a graph; Insertion in a graph, Deleting from a graph, Traversing graph- Breadth First search and Depth-First search.	15 L		
	<b>Trees:</b> Definition and elementary results. Ordered, rooted tree, Binary tree, Complete and extended binary trees, representing binary trees in memory, traversing binary trees, binary search tree, Algorithms for searching and inserting in binary search trees, Algorithms for deleting in a binary search tree.			
Textbooks:				
<ol> <li>Discrete Mathematics and Its Applications, Seventh Edition by Kenneth H. Rosen, McGraw Hill Education (India) Private Limited. (2011)</li> <li>Norman L. Biggs, Discrete Mathematics, Revised Edition, Clarendon Press, Oxford 1989.</li> <li>Data Structures Seymour Lipschutz, Schaum's out lines, McGraw-Hill Inc.</li> </ol>				
Additional F	References:			
1. Elemen	ts of Discrete Mathematics: C.L. Liu, Tata McGraw-Hill Edition			
2. Concret Edition,	2. Concrete Mathematics (Foundation for Computer Science): Graham, Knuth, Patashnik Second Edition, Pearson Education.			

3. Discrete Mathematics: Semyour Lipschutz, Marc Lipson, Schaum's outlines, McGraw-Hill Inc.

Foundations in Discrete Mathematics: K.D. Joshi, New Age Publication, New Delhi.

Sr. No.	Practical of Discrete Mathematics
1	Graphs of standard functions such as absolute value function, inverse function, logarithmic and exponential functions, flooring and ceiling functions, trigonometric functions over suitable intervals.
2	Equivalence Relation and Partial ordering sets.
3	Hasse diagram and Lattices.
4	Recurrence relation.

5	Permutations and Combinations.
6	Different counting principles.
7	Shortest Path algorithms.
8	Operations on graph.
9	Breadth and Depth First search algorithms.
10	Concept of searching, inserting and deleting from binary search trees.

Module/Unit	<b>Course Description</b>	Hrs.	CO No.	PSO No.	PO No.
1	Recurrence Relations	15h	1,2		6
2	Counting Principles	15h	3		1
3	Graphs and Trees	15h	4		1

# **Descriptive Statistics**

Course Description			
Semester	Ι		
Course Name	Descriptive Statistics		
Course Code	UCS1DST		
Credit	2		
Hours	45		

## **Course Objectives:**

The purpose of this course is to familiarize students with the basics of Statistics. This will be essential for prospective researchers and professionals to know these basics.

## **Course Outcome:**

After completing the course, students will be able to

- 1. Understand and present data using table and graphs
- 2. Apply measures of central tendency and dispersion to draw conclusions
- 3. Apply the basic probability rules and theorem in problem-solving
- 4. Apply the method of least squares to estimate the parameters in a regression model

Course	Course Title	Credits 02
UCS1STS	Descriptive Statistics	
Unit I	<ul> <li>Data types and Data presentation:</li> <li>Data Types: attribute, variable, discrete and continuous variable, Different types of scales: nominal, ordinal, interval and ratio.</li> <li>Data presentation: frequency distribution, histogram o gives, curves, stem and leaf display.</li> <li>Introduction to R: Data input, Arithmetic Operators, Vector Operations, Matrix Operations, Data Frames, Built-in Functions. Frequency Distribution, Grouped Frequency Distribution, Diagrams and Graphs, Summary statistics for raw data and grouped frequency distribution.</li> <li>Measures of Central tendency: Mean, Median, mode for raw data, discrete, grouped frequency distribution. Partition Values: Quartiles, Deciles and Percentiles - examples for ungrouped and grouped data.</li> </ul>	15 L
Unit II	<b>Measures dispersion:</b> Concept of dispersion, Absolute and Relative measure of dispersion, characteristics of good measure of dispersion. Range, Semi-interquartile range, Quartile deviation, Standard deviation - Definition, examples for ungrouped and grouped data, effect of shift of origin and change of scale, merits and demerits. Combined standard deviation, Variance. Coefficient of range, Coefficient of quartile deviation and Coefficient of variation (C.V.) <b>Moments:</b> Concept of Moments, Raw moments, Central moments, Relation between raw and central moments. Measures of Skewness and Kurtosis: Concept of Skewness and Kurtosis, measures based on moments, quartiles.	15h
Unit III	<ul> <li>Time-series: Concepts and components of a time series. Representation of trend by freehand curve method, estimation of trend using the moving average method and least-squares methods.</li> <li>Correlation: Concept of correlation, Types and interpretation, Measure of Correlation: Scatter diagram and interpretation; Karl Pearson's coefficient of correlation (r): Definition, examples for ungrouped and grouped data, effect of shift of origin and change of scale, properties; Spearman's rank correlation coefficient: Definition, examples of with and without repetition. Concept of Multiple correlation.</li> <li>Regression: Concept of dependent (response) and independent (predictor) variables, the concept of regression, Types and prediction, difference between correlation and regression, Relation between correlation and regression</li> </ul>	15h
Textbooks:		

- 1. Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. 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.

**Additional References:** 

	1. Mood, A. M. and Graybill, F. A. and Boes D.C. (1974). Introduction to the Theory of Statistics, Ed. 3,				
	McGraw Hill Book Company.				
	2. Hoel P. G. (1971). Introduction to Mathematical Statistics, John Wiley and Sons, New York.				
	3. Hogg, R.V. and Craig R.G. (1989). Introduction to Mathematical Statistics, Ed. MacMillan Publishing				
	Co., New York.				
	4. 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				
Sr.	Practical of Descriptive Statistics				
No.					
1	Data entry using, functions, c(), scan (), Creating vectors, Mathematical Operations: ** $+/-/*//$ , exp, log, log10, etc., creating vector of text type.				
2	Useful functions of R: data frame, matrix operations, seq(), split() etc.				
3	Frequency distribution				
4	Data presentation				
5	Measures of central tendency				
6	Measures of dispersion				
7	Summary Statistics (measures of central tendency, dispersion)				
8	Moments: Raw moments and central moments				
9	Measures of skewness and kurtosis				
10	Correlation and regression				

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	Data types and Data presentation, Introduction to R, Measures of Central tendency	15h	1, 2	2	1
2	Measures dispersion, Moments, Measures of Skewness and Kurtosis	15h	2	2	7
3	Time-series, Correlation, Regression	15h	4	3	2

Course Description	
Semester	Ι

Course Name	Soft Skill Development
Course Code	UCS1SSD
Credit	2
Hours	45

#### **Course Objectives:**

- Understand the significance and essence of a wide range of soft skills
- Learn how to apply soft skills in a wide range of routine social and professional settings
- Learn how to employ soft skills to improve interpersonal relationships
- Learn how to employ soft skills to enhance employability and ensure workplace and career success

## **Course Outcomes**

After completing the course, students will be able to

- Learners will be able to understand the importance and types soft skills
- Learners will develop skills for Academic and Professional Presentations.
- Learners will able to understand Leadership Qualities and Ethics.
- Ability to understand the importance of stress management in their academic & professional life.

Course Code UCS1SSD	Course Title Soft Skill Development	Credits 02
Unit I	<ul> <li>Introduction to Soft Skills</li> <li>Soft Skills: An Introduction – Definition and Significance of Soft Skills; Process, Importance and Measurement of Soft Skill Development.</li> <li>Personality Development: Knowing Yourself, Positive Thinking, Johari"s Window, Physical Fitness</li> <li>Emotional Intelligence: Meaning and Definition, Need for Emotional Intelligence, Intelligence Quotient versus Emotional Intelligence Quotient, Components of Emotional Intelligence, Competencies of Emotional Intelligence, Skills to Develop Emotional Intelligence</li> <li>Positivity and Motivation: Developing Positive Thinking and Attitude; Driving out Negativity; Meaning and Theories of Motivation; Enhancing Motivation Levels</li> <li>Etiquette and Mannerism: Introduction, Professional Etiquette, Technology Etiquette</li> <li>Ethical Values: Ethics and Society, Theories of Ethics, Correlation between Values and Behavior, Nurturing Ethics, Importance of Work Ethics, Problems in the Absence of Work Ethicsand Graphs, Summary statistics for raw data and grouped frequency distribution.</li> </ul>	15 L

Unit II	<b>Basic Skills in Communication:</b> <b>Components of effective communication:</b> Communication process and handling them, Composing effective messages, Non – Verbal Communication: its importance and nuances: Facial Expression, Posture, Gesture, Eye contact, appearance (dress code).	15h
	Communication Skills: Spoken English, Phonetics, Accent, Intonation Employment Communication: Introduction, Resume, Curriculum Vitae, Scannable Resume, Developing an Impressive Resume, Formats of Resume, Job Application or Cover Letter	
	Job Interviews: Introduction, Importance of Resume, Definition of Interview, Background Information, Types of Interviews, Preparatory Steps for Job Interviews, Interview Skill Tips, Changes in the Interview Process, FAQ During Interviews	

	<b>Group Discussion:</b> Introduction, Ambience/Seating Arrangement for Group Discussion, Importance of Group Discussions, Difference between Group Discussion, Panel Discussion and Debate, Traits, Types of Group Discussions, topic based and Case based Group Discussion, Individual Traits	
Unit III	<ul> <li>Academic and Professional Skills:</li> <li>Professional Presentation: Nature of Oral Presentation, planning a Presentation, Preparing the Presentation, Delivering the Presentation</li> <li>Creativity at Workplace: Introduction, Current Workplaces, Creativity, Motivation, Nurturing Hobbies at Work, The Six Thinking Hat Method.</li> <li>Capacity Building: Learn, Unlearn and Relearn: Capacity Building, Elements of Capacity Building, Zones of Learning, Ideas for Learning, Strategies for Capacity Building</li> <li>Leadership and Team Building: Leader and Leadership, Leadership Traits, Culture and Leadership, Leadership Styles and Trends, Team Building, Types of Teams.</li> <li>Decision Making and Negotiation: Introduction to Decision Making, Steps for Decision Making, Decision Making Techniques, Negotiation Fundamentals, Negotiation Styles, Major Negotiation Concepts</li> <li>Stress and Time Management: Stress, Sources of Stress, Ways to Cope with Stress</li> </ul>	15h
Textbooks:		

- 1. Managing Soft Skills for Personality Development edited by B.N.Ghosh, McGraw Hill India, 2017.
- 2. Soft Skills: An Integrated Approach to Maximize Personality, Gajendra S. Chauhan, Sangeeta Sharma, Wiley India

#### **Additional References:**

- 1. Personality Development and Soft Skills, Barun K. Mitra, Oxford Press
- 2. Business Communication, ShaliniKalia, Shailja Agrawal, Wiley India
- 3. Cornerstone: Developing Soft Skills, Sherfield, Pearson India

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
Unit I	Personality Development	15L	1	3	5
Unit II	Employment Communication	15L	2	3	5
Unit III	Leadership and Team Building	15L	3	3	9

# **Semester II**

# **Object Oriented Programming using C++**

Semester	П
Course Name	<b>Object Oriented Programming with C++</b>
<b>Course Code</b>	UCS2OOP
Credit	2
Hours	3

#### **Course Objectives**

The objective of course is to develop programming skills of students, using object oriented programming concepts, learn the concept of class and object using C++ and develop classes for simple applications.

#### **Course Outcomes**

After completing the course, Student will be able to

- 1. Understand object-oriented programming and the difference between structured oriented and objectoriented programming features.
- 2. Explain use of objects and classes for developing programs.
- 3. Apply virtual and pure virtual function & complex programming situations.

4. Illustrate various object-oriented concepts to solve different problems.

Course Code UCS2OOP	Course Title Object Oriented Programming using C++	Credit s 02
Unit I	<ul> <li>Introduction to Programming Concepts: Object oriented programming paradigm, basic concepts of object oriented programming, benefits of object oriented programming. 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.</li> <li>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.</li> <li>Decision Making, Loops, Arrays and Strings: Conditional statements-if, ifelse, switch loops- while, dowhile, for, types of arrays and string and string manipulations</li> <li>Unified Modeling Language (UML): Introduction to UML &amp; class diagrams. Classes, Abstraction &amp; Encapsulation: Classes and objects, Dot Operator, data members, member functions, passing data to functions, scope and visibility of variables in function</li> </ul>	15L
Unit II	<ul> <li>Constructors and Destructors: Default constructor, parameterized constructor, copy constructor, private constructor, destructors.</li> <li>Working with objects: Accessor - mutator methods, static data and static function, access specifiers, array of objects.</li> <li>Polymorphism - Binding-static binding &amp; overloading, constructor overloading function overloading, operator overloading, overloading unary and binary operators.</li> <li>Modelling Relationships in Class Diagrams: Association, Aggregation Composition and examples covering these principles</li> </ul>	15L

Unit III	<b>Inheritance:</b> 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.	15L
	<b>Modelling Relationships:</b> Generalization-Specialization and examples covering these principles Run time Polymorphism - Dynamic Binding, Function overriding, virtual function, pure virtual function, virtual base class, abstract class.	
	<b>Pointers:</b> Introduction to pointers, * and & operators, assigning addresses to pointer variables, accessing values using pointers, pointers to objects & this pointer, pointers to derived classes File Handling: File Stream classes, opening and closing file-file opening modes, text file handling, binary file handling.	
	Applying OOP to solve real life applications: To cover case studies like library management, order management etc. to design classes covering all relationships	

#### Textbooks:

- 1. 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, 2<sup>nd</sup> Edition2015

## Additional References:

- 1. Mastering C++ by Venugopal, Publisher: McGraw-Hill Education, 2017
- 2. Let Us C++ by KanetkarYashwant, Publisher: BPB Publications, 2020
- 3. Object Oriented Analysis and Design by Timothy Budd TMH, 2001

Sr. No.	Practicals of Object Oriented Programming Using C++
1.	Classes and methods
	a. Design an employee class for reading and displaying the employee information, the getInfo() and displayInfo() methods will be used respectively. Where getInfo() will be private method
	<ul> <li>b. Design the class student containing getData() and displayData() as two of its methods which will be used for reading and displaying the student information respectively. Where getData() will be private method</li> </ul>
2.	Classes and methods
	a. Design the class Demo which will contain the following methods: readNo(), factorial() for calculating the factorial of a number, reverseNo() will reverse the given number, isPalindrome() will check the given number is palindrome, isArmstrong() which will calculate the given number is armStrong or not.
	b. Write a program to demonstrate function definition outside class and accessing class members in function definition.

3.	Using friend functions
	a. Write a friend function for adding the two complex numbers, using a single class.
	b. Write a friend function for adding the two different distances and display its sum, using two classes
4.	Constructors and method overloading
	a. Design a class Geometry containing the methods area() and volume() and also overload the area() function .
	b. Design a class Complex for adding the two complex numbers and also show the use of constructor.
5.	Operator Overloading
	a. Overload the operator unary(-) for demonstrating operator overloading.
	b. Overload the + for concatenating the two strings. For e.g., "Py" + "thon" = Python.
6.	Inheritance
	a. Design a class for single level inheritance.
	b. Design a class for multiple inheritance.
	c. Implement hierarchical inheritance.
7.	Virtual functions and abstract classes
	a. Implement the concept of method overriding.
	b. Show the use of virtual functions.
	c. Show the implementation of abstract class.
8.	Exception handling
	a. Show the implementation of exception handling.
	b. Show the implementation for exception handling for strings.
9.	Multi-Threading
	Program to demonstrate multithreading concept.
10.	Mini project on OOPs/ Case Study

Module/Unit	<b>Course Description</b>	Hrs.	CO No.	PSO No.	PO No.
Unit I	Introduction to OOP's	15L	1	2	2
Unit II	Core Concepts of OOP's	15L	2	2	6
Unit III	Advance Concepts of OOP's	15L	3	2	9

# **Programming with Python-II**

Course Description			
Semester	II		
Course Name	Programming with Python –II		
Course Code	UCS2PP2		
Credit	2		
Hours	3		

#### **Course Objectives:**

- 1. Use Python to read and write files
- 2. Make their code robust by handling errors and exceptions properly
- 3. Work with the Python standard library
- 4. Explore Python's object-oriented features
- 5. Search text using regular expressions

#### **Course Outcomes:**

- 1. Demonstrate programs using simple Python statements and expressions.
- 2. Explain files, exceptions, modules and packages in Python for solving problems.
- 3. To develop the skill of designing Graphical user Interfaces in Python
- 4. To Learn how to import modules and packages and game development using Python and the use of Database Connectivity.

Module/Unit	Course Title Programming With Python II	Credits 02
Unit I	<b>Python File Input-Output:</b> Opening and closing files, various types of file modes, reading and writing to files, manipulating directories.	15L
	Iterables: iterators and their problem-solving applications.	
	<b>Exception handling:</b> What is an exception, various keywords to handle exceptions such try, catch, except, else, finally, raise.	
	<b>Regular Expressions:</b> Concept of regular expression, various types of regular expressions, using match function.	

Unit II	<b>GUI Programming in Python</b> (using Tkinter/wxPython/Qt) What is GUI, Advantages of GUI, Introduction to GUI library. Layout management, events and bindings, fonts, colors, drawing on canvas (line, oval, rectangle, etc.) Widgets such as : frame, label, button, check button, entry, list box, message, radio button, text, spin box etc.	15L
Unit III	<b>Database connectivity in Python</b> : Installing MySQL connector, accessing connector module, using connect, cursor, execute & close functions, reading single & multiple results of query execution, executing different types of statements, executing transactions, understanding exceptions in database connectivity <b>Game Design</b> : - Introduction to google collab notebook, Introduction to Jupiter notebook, Pygame:-, how to create the game window, Creating Basic Movements and key Press, changing title and background color, adding images, Adding Sounds Adding Effects etc.	15L

#### **Textbooks:**

Paul Gries , Jennifer Campbell, Jason Montojo, Practical Programming: An Introduction to Computer Science Using Python 3, Pragmatic Bookshelf, 2/E 2014

#### **Additional References:**

- 1. James Payne, Beginning Python: Using Python 2.6 and Python 3, Wiley India, 2010
- 2. A. Lukaszewski, MySQL for Python: Database Access Made Easy, Pact Publisher, 2010
- 3. Making Games with Python & Pygame

Sr. No.	Practicals of Programming with Python-II
1	Programs to read and write files.
2	Programs with iterables and iterators.
3	Program to demonstrate exception handling
4	Program to demonstrate the use of regular expressions
5	Program to show draw shapes & GUI controls.:- a. Advance Calculator b. Simple Interest Form
6	Program to show draw shapes & GUI controls.:- a. Pizza Ordering GUI Form b. BMI Calculator GUI Form
7	Write a Python Program on database connectivity to illustrate the use of DML statements such as update & table
8	Create a Python Game in Google-Collaboratory notebook.
9	Create a Python Game in Jupyter Notebook

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1.	Python File Input-Output	15h	2	2	2
2.	GUI	15h	3	2	2
3.	Database Connectivity& Game Design	15h	4	2	3

# Linux

Course Description		
Semester	Π	
Course Name	Linux	
Course Code	UCS2LIN	
Credit	2	
Hours	3	

## **Course Objectives:**

- 1. This course introduces various tools and techniques commonly used by Linux programmers, system administrators and end users to achieve their day-to-day work in Linux environment.
- 2. It is designed for computer students who have limited or no previous exposure to Linux.

#### **Course Outcomes:**

After completing the course, Student will be able to

- 1. Explain the Importance of Linux in Software Ecosystem and Architecture of Linux
- 2. Apply various command line utilities.
- 3. Design Network using IP address, DNS and different network protocols.
- 4. Apply System Administrative task on network.

Course Code	Course Title	Credits
UCS2LIN	Linux	02
Unit I	<ul> <li>Introduction: History of Linux, Philosophy, Community, Terminology, Distributions, Linux kernel vs distribution. Why learns Linux? Importance of Linux in software ecosystem: web servers, supercomputers, mobile, servers.</li> <li>Installation: Installation methods, Hands on Installation using CD/DVD or USB drive.</li> <li>Linux Structure: Linux Architecture, Filesystem basics, The boot process, init scripts, runlevels, shutdown process, very basic introductions to Linux processes, Packaging methods: rpm/deb Graphical Vs Command line</li> </ul>	15L

Unit II	<b>Graphical Desktop:</b> Session Management, Basic Desktop Operations, Network Management, Installing and Updating Software, Text editors: gedit, vi, vim, emacs, Graphics editors, Multimedia applications.	15L
	Purpose Utilities, Installing Software, User management, Environment variables, Command aliases.	
	Linux Documentation:	
	man pages, GNU info, help command, More documentation sources.	
	File Operations: Filesystem, Filesystem architecture, File types, File attributes, Working with files, Backup, compression	
Unit III	<b>Security:</b> Understanding Linux Security, Uses of root, sudo command, working with passwords, Bypassing user authentication, Understanding ssh	15L
	<b>Networking:</b> Basic introduction to Networking, Network protocols: http, ftp etc., IP address, DNS, Browsers, Transferring files. ssh, telnet, ping, traceroute, route, hostname, networking GUI.	
	<b>Basic Shell Scripting:</b> Features and capabilities, Syntax, Constructs, modifying files, Sed, awk command, File manipulation utilities, dealing with large files and Text, String manipulation, Boolean expressions, File tests, Case, Debugging, Regular expressions	
Textbooks: 1. "Linux	Command line and Shell Scripting Bible", Richard Blum, Wiley India.	

- 2. "Unix: Concepts and Applications", Sumitabha Das, 4th Edition, McGraw Hill.
- 3. "Official Ubuntu Book", Matthew Helmke& Elizabeth K. Joseph with Jose Antonio Rey and Philips Ballew, 8th Ed.

#### **Additional References:**

- 1. "Linux Administration: A Beginner's Guide", Fifth Edition, Wale Soyinka, Tata McGraw-Hill, 2008.
- 2. "Linux: Complete Reference", Richard Petersen, 6th Edition, Tata McGraw-Hill
- 3. "Beginning Linux Programming", Neil Mathew, 4th Edition, Wiley Publishing, 2008.

Sr. No.	Practicals of Linux
1.	Installation of Ubuntu Linux operating system.
	a. Booting and Installing from (USB/DVD)
	b. Using Ubuntu Software center / Using Synaptic Explore useful software packages.

2	<b>Beco</b> a. b. c. d.	ming an Ubuntu power user Administering system and User setting Learning Unity keyboard Using the Terminal Working with windows programs				
3	File	System Commands: touch, help, man, more, less, pwd, cd, n	nkdir, ri	mdir, ls,	find, ls, et	c
	File chgr	handling Commands: cat, cp, rm, mv, more, file, wc, od, cr p, gzip and gunzip, zip and unzip, tar, ln, umask,, chmod, chg	np, diff grp, cho	, comm, wn, etc	chmod, ch	iown,
4.	Gener stty, e	ral purpose utility Commands:cal, date, echo, man, printf, p tc	basswd,	script, v	vho, unam	e, tty,
	Simp	e Filters and I/O redirection: head, tail, cut paste, sort, grep	o family	v, tee, un	iq, tr, etc.	
	Netwo	orking Commands: who, whoami, ping, telnet, ftp, ssh, etc				
5.	Editors: vi, sed, awk					
6.	Working and Managing with processes- sh, ps, kill, nice, at and batch etc.					
7.	<b>Shell scripting I:</b> Defining variables, reading user input, exit and exit status commands, , expr, test, [], if conditional, logical operators					
8.	Shell scripting II: Conditions (for loop, until loop and while loop) arithmetic operations, examples					
9.	Shell scripting III: Redirecting Input / Output in scripts, creating your own Redirection					
10.	Installation of C/C++/Java/Python Compiler and creating an environment for app development. Basic programming using C and Python Languages.					
Module/UnitCourse DescriptionHrs.CCNo		CO No.	PSO No.	PO No.		
1.		Introduction, Installation, Linux Structure	15h	1	3	1
2.		Graphical Desktop, Command Line , Linux Documentation, File Operations	15h	2	2	1
3.Security, Networking, Basic Shell Scripting15h33		3	3			

# **Data Structures**

<b>Course Description</b>		
Semester	II	
Course Name	Data Structures	
<b>Course Code</b>	UCS2DST	

Credit	2
Hours	45

#### **Course Objectives:**

- 1. Explain data structures and different abstract data types.
- 2. Learn how to implement linked and linear structures.
- 3. Discuss various probing techniques and clustering.
- 4. Demonstrate binary tree traversals, heap and search trees.

#### **Course Outcomes:**

- 1. Describe data structures and different abstract data types.
- 2. Apply implementation of linked and linear structures.
- 3. Differentiate linear probing, rehashing and clustering.
- 4. Evaluate different binary tree traversals, heap and search trees and its implementation.

Course Code: UCS2DST	Course Title Data Structure	Credits 02
	Abstract Data Types: Introduction, The Date Abstract Data Type, Bags, Iterators. Application	15 L
∐nit I	Arrays: Array Structure, Python List, Two Dimensional Arrays, Matrix Abstract Datatype, Application	
	Sets and Maps: Sets-Set ADT, Selecting Data Structure, List based Implementation, Maps-Map ADT, List Based Implementation, Multi-Dimensional Arrays-Multi-Array ADT, Implementing Multiarray, Application	
	Algorithm Analysis: Complexity Analysis-Big-O Notation, Evaluating Python Code, Evaluating Python List, Amortized Cost, Evaluating Set ADT, Application	
	<b>Searching and Sorting:</b> Searching-Linear Search, Binary Search, Sorting- Bubble, Selection and Insertion Sort, <b>Working with Sorted Lists</b> -Maintaining Sorted List, Maintaining sorted Lists	
	<b>Linked Structures:</b> Introduction, Singly Linked List-Traversing, Searching, Prepending and Removing Nodes, Bag ADT-Linked List Implementation. Comparing Implementations, Linked List Iterators, More Ways to Build Linked Lists, Applications-Polynomials.	15 L
Unit II	<b>Stacks:</b> Stack ADT, Implementing Stacks-Using Python List, Using Linked List, Stack Applications-Balanced Delimiters, Evaluating Postfix Expressions.	
	<b>Queues:</b> Queue ADT, Implementing Queue-Using Python List, Circular Array, Using List, Priority Queues- Priority Queue ADT, Bounded and unbounded Priority Queues.	
	Advanced Linked List: Doubly Linked Lists-Organization and Operation, Circular Linked List-Organization and Operation, Multi Lists	

	<b>Recursion:</b> Recursive Functions, Properties of Recursion, Its working, Recursive Applications.	15 L
Unit III	Hash Table: Introduction, Hashing-Linear Probing, Clustering, Rehashing, Separate Chaining, Hash Functions Advanced Sorting: Merge Sort, Quick Sort, Radix Sort, Sorting Linked List	
	<b>Binary Trees:</b> Tree Structure, Binary Tree-Properties, Implementation and Traversals, Expression Trees, Heaps and Heapsort, Search Trees	
Textbooks:		

- 1. Data Structure and algorithm Using Python, Rance D. Necaise, 2016 Wiley India Edition
- 2. Data Structure and Algorithm in Python, Michael T. Goodrich, Robertom Tamassia, M. H. Goldwasser, 2016 Wiley India Edition

#### **Additional References:**

- 1. Data Structure and Algorithmic Thinking with Python- Narasimha Karumanchi, 2015, Careermonk Publications
- 2. Fundamentals of Python: Data Structures, Kenneth Lambert, Delmar Cengage Learning

Sr.No.	Practicals of Data Structure
1	Implement Linear Search to find an item in a list.
2	Implement binary search to find an item in an ordered list
3	Implement Sorting Algorithms a. Bubble sort b. Insertion sort c. Quick sort d. Merge sort
4	Implement use of Sets and various operations on Sets.
5	Implement working of Stacks. (pop method to take the last item added off the stack and a push method to add an item to the stack)
6	Implement Program for a. Infix to Postfix conversion b. Postfix Evolution
7	Implement the following a. A queue as a list which you add and delete items from. b. A circular queue. (The beginning items of the queue can be reused).
8	Implement Linked list and demonstrate the functionality to add and delete items in the linked list.
9	Implement Binary Tree and its traversals.
10	Recursive implementation of a. Factorial b. Fibonacci c. Tower of Hanoi

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	Introduction to data structures and ADTs	15h	1	1	1
2	Linked and linear structures	15h	2	1	2
3	Clustering, Binary tree	15h	4	1	2

# Calculus

<b>Course Description</b>		
Semester	II	
Course Name	Calculus	
<b>Course Code</b>	UCS2CAL	
Credit	2	
Hours	45	

## **Course Objectives:**

The course is designed to have a grasp of important concepts of Calculus in a scientific way. It covers topics from as basic as definition of functions to partial derivatives of functions in a gradual and logical way. The learner is expected to solve as many examples as possible to a get compete clarity and understanding of the topics covered.

#### **Course Outcome:**

After completing the course, Student will be able to

- 1. Recall the limit, continuity and derivative of a function
- 2. Analyze the properties of Functions
- 3. Evaluate partial derivatives, directional derivatives, maxima and minima of functions of two variables
- 4. Classify the first order differential equation

Course Code: UCS2CAL	Course Title Calculus	Credits 02
Unit I	<b>Derivatives and its applications:</b> Review of Functions, limit of a function, continuity of a function, derivative of a function. Derivative in Graphing and Applications: Analysis of Functions: Increasing, Decreasing, Concavity, Relative Extrema. Graphing Polynomials, Rational Functions, Cusps and Vertical Tangents. Absolute Maxima and Minima, Applied Maximum and Minimum Problems, Newton's Method.	15 L
Unit II	<b>Partial derivatives and its applications:</b> Functions of Two or More Variables, Limits and Continuity, Partial Derivatives, Differentiability, Differentials and Local Linearity, Chain Rule, Directional Derivatives and Gradients, Tangent Planes and Normal Vectors, Maxima and Minima of Functions of Two Variables.	15 L
Unit III	<ul> <li>First order first degree differential equations: Solutions of homogeneous and non-homogeneous differential equations of first order and first degree, Notion of partial derivative, solving exact differential equations. Rules for finding integrating factor (I.F) (without proof) for non-exact equations such as:</li> <li>(a) 1Mx+Ny is an I.F., if Mx+Ny0 and Mdx+Ndy is homogeneous</li> <li>(b) 1Mx-Ny is an I.F., if Mx-Ny0 and Mdx+Ndy is of the type f1xyydx+f2xyxdy=0</li> <li>(c) efxdx is an I.F., if N0 and 1N∂M∂y-∂N∂x is a function of x alone say f(x)</li> <li>(d) egydy is an I.F., if M0 and 1M∂N∂x-∂M∂y is a function of y alone say g(y)</li> <li>Finding solutions of first order differential equations of the type dydx+Pxy=Q(x)yn, for n0. Applications to orthogonal trajectories, population growth, and finding the current at a given time.</li> </ul>	15 L
Textbooks: 1. Calculus & sons	s: Early Transcendental (10th Edition): HOward Anton , Irl Bivens, Stephen Davis, Joh , 2012	m Wiley
Additional Re 1. Calcult 1995	eferences: us and analytic geometry (9th Edition) : George B Thomas, Ross L Finney, Addison W	vesley,

- 2. Calculus: Early Transcendental (8th Edition): James Stewart, Brooks Cole, 2015
- 3. Calculus (10th Edition): Ron Larson, Bruce H. Edwards, Cengage Learning, 2013
- 4. Thomas Calculus (13th Edition): George B. Thomas, Maurice D. Weir, Joel R. Hass, Pearson, 2014

Sr. No.	Practicals of Calculus
1	Continuity of functions, Derivative of Functions
2	Increasing, Decreasing, Concave up and Concave down functions

3	Relative maxima, relative minima, absolute maxima, absolute minima
4	Newton's method to find approximate solution of an equation
5	Calculation of Partial derivatives of functions
6	Local linear approximation and directional derivatives

7	Maxima and minima of functions of two variables
8	Solution of a Homogeneous, Non-Homogeneous differential equation
9	Solution of a Exact, Non-exact differential equation
10	Solution of a first order first degree differential equation by using integrating factor

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	Derivatives and its applications	15h	1,2	1,2	6
2	Partial derivatives and its applications	15h	3	2	6
3	First order first degree differential equations	15h	4	1	6

# **Statistical Methods**

<b>Course Description</b>			
Semester	II		
Course Name	Statistical Methods		
<b>Course Code</b>	UCS2STM		
Credit	2		
Hours	45		

## **Course Objectives**

1. The purpose of this course is to familiarize students with the basics of Statistics this will be essential for prospective researchers and professionals to know these basics.

#### **Course Outcomes**

After completing the course, Students will be able to

- 1. Illustrate different probability functions with respect to discrete and continuous random variables.
- 2. Determine the hypotheses and validate using appropriate statistical tests.
- 3. Recognize when analysis of variance (ANOVA) is appropriate and be able to perform one-way and two-way ANOVAs.
- 4. CO4 Comparison of parametric and nonparametric tests and identification of applications where nonparametric approaches are appropriate.

Course Code UCS1STM	e Code Course Title Statistical Methods Cu ISTM				
Unit I	<b>Probability</b> : 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.				
	<b>Random Variables</b> : 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.				
Unit II	<b>Mathematical Expectation and Variance</b> : Expectation of a function, Variance and S.D of a random variable, properties.	15h			
	<b>Standard Probability distributions</b> : Introduction, properties, examples and applications of each of the following distributions: Binomial distribution, Norma distribution, Chi-square distribution, t distribution, F distribution				
Unit III	<b>Hypothesis testing:</b> one-sided, two-sided hypothesis, critical region, p-value, tests based on t, Normal and F, confidence intervals.	15h			
	Analysis of variance: one-way, two-way analysis of variance				
	<b>Non-parametric tests:</b> need of non-parametric tests, sign test, Wilcoxon's signed rank test, run test, Kruskal-Wallis tests.				
Textbooks:					
1. Gupta, S	.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. and Sons,	New			
2. Goon, A Sixth Re	<ol> <li>Delhi</li> <li>Goon, A. M., Gupta, M. K. and Dasgupta, B. (1983). Fundamentals of Statistics, Vol. 1, Sixth Revised Edition, The World Press Pvt. Ltd., Calcutta.</li> </ol>				
Additional Ref	erences: M and Gravhill F. A. and Boes D.C. (1074). Introduction to the Theory of Statistic	Ed 3			
McGraw	Mood, A. M. and Grayoni, F. A. and Boes D.C. (1974). Introduction to the Theory of Statistics, Ed. 5, McGraw Hill Book Company.				
2. Hoel P.	Hoel P. G. (1971). Introduction to Mathematical Statistics, John Wiley and Sons, New York.				
Co., Nev	Co., New York.				
4. Walpole	Walpole R. E., Myers R. H. and Myers S. L. (1985), Probability and Statistics for Engineers and Scientists				
5. Agarwal	. Agarwal, B. L. (2003). Programmed Statistics, Ed 2, New Age International Publishers, New Delhi.				
Sr No	Duppticals of Statistical Mathada				

Probability
a. Examples based on Probability definition: classical, axiomatic
b. Examples based on elementary Theorems of probability

2	Conditional probability and independence a. Examples based on Conditional probability b. Examples based on Bayes" theorem c. Examples based on independence
3	Problems based on binomial distribution

4	Problems based on normal distribution
5	Discrete random variable- a. Probability distribution of discrete random variable. b. Probability mass function.
6	<ul><li>Continuous random variable-</li><li>a. Probability distribution of continuous random variable.</li><li>b. Probability density function.</li></ul>
7	t test, normal test, F test
8	Analysis of Variance
9	Non parametric tests- I
10	Non- Parametric tests – II

Module/Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
Ι	Probability, Random Variables	15h	1	2	1
II	Mathematical Expectation and Variance, Standard Probability distributions	15h	1	2	1, 2
III	Hypothesis testing, Analysis of variance, Non-parametric tests	15h	2, 3, 4	2, 3	2, 7

# **Digital Marketing**

Course Description			
Semester	II		
Course Name	Digital Marketing		
<b>Course Code</b>	UCS2DIM		
Credit	2		
Hours	45		

#### **Course Objectives**:

- 1. To understand the key concepts of social media and Digital Marketing
- 2. To understand Legal and Ethical issues in digital marketing
- 3. To learn various tools of social media and Digital Marketing
- 4. To acquaint with the techniques of SEO and SEM

#### **Course Outcome:**

After completion of this course, student will be able to:

- 1. Explain use of Digital Media in Marketing.
- 2. Aware about cyber laws related to digital marketing
- 3. Explain benefits of digital marketing over traditional marketing
- 4. Summarize various tools of social media and Digital Marketing
- 5. Use SEO and SEM techniques to improve website's performance.

Course Code USCDIM	Course Title Digital Marketing					
Unit I	<b>Introduction of Digital Marketing:</b> Definition of digital marketing, origin of digital Marketing, Benefits of Digital marketing, The Digital Landscape, P-O-E-M Framework, Digital Advertising Market in India, Segmenting and Customizing Messages, Digital Marketing Plan					
	<b>Digital Marketing VS Traditional Marketing :</b> Difference between digital marketing and traditional marketing. Benefits of Traditional Marketing, The Downside to Traditional Marketing. Why Digital Marketing Wins Over Traditional Marketing, How We Use Both Digital & Traditional marketing?					
	<b>Digital marketing and law:</b> Legal and Ethical issues in digital marketing, Privacy, Digital property and Digital protection, Security of clients and service provider, Cyber laws- Relevant provisions of information Technology Act 2000					
Unit II	<b>Tools of Social Media Marketing:</b> Meaning, Purpose, types of social media websites, social media engagement, Target audience	15L				
	<b>Facebook Marketing:</b> Business through Facebook Marketing, Creating Advertising Campaigns, Adverts, Facebook Marketing Tool					
	<b>LinkedIn Marketing:</b> Introduction and Importance of LinkedIn Marketing, Framing LinkedIn Strategy, Lead Generation through LinkedIn, Content Strategy, Analytics and Targeting					
	Twitter Marketing: Introduction to Twitter Marketing How Twitter Marketing is different than other forms of digital marketing, Framing content strategy, Twitter Advertising Campaigns					
	<b>Content Marketing:</b> 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.					
Unit III	<b>Mobile Advertising:</b> Forms of Mobile Marketing, Features, Mobile Campaign Development, Mobile Advertising Analytics	15L				
	<b>Email marketing:</b> Types of Emails, Mailing List, Email Marketing tools, Email Deliverability & Email Marketing automation					
	Search Engine optimization and Search Engine Marketing					
	Meaning, Common SEO techniques, Understanding Search Engines, basics of Keyword search, Google rankings, Link Building, Steps to optimize website					
	<b>Search Engine Marketing:</b> Introduction to SEM, Introduction to Ad words-Google Ad Words, Ad Words fundamentals, Ad Ranks, Creating Ad Campaigns, display marketing, Buying Models cost per Click (CPC), Cost per Milli (CPM), Cost per Lead (CPL), Cost per Acquisition (CPA).					

# Textbooks: "Digital Marketing": Seema Gupta, McGraw Hill Education 2nd Edition "Introduction to Digital Marketing 101": Cecilia Figuera, bpb Publications "The Art of Digital Marketing ": Ian Dodson , Wile Publication Additional References: "Moving from Traditional to Digital": Philip Kotler Marketing 4.0, Wile Publications "Fundamentals of Digital Marketing ", Puneet Singh Bhatia, Pearson Edition "Digital Social Media Marketing", Prof. Nitin C. Kamat, Mr. Chinmay Nitin Kama, Himalaya Publishing House Pvt. Ltd. "Social Media Marketing: A Strategic Approach", Melissa S. Barker, Donald I. Barker, Nicholas F.

4.	"Social Media Marketing: A Strategic Approach", Melissa S. Barker, Donald I. Barker, Nicholas F.
	Bormann, Debra Zahay, Mary Lou Roberts, Cengage Publication

Module/Unit	Course Description	Hrs.	СО	PSO	РО
			No.	No.	No.
1	Introduction of Digital Marketing	15L	1,2,3	2	3,11
2	Tools of Social Media Marketing	15L	4	2	7
3	Search Engine optimization and Search Engine Marketing	15L	5	2	6

#### Practical: Practical's of UCS1COD+UCS1PP1

Module/Unit	Course Description	Hrs.	CO	PSO	РО
			No.	No.	No.
1	Computer Organization and Design	15L	1	1	1
2	Programming with python-1	15L	3	2	3

#### Course: Practical's of UCS1PWC+UCS1DM1

Module/Unit	Course Description	Hrs.	CO	PSO	РО
			No.	No.	No.
1	Programming with C	15L	2	2	3
2	Datbase Management System-1	15L	3	2	3

#### Practical's of UCS1DMA+UCS1DST

Module/Unit	Course Description	Hrs.	СО	PSO	РО
			No.	No.	No.

1	Discrete Mathematics	15L	1	2	5
2	Descriptive Statistics	15L	3	3	6