TUMKUR UNIVERSITY

Board of Studies in Computer Science

Curriculum Structure and Syllabus for I and II Semester

Bachelor of Science

in

Data Science

Choice Based Credit System

2024-25 Onwards

BOS Computer Science Committee Members

1.	**Dr. Kusuma Kumari B.M** MCA Coordinator & Assistant Professor Department of Studies and Research in Computer Applications Tumkur University, Tumakuru	Chairperson
2.	**Dr. Ramani. R** Assistant Professor Department of Computer Science University College of Science Tumkur University, Tumakuru	Member
3.	**Dr. Chandrali Baishya** Associate Professor Department of Studies and Research in Mathematics Tumkur University, Tumakuru	Member
4.	**Dr. Prakash B.R** Assistant Professor Department of Computer Science Government First Grade College, Tipatur.	Member
5.	**Sri. Mohan Kumar N** Assistant Professor Department of Computer Science Y.E.R Government First Grade College, Pavagada.	Member
6.	**Capt. Ramalinga Reddy S** Assistant Professor Sri Siddaganga Arts and Commerce Evening College Tumkur.	Member
7	**Dr. Manjunath S** Assistant Professor, Department of Computer Science Y.E.R Government First Grade College, Pavagada.	Member
8	**Dr. Nagamani H.S** Associate Professor, Department of Computer Science Maharani Claster University, Bengaluru.	Member
9	**Dr. Asha Gowda Karegowda** Associate Professor , Department of MCA Siddaganga Institute of Technology, Tumkur.	Member
10	**Dr. Sumathi R Gowda** Assistant Professor MCA Department, Karnataka State Open University, Muktha Gangotri, Mysore.	Member
11.	**Dr. Haridas S.** Associate Professor Department of Computer Science Government First Grade College, Tumkur.	Member

SEMESTER – I

Sl. No	Paper	Title of the Paper	Instruction Hrs. per Week	No. of Credits	Duration of the Exam.	Marks		
						Internal Assessment	Semester End Exam.	Total
1	BSCDST 101	Discrete Mathematics	4	4	3 Hrs.	20	80	100
2	BSCDSP 102	Soft Skill Enhancement Lab	4	2	3 Hrs.	10	40	50
3	BSCDST 103	Principles of Programming Languages	4	4	3 Hrs.	20	80	100
4	BSCDSP 104	MS Office and PPL Lab	4	2	3 Hrs.	10	40	50
5	BSCDST 105	Computer Fundamentals and Programming in C	4	4	3 Hrs.	20	80	100
6	BSCDSP 106	Programming in C Lab	4	2	3 Hrs.	10	40	50
TOTAL				**18**				**450**

NOTE:
Theory: 1 Hr. = 1 Credit **Practical: 2 Hrs. = 1 Credit**

BSCDST: BSC Data Science Core Paper Theory
BSCDSP: BSC Data Science Core Paper Practical

SEMESTER – II

Sl. No	Paper	Title of the Paper	Instruction Hrs. per Week	No. of Credits	Duration of the Exam.	Marks		
						Internal Assessment	Semester End Exam.	Total
1	BSCDST 201	Probability and Statistics for Data Science	4	4	3 Hrs.	20	80	100
2	BSCDSP 202	Probability and Statistics for Data Science Lab	4	2	3 Hrs.	10	40	50
3	BSCDST 203	Data Structure Using C	4	4	3 Hrs.	20	80	100
4	BSCDSP 204	Data Structure Using C Lab	4	2	3 Hrs.	10	40	50
5	BSCDST 205	Python Programming	4	4	3 Hrs.	20	80	100
6	BSCDSP 206	Python Programming Lab	4	2	3 Hrs.	10	40	50
TOTAL				**18**				**450**

NOTE:
Theory: 1 Hr. = 1 Credit **Practical: 2 Hrs. = 1 Credit**

BSCDST: BSC Data Science Core Paper Theory
BSCDSP: BSC Data Science Core Paper Practical

Internal Assessment Marks for Theory and Practical

Internal Assessment Marks Allotment for Theory	
Internal Test 1	05
Internal Test 2	05
Assignment	05
Seminar/Quiz	05
TOTAL	**20**

Internal Assessment Marks Allotment for Practical	
Internal Test	05
Record and Attendance	05
TOTAL	**10**

Evaluation Scheme for Lab Examination

Assessment Criteria	
Writing 2 Programs	**10**
Execution of 2 Programs	**20**
Viva	**10**
Total	**40 Marks**

First Semester			
Subject Name: Discrete Mathematics			
Subject Code	BSCDST 101	CIE Marks	20
No of Hours/Week	4	SEE Marks	80
Total Hours:	60	Credits	4

COURSE OBJECTIVES:

1. Provide a survey of Discrete Mathematics, the study of finite systems, needed in computer science.
2. Further develop the mathematical concepts and technique which should serve as a preparation for more advanced quantitative courses.

COURSE OUTCOMES:

On successful completion of this course; the student will be able to:

1. Verify the correctness of an argument using propositional and predicate logic and truth tables.
2. Solve problems on Sets, Relations and functions.
3. Construct the proofs by using proof by mathematical induction, principles of counting and application of permutation and combination
4. Demonstrate the ability to solve problems using operations of matrices and determinants.

Unit 1	**Mathematics Logic**	**Hours 15**
	Definition of proposition, proposition variables, negation of statements, logical connectives and truth tables, predicates and quantifiers, Conditional statement, contra-positive and converse, bi-conditional, tautology, contradiction, logical equivalence, properties of proposition operation-commutative, associative, distributive, idempotent negation. Introduction to proofs. Methods of proof.	
Unit 2	**Set Theory**	**Hours 10**
	Definition of a set, sub-set with examples, Venn diagrams, types of sets-equal sets, null set, disjoint sets, finite set, infinite set, power set, cardinality of set. Operations on sets. Definition of a relation with examples, types of relations. Definition of a function with examples, types of function.	
Unit 3	**Counting**	**Hours 10**
	Principle of mathematical induction, problems on principle of mathematical induction. Fundamental principle of counting (statement with examples only), permutations-definition and problems. Combinations - definition and problems. Pigeon hole principle- statement and proof, extended pigeonhole principle- statement and proof, applications of Pigeon hole principle	
Unit 4	**Matrices and Determinants**	**Hours 15**
	Definition of matrix and order of matrix, types of matrices, operations on matrices, inverse of a matrix, determinant of matrix, properties of determinant, applications of determinants and matrices. Finding rank of a matrix, normal form, echelon form, Cayley Hamilton theorem, Eigen values and Eigen vectors.	

	Graphs and Trees	Hours 10
Unit 5	**Graphs** : Introduction to graph theory, types of graphs, subgraphs, distance, standard graphs, bipartite graph, regular graph, complement of a graph, graph isomorphism, graph operations. Eulerian and Hamiltonian graphs, planar graphs, coloring graphs. **Trees**: Introduction, applications of trees, tree traversal, rooted trees, binary trees, spanning trees, minimum spanning trees, Weighted trees and Prefix Codes. Directed graphs: Fundamentals of digraphs, computer recognition - zero-one matrices and directed graphs, out-degree, in-degree, connectivity, orientation.	

TEXT BOOKS:

1. Ralph P. Grimaldi, Discrete and Combinatorial Mathematics, 5thEdition, Pearson Education, 2004.
2. C. L. Liu and D. P. Mohapatra, Elements of Discrete Mathematics: A Computer Oriented Approach, McGraw Hill, Third Edition, 2012.
3. C. L. Liu, Elements of Discrete Mathematics, Tata McGraw-Hill, 2000.
4. B. S. Grewal, Higher Engineering Mathematics, Khanna Publishers.

REFERENCE BOOKS:

1. Kenneth H Rosen: Discrete Mathematics and its Applications, McGraw Hill publications, 7th edition, 2007.
2. J. P. Tremblay and R.P. Manohar: Discrete Mathematical Structures with applications to Computer Science, Mc Graw Hill Ed. Inc. 2002.
3. Krishnamurthy, "Combinatorics Theory & Application", East-West Press Pvt. Ltd., New Delhi.

First Semester			
Subject Name: Soft Skill Enhancement Lab			
Subject Code	BSCDSP 102	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours:	60	Credits	2

LIST OF PRACTICAL PROGRAMS

PART A:

1. Basics of Communication Skills and Listening comprehension.
2. Group Discussion – dynamics of group discussion, Lateral thinking, Brainstorming.
3. Personality Development: Decision-Making, Problem Solving, Goal Setting, Time Management & Positive Thinking.
4. Writing Skills: Letter writing, Essays for Competitive examinations.
5. Resume writing – structure and presentation, planning, defining the career objective.
6. Body Language-To reveals your inner self and personality.
7. Meetings- making meeting effective, chairing a meeting, decision-making, seeking opinions, interrupting and handling interruptions, clarifications, closure, negotiation skills.
8. Interview Skills – concept and process, pre-interview planning, opening strategies, answering strategies, interview through tele and video-conferencing.

PART B:

1. Mock interview.
2. Official Mail id Creation
3. Micro Presentation
4. NAD registration Step by Step
5. Use of word processing, presentation tools for communication and presentation skills.
6. Use of Google forms, drive for collaborative work.
7. Use of spreadsheet for data interpretation and data analysis.
8. Using netiquettes in online mode of communication using Zoom / Google Meet / MS-Teams etc.

TEXT BOOKS:

1. Personality Development and SOFT SKILLS, BARUN K. MITRA Oxford University Press.
2. M. Ashraf Rizvi, “Effective Technical Communication”, Tata McGraw-Hill Publishing Company Ltd. 2005.

REFERENCE BOOKS:

1. Andrea J. Rutherford, “Basic Communication Skills for Technology”, 2nd Edition, Pearson Education, 2007.
2. Meenakshi Raman & Sangeeta Sharma, “Technical Communication”, Oxford University Press, 2011. DELTA’s key to the Next Generation TOEFL Test: “Advanced Skill Practice,” New Age International (P) Ltd., Publishers, New Delhi.

First Semester			
Subject Name: Principles of Programming Languages			
Subject Code	BSCDST 103	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours:	60	Credits	4

COURSE OBJECTIVES:

1. For the beginner students learning by using text based is difficult to programming concepts.
2. By using flowcharts, Students can concentrate on programming concepts rather than all the nuances of a typical programming language

COURSE OUTCOMES:

After studying this course, the students able to,

1. To introduce the various programming paradigms.
2. To understand the evolution of programming languages.
3. To understand the concepts of OO languages, functional languages, logical, scripting Languages and modern programming languages.
4. To introduce the notations to describe the syntax and semantics of programming languages.

Unit 1	**Preliminary Concepts**	**Hours 15**
	Reasons for studying concepts of programming languages, programming domains, language evaluation criteria, influences on language design, language categories, language design trade-offs, implementation methods, programming environments. Syntax and Semantics: General problem of describing syntax, formal methods of describing syntax, attribute grammars, describing the meanings of programs, Main Programming Structures, Algorithm, Flow Charts & Pseudocode, documenting a Program, understand programming logic through Flogorithm	
Unit 2	**Programming Language**	**Hours 15**
	Introduction, Key Concepts and overview of: Functional Programming Languages, Logic Programming Language, Scripting Languages. Object-Oriented Programming: Design issues for OOP, Using the Unified Modelling Language to Design Classes, Inheritance, Polymorphism GUI Programming Languages: GUI Applications, Graphical User Interfaces, Designing the User Interface and Event Handler for a GUI Program.	
Unit 3	**Basic concepts of Program**	**Hours 10**
	Variables and scopes: Introduction, names, variables, concept of binding, scope, scope and lifetime, referencing environments, named constants, Operators Data types: Introduction, primitive, character, string types, user defined ordinal types, array, associative arrays, record, tuple types, list types, union types, pointer and reference types, type checking, strong typing, type equivalence.	
Unit 4	**Statements and Control Structures**	**Hours 10**
	Expressions and Statements: Arithmetic expressions, overloaded operators, type conversions, relational and Boolean expressions, short- circuit evaluation, assignment	

	statements, mixed mode assignment. Control Structures – introduction, selection statements, iterative statements, unconditional branching, guarded commands.	
Unit 5	**Sub Programs**	**Hours 10**
	Introduction to Sub-Programs (Functions): Defining and Calling a Module, Local Variables, Passing Arguments to Modules, Global and Local Variables. Calling subprograms indirectly, implementing subprograms, General semantics of calls and returns, implementing subprograms with stack-dynamic local variables, nested subprograms, blocks. Concept of Abstract Data types. Library Functions, Introduction to Recursion and Problem Solving with Recursion. Introduction to Menu-Driven Programs: Using a Loop to Repeat the Menu, Modularizing a Menu-Driven Program, Multiple-Level Menus	

TEXT BOOKS:

1. Concepts of Programming Languages, 11E, Robert Sebesta, Pearson, Global Edition.

REFERENCE BOOKS:

1. Starting out with Programming Logic & Design, 3E, Tony Gaddis, Pearson Education
2. Programming language design concepts, Watt, David A. John Wiley & Sons Ltd.
3. The Art of Programming through Flowcharts & Algorithms, 2E, A. B. Chaudhuri, Firewall Media

First Semester			
Subject Name: MS Office and PPL Lab			
Subject Code	BSCDSP 104	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours:	60	Credits	2

LIST OF PRACTICAL PROGRAMS

PART A:

1. Design a visiting card for Managing Director of a company as per the following specification.
   - Size of visiting card is 3½×2
   - Name of the company with big font
   - Phone number, Fax number and E-mail address with appropriate symbols.
   - Office and Residence address separated by a line.

2. Create a table with following columns and display the result in separate cells for the following
   - Emp Name, Basic pay, DA, HRA, Total salary.
   - Sort all the employees in ascending order with the name as the key
   - Calculate the total salary of the employee
   - Calculate the Grand total salary of the employee
   - Finding highest salary and
   - Find lowest salary

3. Prepare an advertisement to a company requiring software professional with the following
   - Attractive page border
   - Design the name of the company using WordArt
   - Use at least one clipart.
   - Give details of the company (use bullets etc.)
   - Give details of the Vacancies in each category of employee's (Business manager, Software engineers, System administrators, Programmers, Data entry operators) qualification required.

4. Create two pages of curriculum vitae of a graduate with the following specifications
   - Table to show qualifications with proper headings
   - Appropriate left and right margins
   - Format ½ page using two-column approach about yourself
   - Name on each page at the top right side
   - Page no. in the footer on the right side.

5. Create a letter as the main document and create 10 records for the 10 persons use mail merge to create letter for selected persons among 10.

6. Create a excel spreads sheet to calculate the net pay of the employees following the conditions below.

	A	B	C	D	E	F	G	H	I
1	EMP ID	EMP NAME	BASIC	DA	HRA	GPF	INCOME TAX	GROSS SAL	NET SAL
2									

- DA: -56% of the basic pay if Basic pay is greater than 20000 or else 44%.
- HRA: -15% of the Basic pay subject to maximum of Rs. 4000.
- GPF: -10% of the basic pay.
- INCOMETAX: - 10% of basic if Basic pay is greater than 20000.
- Find who is getting highest salary & who is get lowest salary?

7. The ABC Company shows the sales of different product for 5 years. Create BAR Graph, 3D and Pie chart for the following.

A	B	C	D	E	F
SL.NO.	YEAR	PROD1	PROD2	PROD3	PROD4
1	2019	1000	800	90	1000
2	2020	800	90	70	1100
3	2021	1200	190	100	900
4	2022	600	260	58	1400
5	2023	1800	510	80	800

8. Create a suitable examination data excel sheet and find the sum of the marks (total) of each student and respective, class secured by the student.

- Pass: if marks in each subject>=35
- Distinction: if average>=75
- First class: If average>=60 but <75
- Second Class: if average >=50 but less than 60
- Third class: if average>=35 but less than 50
- Fail: if marks in any subject<35

9. Enter the following data into the sheet.

NAME	DEPARTMENT	SALARY
Anusha	Accounts	30000
Ramesh	Marketing	20000
Tejaswi	Engineering	44000
Harika	Accounts	35000
Poornima	Engineering	56000
Vijay	Marketing	22000
Prasad	Accounts	48000
Swetha	Engineering	60000
Rajesh	Marketing	19000

- Extract records for department Accounts and Salary > 10000
- Sort the data by salary with the department using “sort commands”.
- Calculate total salary for each department using Subtotals

10. Create a presentation with slide transitions and animation effects.

PART B:

Design Flowchart and submit to generate pseudo code and run the program for the following problems (Any 12):

1. Read radius to calculate and display area of circle.
2. Find maximum of three number
3. Count no of digits in a number.
4. Reverse a given number.
5. Print odd number from given numbers.
6. Find minimum, maximum and average score of labs.
7. Print even number between 2 to n
8. Determine the give year is leap or not
9. Finding Duplicates in an Array.
10. Defining and calling a function.
11. To perform bubble sort.
12. Print triangle pattern
13. To search an element using linear search.
14. Solve quadratic equation.
15. Reverse a string.

First Semester			
Subject Name: Computer Fundamentals and Programming in C			
Subject Code	BSCDST 105	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours:	60	Credits	4

COURSE OBJECTIVES:

1. To prepare students understand various number systems
2. To acquire the basic knowledge of digital logic levels and application of knowledge to understand digital electronics circuits.
3. To learn the fundamentals of computers.
4. To understand the Basic concepts and logic in program development.
5. To learn the syntax and semantics of the C programming language.
6. To learn the usage of structured programming approaches in solving problems.

COURSE OUTCOMES:

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

1. Have a thorough understanding of the fundamental concepts and techniques used in programming

2. To write algorithms and to draw flowcharts for solving problems.

2. Use the 'C' language constructs in the right way

3. Design, develop and test programs written in 'C'

4. To decompose a problem into functions and to develop modular reusable code.

Unit 1	**Computer Fundamentals and Organization**	**Hours 15**
	Introduction, Evolution of Computers—A Brief History, Classification of Computers, Anatomy of a Computer, Basic Organization of a Computer, Number System: Binary – Decimal – Conversion – Problems. Boolean Algebra: Logical gates - Combinational Circuits. Types of Programming Languages, Translators. Need for Logical Analysis and Thinking, Algorithm – Pseudocode – Flowchart, Problem Formulation Problem Solving.	
Unit 2	**C Programming Basics**	**Hours 12**
	Introduction to "C" Programming – Fundamentals – Features of C, Structure of a C Program, Compilation and Linking Processes. **C Programming Basic Concepts**– Tokens- Identifies, Keywords, Constants, variable. Data Types – Declaration and initialization of variables, typedef, typecasting. Expressions and Type of operators. Precedence and order of Evaluation **Managing Input and Output Operations** – Unformatted, Formatted I/O operations- printf and scanf, escape sequence characters. **Control Statements:** Sequence, Decision Making and Branching – simple if, if-else, nested if – if ladder and elseif ladder, switch- case, Looping Statements – while, for, do-while, jumping statements- break, continue, exit, goto-labels. Solving Simple Scientific	

	and Statistical Problems.	
Unit 3	**Arrays and Strings**	**Hours 13**
	Arrays: Initialization, Declaration, One Dimensional and Two-Dimensional Arrays. Searching: linear and binary searching, Sorting: selection and bubble sorting techniques. Matrix operations- addition and multiplication of two matrices. **String:** Declaration and initialization, String handling functions, character handling functions, Command Line Arguments.	
Unit 4	**Functions and Pointers**	**Hours 10**
	Function: User defined and Library functions, Basics of functions, Definition of Function, Declaration of Function, Types of user defined functions, Pass by Value, Pass by Reference, Return values, Recursion. **Pointers:** Definition, Initialization, Pointers Arithmetic, Pointers and Arrays, Example Problems.	
Unit 5	**Structures and Unions**	**Hours 10**
	Introduction, Need for Structure Data Type, Structure Definition, Structure Declaration, Structure within a Structure, Union, Programs Using Structures and Unions, Storage Classes, Pre-processor Directives.	

TEXT BOOKS:

1. E. Balaguruswamy: Programming in ANSI C (TMH)
2. Computer Fundamentals and Programming in C, by Anjay Mittal & Anita-Goel,Pearson

REFERENCE BOOKS:

1. P. K. Sinha & Priti Sinha: Computer Fundamentals (BPB)
2. Kamthane: Programming with ANSI and TURBO C (Pearson Education)
3. V. Rajaraman: Programming in C (PHI – EEE)
4. S. Byron Gottfried: Programming with C (TMH)
5. Yashwant Kanitkar: Let us C
6. P.B. Kottur: Programming in C (Sapna Book House)

First Semester			
Subject Name: Programming in C Lab			
Subject Code	BSCDSP 106	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours:	60	Credits	2

LIST OF PRACTICAL PROGRAMS

INSTRUCTIONS:

- Every student should identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.
- The following activities be carried out/ discussed in the lab during the initial period of the semester.
- Familiarization of Basic Software – Operating System, DOS Commands Word Processors, Internet Browsers, Integrated Development Environment (IDE) with Examples.
- Type Program Code, Debug and Compile basic programs covering C Programming fundamentals discussed during theory classes.
- Do's and Don'ts, Safety Guidelines in Computer Lab.

PART A:

1. Convert the temperature given in Fahrenheit to Celsius
2. An object undergoes uniformly accelerated motion. The initial velocity (u) of the object and the acceleration (a) are known. Write a C program to find the velocity (v) of the object after time t
3. Write a C program to Find one's and two's complement of a number
4. Write a C program to multiply a given number with 2n, without using a multiplication operator.
5. The value of n will be entered by the user
6. Write a C program to check whether a given number is perfect or not.
7. Write a C program to Print Floyd's triangle
8. Write a C program to check whether a given number is even or odd without /with using modulus operator
9. Write a C program to calculate the roots of a quadratic equation
10. Making use of sine series, evaluate the value of sin(x), where x is in radians
11. Write a C program to Find arithmetic mean, variance and standard deviation of n elements
12. Write a C program to find a given value called Key in a list of N numbers using binary search

PART B:

1. Write a C program that sorts the list in ascending order by using bubble sort.
2. Given two sorted one-dimensional arrays A and B of size m and n, respectively. Write a C program to merge them into a single-sorted array C that contains every element from arrays A and B in ascending order
3. Write a C program to Add and Multiply two matrices of order m × n
4. Write a C program to find the sum of the rows and columns of a matrix.
5. Write a C program to check whether a given square matrix is symmetric or not.
6. A class consists of a number of students whose names are entered in a random order. Write a C program to display the names of all the students that start with a particular character
7. Write a C program to Input a string and count the occurrences of vowels in the particular string in the string
8. Write a C program that illustrates pass by reference and pass by value
9. Write a C program that makes the use of a recursive function to find the factorial of a number.
10. Write a C function that Illustrate the use of built in mathematical function.
11. Write a C program that illustrates the use of structure and union
12. Write a C program that illustrates to Pre-processor Directives

Second Semester			
Subject Name: Probability and Statistics for Data Science			
Subject Code	BSCDST 201	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours:	60	Credits	4

COURSE OBJECTIVES:

1. To provide strong foundation for data science and application area related to it and understand the underlying core concepts and emerging technologies in data science.
2. Understand data analysis techniques for applications handling large data.

COURSE OUTCOMES:

1. Understand the central tendency, correlation and correlation coefficient and also regression.
2. Apply the statistics for testing the significance of the given large and small sample data by using t- test, f- test and chi-square test
3. Understand the fitting of various curves by method of least square.
4. Enable learners to know descriptive statistical concepts
5. Enable study of probability concept required for Computer learners
6. Understand the terminologies of basic probability, two types of random variables and their probability functions
7. Observe and analyze the behavior of various discrete and continuous probability distributions

Unit 1	**Introduction to Statistics and Data Representation**	**Hours 10**
	Introduction to Statistics and its Data types, attribute, variable- Qualitative & Quantitative, discrete and continuous variable. Collection of data, classification and tabulation of data, Presentation of data Diagrammatic and Graphical Representation: Histogram, frequency curve, frequency polygon; Pie Charts; Bar Charts: Pareto Chart, Scatter Plots (Degree of Association); Line Charts;	
Unit 2	**Descriptive Statistics**	**Hours 10**
	Central Tendency; Mean and its Characteristics, Median and its Characteristics, Quartiles and Percentiles, Mode; Measures of skewness and kurtosis. Dispersion: Range, Mean Absolute Deviation, Interquartile Range (IQR); Variance, Standard Deviation and its Characteristics, Coefficient of Variation. Correlation: Measures of Association: Covariance, Correlation, Coefficient of Correlation; Correlation and Causation. methods of measuring correlation	
Unit 3	**Sampling And Testing**	**Hours 15**
	Methods of sampling, Simple random sampling with and without replacement (SRSWR and SRWOR) stratified random sampling, systematic sampling Formation of Hypothesis, Rules of Hypothesis Testing, Test of significance: Tests of significance – z, t, chi-square and F.	

Unit 4	**Probability**	**Hours 10**
	Introduction, random experiments, sample space, events and algebra of events. Definitions of Probability – classical, statistical, and axiomatic. Conditional Probability, laws of addition and multiplication, independent events, Bayes' rule, Bernoulli trials.	
Unit 5	**Random variables**	**Hours 15**
	Random variables: Discrete and Continuous random variables, Probability density function, Probability distribution of random variables, Expected value, Variance. Distributions: Discrete distributions: Uniform, Binomial, Poisson, Bernoulli, Continuous distributions: uniform distributions, exponential, Normal distribution state all the properties and its application	

TEXT BOOKS:

1. Gupta, S.C. and V. K. Kapoor – Mathemathical Statistics, Sultan Chand and sons.
2. D. C. Montgomery and G. C. Runger, Applied Statistics and Probability for Engineers, Wiley.

REFERENCE BOOKS:

1. Goon A.M., Gupta M.K. and Dasgupta B. (2002): Fundamentals of Statistics, Vol. I & II, 8th Edn. The World Press, Kolkata.
2. Hogg, R.V., Tanis, E.A. and Rao J.M. (2009): Probability and Statistical Inference, Seventh Ed, Pearson Education, New Delhi
3. Biswas and Srivastava – A textbook, mathematical Statistics, Ist Edition, Narosa Publishing House, New Delhi.
4. Hogg,R.V. and Craig, A.T: Introduction to Mathematical Statistics, McMillan.
5. S. C. Gupta – Fundamentals of Statistics, Himalaya Publishing House.
6. J. L. Devore, Probability and Statistics for Engineering and the Sciences, Cengage Learning.

Second Semester			
Subject Name: Probability and Statistics for Data Science Lab			
Subject Code	BSCDSP 202	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours:	60	Credits	2

LIST OF PRACTICAL PROGRAMS

PART A:

Designing laboratory programs for statistics and probability using MS Excel involves creating practical exercises that allow students to explore concepts interactively.

1. Descriptive Statistics Lab

Objective: To compute and interpret descriptive statistics using Excel.

Activities:

[1] Dataset Import: Students will import a real-world dataset into Excel (e.g., sales data, customer demographics, etc.).

[2] Summary Statistics: Calculate mean, median, mode, variance, standard deviation, and range using Excel functions AVERAGE (), MEDIAN (), MODE (), VAR.P(), STDEV.P().

[3] Data Visualization: Create visualizations such as histograms, box plots, and scatter plots using Excel's Chart Tools to summarize and explore the data distribution.

[4] Pivot Tables: Use PivotTables to summarize data, group by categories, and calculate aggregations such as counts, averages, and sums.

[5] Data Visualization: Create histograms, box plots, and scatter plots using the Excel Chart Tools.

2. Probability Distributions Lab:

Objective: To explore and understand different probability distributions using Excel.

Activities:

[6] Uniform Distribution: Have students generate random numbers using 'RAND()' to simulate a uniform distribution and plot a histogram.

[7] Binomial Distribution: Use the 'BINOM.DIST()' function to calculate probabilities and create a binomial distribution chart.

[8] Normal Distribution: Instruct students to use the 'NORM.DIST()' function to compute probabilities and create a normal distribution curve.

[9] Poisson Distribution: Explore the 'POISSON.DIST()' function to calculate probabilities and plot the Poisson distribution.

PART B:

3. Hypothesis Testing Lab:

Objective: To perform hypothesis testing using Excel tools.

Activities:

[10] t-Test: Provide two datasets and ask students to perform an independent sample t-test using the 'T.TEST()' function.

[11] Chi-Square Test: Have students create contingency tables and conduct a chi-square test using 'CHISQ.TEST()'.

[12] ANOVA: conduct a one-way ANOVA using the 'Data Analysis Toolpak' in Excel.

4. Regression Analysis Lab

Objective: To understand and apply linear regression using Excel.

Activities:

[13] Scatter Plot and Trendline: Plot data points on a scatter plot and add a trendline to observe the relationship.

[14] Linear Regression: Use the 'LINEST()' function or 'Data Analysis Toolpak' to perform linear regression and interpret the output, including the slope, intercept, and R-squared value.

[15] Multiple Regression: Extend the analysis by introducing multiple variables and using the 'Data Analysis Toolpak' to perform multiple regression.

5. Correlation Analysis Lab

Objective: To analyze the correlation between variables using Excel.

Activities:

[16] Correlation Coefficient: Provide a dataset and instruct students to calculate the Pearson correlation coefficient using the 'CORREL()' function.

[17] Correlation Matrix: Guide students in creating a correlation matrix for multiple variables using the 'Data Analysis Toolpak'.

[18] Interpreting Correlation: Discuss the meaning of positive, negative, and no correlation based on the results obtained.

Implementation Tips:

Instructions: Provide clear step-by-step instructions for each activity.

Templates: Create Excel templates with pre-loaded data and functions to guide students.

Assessment: Include questions that require interpretation of the results obtained from Excel.

Group Work: Encourage collaborative work to enhance learning and problem-solving skills.

Second Semester			
Subject Name: Data Structure Using C			
Subject Code	BSCDST 203	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours:	60	Credits	4

COURSE OBJECTIVES:

1. To explore and understand the concepts of Data Structures and its significance in programming.
2. To teach efficient storage mechanisms of data for an easy access
3. Provide and holistic approach to design, use and implement abstract data types.
4. Understand the commonly used data structures and various forms of its implementation for different applications using C
5. To improve the logical ability.

COURSE OUTCOMES:

1. Learn about Data structures, its types and significance in computing.
2. Student will be able to handle operations like searching, insertion, deletion, traversing mechanism etc. on various data structures.
3. Explore about Abstract Data types and its implementation
4. Ability to program various applications using different data structure in C

	Introduction to Data Structure:	**Hours 10**
Unit 1	**Data structure** – Definition, Abstract data type, Classification of Data Structures, Operations on Data Structures-primitive and non-primitive. **String Processing**: Definition, Storing Stings, String as ADT, String operations – string length, copy, compare, pattern matching using user defined function and pointer. **Memory allocation**: Definition, types-static and dynamic, difference between static and dynamic. Memory management functions- malloc, calloc, realloc and free.	
Unit 2	**Arrays**	**Hours 10**
	Arrays: Definition, Linear arrays, memory representation of Linear Array, Accessing the Elements of an Array, Calculating the Address of Array Elements. Array operations - Inserting, deleting and merging. **Sorting**: Bubble sort, Insertion sort, Selection sort. **Searching**: Linear Search and Binary search. **Multidimensional array**- Declaration and memory representation, Sparse matrices.	
Unit 3	**Linked Lists**	**Hours 15**
	Linked list: Definition, Types. Memory representation of Singly linked list, operations- create, display, insert and delete. Doubly liked list- definition and memory representation. Circular linked list-definition and memory representation. Advantages	

	and disadvantages of SLL, DLL and CLL. Garbage collection.	
Unit 4	**Stacks and Queues**	**Hours 15**
	Stacks: Definition, Array representation of stacks, Linked representation of stacks, Operations on stack. Expressions: Polish Notation, Application of Stacks. Algorithm – Convert infix to postfix expression and evaluate postfix expression. **Queues** – Definition, Array representation of queue, Linked list representation of queues. Types of queues: Simple queue, Circular queue, Double ended queue, Priority queue, Operations on Queues, Applications of queues.	
Unit 5	**Graphs and Trees**	**Hours 10**
	Graphs: Graph theory terminology: pendent vertex, isolated vertex, degree of vertex, path, walk, self-loop, cycle, Sequential representation of Graphs: Adjacency matrix. **Tree** – Definitions, Types of trees, memory representation of binary tree, tree traversing techniques, Binary Search Trees- definition and construction of binary search tree. AVL Tree- definition. Threaded Binary tree – Definition.	

TEXT BOOK:

1. Data Structure using C by Reema Thereja, Oxford University Press.

REFERENCE BOOKS:

1. Data Structure by S. Lipschutz - (Schaum Series)
2. Introduction to Data Structure in C by: A.N.Kamthane; Pearson Education
3. Alfred V Aho, John E Hopcroft and Jeffery D Ullman, “Data Structures and Algorithms”, Pearson Education.

Second Semester			
Subject Name: Data Structure Using C Lab			
Subject Code	BSCDSP 204	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours:	60	Credits	2

LIST OF PRACTICAL PROGRAMS

PART A:

1. Develop a Program in C for the operations on Strings like finding the string of length, copying two strings, comparing of two string and pattern matching & replacing. Support the program with functions for each of the above operations. Don't use Built-in functions
2. Write a C program to read name and roll number of n number of students from user and store them in a file.
3. Write a C Program to implement dynamic array, find smallest and largest element of the array.
4. Write a C Program read, display and to find the trace of a square matrix
5. Write a C Program to read, display and add two m x n matrices using functions
6. Write a C Program to read, display and multiply two m x n matrices using functions
7. Write a C Program to read the names of cities and arrange them alphabetically.
8. Write a C Program to search an element using linear search technique.
9. Write a C Program to sort the given list using selection sort technique.
10. Write a program to implement merge sort.

PART B:

1. Program to implement linear linked list to perform insert and delete operations on it.
2. Write a C Program to implement Stack and its different operations.
3. Write a C Program to convert an infix expression to postfix.
4. Write a C Program to evaluate a postfix infix expression.
5. Write a C Program to implement simple queue and its different operations.
6. Write a program to implement circular queue using array.
7. Program to create and display different traversal of a binary tree.
8. Write a program to implement BFS.
9. Write a program to implement DFS.
10. Write a program to implement AVL Tree

Second Semester			
Subject Name: Python Programming			
Subject Code	BSCDST 205	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours:	60	Credits	4

COURSE OBJECTIVES:

1. To introduce programming concepts using python
2. To use python programming to solve problems of different domains

COURSE OUTCOMES:

After completing this course satisfactorily, a student will be able to:

1. Explain the basic concepts of Python Programming.
2. Demonstrate proficiency in the handling of loops and creation of functions.
3. Identify the methods to create and manipulate lists, tuples and dictionaries.
4. Discover the commonly used operations involving file handling.
5. Interpret the concepts of Object-Oriented Programming as used in Python.
6. Develop the emerging applications of relevant fields using Python.

Unit 1	**Python Basics**	**Hours 10**
	Introduction to Features and Applications of Python; Python Versions; Installation of Python; Python Command Line mode and Python IDEs; Simple Python Program. **Python Basics:** Identifiers; Keywords; Statements and Expressions; Variables; Operators; Precedence and Association; Data Types; Indentation; Comments; Built-in Functions- Console Input and Console Output, Type Conversions; Python Libraries; Importing Libraries with Examples. Python Control Flow: Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range () and exit () functions.	
Unit 2	**Functions and Strings**	**Hours 10**
	Python Functions: Types of Functions; Function Definition- Syntax, Function Calling, Passing Parameters/arguments, the return statement;Default Parameters; Command line Arguments; Key Word Arguments; Recursive Functions; Scope and Lifetime of Variables in Functions. **Strings:** Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Met **Exception Handling:** Types of Errors; Exceptions; Exception Handling usingtry, except and finally.	
Unit 3	**Python Data Structure**	**Hours 13**
	Lists: Creating Lists; Operations on Lists; Built-in Functions on Lists; Implementation of Stacks and Queues using Lists; Nested Lists. **Dictionaries:** Creating Dictionaries; Operations on Dictionaries; Built-in Functions on Dictionaries; Dictionary Methods; Populating and Traversing Dictionaries. **Tuples and Sets:** Creating Tuples; Operations on Tuples; Built-in Functions	

	on Tuples; Tuple Methods; Creating Sets; Operations on Sets; Built-in Functions on Sets; Set Methods.	
Unit 4	**Object Oriented Programming System**	**Hours 12**
	Object Oriented Programming: Classes and Objects; Creating Classes and Objects; Constructor Method; Classes with Multiple Objects; Objects as Arguments; Objects as Return Values; Inheritance- Single and Multiple Inheritance, Multilevel and Multipath Inheritance; Encapsulation- Definition, Private Instance Variables; Polymorphism- Definition, Operator Overloading. **File Handling:** File Types; Operations on Files– Create, Open, Read, Write, Close Files; File Names and Paths; Format Operator	
Unit 5	**SQLite, GUI and Visualization**	**Hours 15**
	Python SQLite: The SQLite3 module; SQLite Methods- connect, cursor, execute, close; Connect to Database; Create Table; Operations on Tables- Insert, Select, Update. Delete and Drop Records. **GU Interface:** The tkinter Module; Window and Widgets; Layout Management- pack, grid and place. **Data Analysis:** NumPy- Introduction to NumPy, Array Creation using NumPy, Operations on Arrays; Pandas- Introduction to Pandas, Series and DataFrames, Creating DataFrames from Excel Sheet and .csv file, Dictionary and Tuples. Operations on DataFrames. **Data Visualization:** Introduction to Data Visualisation; Matplotlib Library; Different Types of Charts using Pyplot- Line chart, Bar chart and Histogram and Pie chart.	

TEXT BOOKS:

1. Introduction to Python Programming, Gowrishankar S et al., CRC Press, 2019.
2. Python Data Analytics: Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language, Fabio Nelli, Apress®, 2015

REFERENCE BOOKS:

1. Think Python How to Think Like a Computer Scientist, Allen Downey et al., 2nd Edition,
2. Green Tea Press. Freely available online @ https://www.greenteapress.com/thinkpython/thinkCSpy.pdf, 2015.
3. Advance Core Python Programming, MeenuKohli, BPB Publications, 2021.
4. Core PYTHON Applications Programming, Wesley J. Chun, 3rd Edition, Prentice Hall, 2012.
5. Automate the Boring Stuff, Al Sweigart, No Starch Press, Inc, 2015.
6. Data Structures and Program Design Using Python, D Malhotra et al., Mercury Learning and
7. Information LLC, 2021.
8. http://www.ibiblio.org/g2swap/byteofpython/read/
9. https://docs.python.org/3/tutorial/index.html

Second Semester			
Subject Name: Python Programming Lab			
Subject Code	BSCDSP 206	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours:	60	Credits	2

LIST OF PRACTICAL PROGRAMS

PART A:

1. Write Python programs to demonstrate the following:
    a) input( )
    b) print( )
    c) 'sep' attribute
    d) 'end' attribute
    e) replacement Operator ({ })
2. Check if a number belongs to the Fibonacci Sequence
3. Solve Quadratic Equations
4. Find the sum of n natural numbers
5. Display Multiplication Tables.
6. Check if a given number is a Prime Number or not
7. Implement a sequential search
8. Create a calculator program
9. Explore string functions
10. Implement Selection Sort
11. Implement Stack
12. Read and write into a file

PART B:

1. Demonstrate usage of basic regular expression
2. Demonstrate use of advanced regular expressions for data validation.
3. Demonstrate use of List
4. Demonstrate use of Dictionaries
5. Create SQLite Database and Perform CRUD Operations on Tables
6. Create a GUI using Tkinter module
7. Demonstrate Exceptions in Python
8. Drawing Line chart and Bar chart using Matplotlib
9. Drawing Histogram and Pie chart using Matplotlib
10. Create Array using NumPy and Perform Operations on Array.
11. Create DataFrame from Excel sheet using Pandas and Perform Operations on DataFrames

Question Paper Pattern for Semester End Examination (SEE)
(Common for I and II Semester)
SUBJECT NAME

Time: 3 Hrs **Max. Marks: 80**

Instruction to Candidate: Answer all the Sections

SECTION A

I. Answer any ten of the following questions (10X2 = 20)

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.

SECTION B

II. Answer any five of the following questions (5X5 = 25)

13.
14.
15.
16.
17.
18.
19.

SECTION C

III. Answer any five of the following questions (5X7 = 35)

20.
21.
22.
23.
24.
25.
26.