



SCHEME AND SYLLABUS

For the Course

BACHELOR OF COMPUTER APPLICATIONS

I, II, III & IV Semester - State Education Policy 2024

V & VI - National Education Policy 2024

Academic Year 2025- 2026 and onwards



I Semester SEPS

	SUBJECTS	PAPER TITLE	PAPER CODE	WORKING HOURS (L+T+P)	DURATION OF EXAMS (hrs)	MARKS				TOTAL MARKS	CREDITS
						IA		ESE			
						Max	Min	Max	Min		
PART 1: Languages	Language 1: Kannada/Hindi/Sanskrit	KANNADA 1/ HINDI 1 /SANSKRIT 1		(3+1+0) 4	3	20	8	80	32	100	3
	Language 2: General English	GENERAL ENGLISH 1		(3+1+0) 4	3	20	8	80	32	100	3
PART 2: Core papers	Discrete Structure		BCADS11T	3	3	20	8	80	32	100	3
	Programming in C		BCAPC12T	(3+0+2) 4	3	20	8	80	32	100	3
	Computer Architecture		BCACA13T	(3+0+2) 4	3	20	8	80	32	100	3
	Programming in C Practical		BCAPC12P	3	3	10	4	40	16	50	2
	Computer Architecture Practical		BCACA13P	3	3	10	4	40	16	50	2
	Office Automation Tools Practical		BCAOA14T	4	3	10	4	40	16	50	2
PART 3: Compulsory courses	Computer Applications		CA11	(1+0+2) 3	1.5	10	4	40	16	50	2
PART 4: VAC	Indian Knowledge System		IKS11T	2	1.5	10	4	40	16	50	2
Total Credits											25



II Semester SEPS

	SUBJECTS	PAPER TITLE	PAPER CODE	WORKING HOURS (L+T+P)	DURATION OF EXAMS (hrs)	MARKS				CREDITS	
						IA		ESE			TOTAL MARKS
						Max	Min	Max	Min		
PART 1: Languages	Language 1: Kannada/Hindi/Sanskrit	KANNADA II/ HINDI II /SANSKRIT II		(3+1+0) 4	3	20	8	80	32	100	3
	Language 2: General English	GENERAL ENGLISH II		(3+1+0) 4	3	20	8	80	32	100	3
PART 2: Core papers	Data Structure		BCADS21T	(3+0+2) 4	3	20	8	80	32	100	3
	Object Oriented Programming Using Java		BCA00J22T	(3+0+2) 4	3	20	8	80	32	100	3
	Operating Systems		BCAOS23T	4	3	20	8	80	32	100	3
	Data Structure Practical		BCADS21P	3	3	10	4	40	16	50	2
	Object Oriented Programming Using Java Practical		BCA00J22P	3	3	10	4	40	16	50	2
	LINUX and Shell Programming Practical		BCAOS23P	3	3	10	4	40	16	50	2
PART 3: Compulsory courses	Environmental studies		EVS21	(1+0+2) 3	1.5	10	4	40	16	50	2
PART 4: VAC	EC/CC									50	2
Total Credits											25

III Semester SEPS



	SUBJECTS	PAPER TITLE	PAPER CODE	WORKING HOURS (L+T+P)	DURATION OF EXAMS (hrs)	MARKS				TOTAL MARKS	CREDITS
						IA		ESE			
						Max	Min	Max	Min		
PART 1: Languages	Language 1: Kannada/Hindi/Sanskrit	KANNADA A III/ HINDI III /SANSKRIT III		(3+1+0) 4	3	20	8	80	32	100	3
	Language 2: General English	GENERAL ENGLISH III		(3+1+0) 4	3	20	8	80	32	100	3
PART 2: Core papers	Database Management System		BCADB31T	(3+0+2) 4	3	20	8	80	32	100	3
	Artificial Intelligence		BCAAI32T	(3+0+2) 4	3	20	8	80	32	100	3
	Probability and Statistics		BCAPS33T	(3+0+2) 4	3	20	8	80	32	100	4
	Database Management System Lab		BCADB31P	3	3	10	4	40	16	50	2
	Artificial Intelligence Lab using Python		BCAAI32P	3	3	10	4	40	16	50	2
	Elective: I Feature Engineering (AI & ML) Basics of Web Programming-I (Full Stack Development)		BCAE341T BCAE342T	2	1.5	10	4	40	16	50	2
PART 3: Compulsory courses	Communication skills and Translation		CT31T	(1+0+2) 3	1.5	10	4	40	16	50	2
PART 4: VAC	Human Values		HV31T			20	50				2
Total Credits											26

IV Semester SEPS



	SUBJECTS	PAPER TITLE	PAPER CODE	WORKING HOURS (L+T+P)	DURATION OF EXAMS (hrs)	MARKS				CREDITS	
						IA		ESE			TOTAL MARKS
						Max	Min	Max	Min		
PART 1: Languages	Language 1: Kannada/Hindi/Sanskrit	KANNADA A IV/ HINDI IV /SANSKRIT IV		(3+1+0) 4	3	20	8	80	32	100	3
	Language 2: General English	GENERAL ENGLISH IV		(3+1+0) 4	3	20	8	80	32	100	3
PART 2: Core papers	Computer Networks		BCACN41 T	(3+0+2) 4	3	20	8	80	32	100	3
	Design and Analysis of Algorithms		BCADA42 T	(3+0+2) 4	3	20	8	80	32	100	3
	Software Engineering		BCASE43 T	(3+0+2) 4	3	20	8	80	32	100	4
	Computer Networks Practical		BCACN41 P	3	3	10	4	40	16	50	2
	Design and Analysis of Algorithms practical		BCADA42 P	3	3	10	4	40	16	50	2
	Elective: II Feature Engineering (AI & ML) Basics of Web Programming-II (Full Stack Development)		BCAE441 T								
			BCAE452 T	2	1.5	10	4	40	16	50	2
PART 3: Compulsory courses	Indian Constitution		IC41	(1+0+2) 3	1.5	10	4	40	16	50	2
PART 4: VAC	Computer Assembly and Repair Practical		CAR-P	3	1.5	25	10	25	10	50	2
Total Credits										26	

V Semester NEPS



	SUBJECTS	PAPER TITLE	PAPER CODE	WORKING HOURS (L+T+P)	DURATION OF EXAMS (hrs)	MARKS				TOTAL MARKS	CREDITS
						IA		ESE			
						Max	Min	Max	Min		
PART 1: Core papers	Artificial Intelligence		DSAI51	(3+0+2) 4	2.5	40	16	60	24	100	4
	Data Analytics		DSDA52	(3+0+2) 4	2.5	40	16	60	24	100	4
	Web Programming		DSWP53	(3+0+2) 4	2.5	40	16	60	24	100	4
	Data analytics Practical		DSDA52P	3	3	25	10	25	10	50	2
	Web Programming Practical		DSWP53P	3	3	25	10	25	10	50	2
	Data Mining		DSEDM54	3	2.5	40	16	60	24	100	3
	Cyber Crimes, Cyber Laws and Intellectual Property Rights		SEC51	3	1.5	25	10	25	10	50	2
	Quantitative Techniques		VOCQT55	3	2.5	40	16	60	24	100	3
PART2: Compulsory courses	Employability skills		EMP51	(1+0+2) 3	2.5	40	16	60	24	100	3
Total Credits											27



VI Semester NEPS

	SUBJECTS	PAPER TITLE	PAPER CODE	WORKING HOURS (L+T+P)	DURATION OF EXAMS (hrs)	MARKS				CREDITS	
						IA		ESE			TOTAL MARKS
						Max	Min	Max	Min		
PART 1: Core papers	Machine Learning		DSML61	(3+0+2) 4	2.5	40	16	80	24	100	4
	Mobile Application Development		DSMAD62	(3+0+2) 4	2.5	40	16	80	24	100	4
	Machine Learning Practical		DSML61P	3	3	25	10	25	10	50	2
	Mobile Application Development Practical		DSMAD62P	3	3	25	10	25	10	50	2
	Software Testing		DSEST63	3	2.5	40	16	80	24	100	3
	Electronic Content Design		VOCECD64	3	2.5	40	16	80	24	100	3
PART 2: Compulsory courses	Internship		INT61	90		40	16	60	24	100	3
	Project work		PWBCA61	6		40	16	60	24	100	4
Total Credits										25	

**Program Outcomes (POs)**

PO No.	Title	Description
PO1	Computational Knowledge	Acquire fundamental knowledge in computing, algorithms, and mathematical foundations relevant to computer applications.
PO2	Problem Solving Skills	Identify, formulate, and solve complex computing problems using logical and analytical thinking.
PO3	Software Design and Development	Design, develop, and evaluate software solutions for real-time applications using modern tools and technologies.
PO4	Modern Tool Usage	Use contemporary computing tools and techniques for software development, testing, deployment, and maintenance.
PO5	Communication and Teamwork	Demonstrate effective communication, presentation skills, and the ability to function efficiently in teams.
PO6	Ethical and Professional Practice	Apply ethical principles and adhere to professional standards in the use and development of computing technologies.
PO7	Lifelong Learning	Continuously upgrade skills and adapt to technological advancements in computing and IT.
PO8	Employability and Entrepreneurship	Develop entrepreneurial mindset and employability skills by identifying business opportunities, creating innovative computing solutions, and contributing effectively to start-ups or organizational growth.

**Program Specific Outcomes (PSOs)**

PSO No.	Title	Description
PSO1	Application of Computing Knowledge	Apply fundamental concepts of computer science, programming, and mathematics to design, develop, and test efficient software applications.
PSO2	Software Development Skills	Demonstrate proficiency in modern programming languages, databases, web technologies, and software development tools to build real-world applications.
PSO3	Problem Solving and Analytical Thinking	Analyze problems, identify computing requirements, and develop logical solutions using structured and object-oriented approaches.
PSO4	Professional and Ethical Practices	Apply ethical principles, teamwork, communication, and project management skills in software development and IT-enabled services.
PSO5	Lifelong Learning and Emerging Technologies	Adapt to emerging technologies, platforms, and practices in the IT industry through continuous learning and research-oriented skills.



I SEMESTER

Course Code: 24BCA11	Course Title: Discrete Structure
Course Credits: 03	Hours/Week: 04
Total Contact Hours: 45	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 3Hours

Course Outcome	
CO1	Apply Set theory, Relations, Functions and Mathematical Induction for solving problems
CO2	Getting familiar with counting techniques and Mathematical Logic, and apply the same to address the real-life problems
CO3	Learn different operations of a Matrix and apply them to solve real life problems.
CO4	Apply graph theory concepts to solve the problems of the real world.

UNIT – I	11 Hours
Set Theory: Fundamentals of Set theory, Set Operations, Laws of Set Theory, Counting and Venn Diagrams, Cartesian Product, Relations, Types of Relations, Functions, Types of Functions, Function Composition, Inverse Functions. Mathematical Induction.	
UNIT – II	11 Hours
Logic and Counting: Fundamentals of Logic, Propositional Logic, Logical Connectives and Truth Tables, Logic Equivalence, Tautology and Contradiction. Basics of counting, Counting Principles, Pigeonhole Principle, Permutation, Combinations.	
UNIT - III	11 Hours
Matrices: Basics of Matrix, Types of Matrices, Operations on Matrices, Inverse of a matrix, Solution for system of linear equations, Determinant, Properties of Determinant, Cramer's Rule, Introduction to Eigen Values and Eigen Vectors.	
UNIT - IV	12 Hours
Graph Theory: Graphs: Introduction, Representing Graphs, Operations on graphs, Directed Graphs Graph Isomorphism, Paths, Cycles, Euler Graph, Hamilton Graph, Planar Graphs. Trees: Introduction, Applications of Trees, Spanning Trees, Minimum Spanning Trees, Prim's and Kruskal's Algorithms.	



Text Book	
1	Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education.
2	Richard Bronson, Schaum’s Outline of Matrix Operations, McGraw-Hill publications, 2nd Edition,

Reference Books	
1	Gregory Hartman, Fundamentals of Matrix Algebra, Third Edition
2	Gary Haggard, John Schlipf, Discrete Mathematics for Computer Science, Thomson Books

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	-	-	-	-	1	-	3	-	2	-	-
CO2	3	3	-	-	-	-	1	1	3	-	3	-	-
CO3	2	2	-	1	-	-	-	1	2	-	2	-	-
CO4	3	3	1	1	-	-	1	1	3	-	3	-	-



Course Code: 24BCA12	Course Title: Programming in C
Course Credits: 03	Hours/Week: 04
Total Contact Hours: 55	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 3Hours

Course Outcomes

CO1: Understand the basics of computers and how to plan programs using algorithms and flowcharts.

CO2: Write simple C programs using data types, variables, and input/output functions.

CO3: Use operators and control statements (if, loops, switch) to solve problems in C.

CO4: Work with arrays, strings, and pointers for handling data.

CO5: Create modular programs using functions, structures, and unions.



UNIT –1	12 Hours
Fundamentals of Computers: Introduction to Computers - Computer Definition, Characteristics of Computers, Evolution and History of Computers, Types of Computers, Basic Organization of a Digital Computer; Computer Languages - Machine Level, Assembly Level & High Level Languages, Translator Programs – Assembler, Interpreter and Compiler; Planning a Computer Program - Algorithm, Flowchart and Pseudo code with Examples.	
UNIT-II	11 Hours
<p>Introduction to C Programming: Over View of C; History and Features of C; Structure of a C Program with Examples; Creating and Executing a C Program; Compilation process in C.</p> <p>C Programming Basic Concepts: C Character Set; C tokens - keywords, identifiers, constants, and variables; Data types; Declaration & initialization of variables; Symbolic constants.</p> <p>Input and output with C: Formatted I/O functions - <i>printf</i> and <i>scanf</i>, control strings and escape sequences, output specifications with <i>printf</i> functions; Unformatted I/O functions to read and display single character and a string - <i>getchar</i>, <i>putchar</i>, <i>gets</i> and <i>puts</i> functions.</p>	
UNIT - III	11 Hours
<p>C Operators & Expressions: Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators; Bitwise operators; Conditional operator; Special operators; Operator Precedence and Associativity; Evaluation of arithmetic expressions; Type conversion.</p> <p>Control Structures: Decision making Statements - <i>Simple if</i>, <i>if_else</i>, <i>nested if_else</i>, <i>else_if ladder</i>, <i>Switch-case</i>, <i>goto</i>, <i>break</i> & <i>continue</i> statements; Looping Statements - Entry controlled and Exit controlled statements, <i>while</i>, <i>do-while</i>, <i>for</i> loops, Nested loops.</p>	
UNIT - IV	11 Hours
<p>Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays - Declaration, Initialization and Memory representation.</p> <p>Strings: Declaring & Initializing string variables; String handling functions - <i>strlen</i>, <i>strcmp</i>, <i>strcpy</i> and <i>strcat</i>; Character handling functions - <i>tolower</i>, <i>toupper</i>, <i>isalpha</i>, <i>isnumeric</i> etc.</p> <p>Pointers in C: Understanding pointers - Declaring and initializing pointers, accessing address and value of variables using pointers; Pointers and Arrays; Advantages and disadvantages of using pointers.</p>	
UNIT - V	10 Hours
<p>User Defined Functions: Need for user defined functions; Format of C user defined functions; Components of user defined functions - return type, name, parameter list, function body, return statement and function call; Categories of user defined functions - With and without parameters and return type.</p> <p>User defined data types: Structures - Structure Definition, Advantages of Structure, declaring structure variables, accessing structure members, Structure members initialization, comparing structure variables, Array of Structures; Unions - Union definition; difference between Structures and Unions.</p>	
Text Book	
1	R. G. Dromey, “How to Solve it by Computer”, Person Education India, 2008.



2	Brain M. Kernighan and Dennis M. Ritchie, “ The C Programming Language”, 2 nd edition, Princeton Hall Software Series, 2012.
3	Thomas H Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, “Introduction to Algorithms”, 3 rd Edition, The MIT Press Cambridge, Massachusetts London, England, 2008.

Reference Books

1	E. Balaguruswamy, “Programming In ANSI C”, 4th edition, TMH Publications, 2007
2	Greg Perry and Dean Miller, “C programming Absolute Beginner’s Guide”, 3rd edition, Pearson Education, Inc, 2014.
3	Donald E. Knuth, The Art of Computer Programming”, Volume 2: Seminumerical Algorithms, 3rd Edition, Addison Wesley Longman, 1998.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	-	-	1	-	-	-	3	1	2	-	-
CO2	2	3	2	1	-	-	1	-	3	2	2	-	-
CO3	2	2	3	1	-	-	-	1	2	2	3	-	-
CO4	1	2	2	-	-	-	-	-	2	3	3	-	-
CO5	1	2	3	2	-	-	-	-	2	3	3	-	-



Course Code: 24BCA12	Course Title: Computer Architecture
Course Credits: 03	Hours/Week: 04
Total Contact Hours: 44	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 3 Hours

Course Outcome	
CO1	Understand number systems, binary arithmetic, codes, Boolean algebra, and digital logic simplification techniques.
CO2	Analyze and design combinational & sequential logic circuits such as adders, flip-flops, multiplexers, registers, and counters.
CO3	Demonstrate knowledge of basic computer organization, instruction cycle, CPU design, addressing modes, and compare RISC vs CISC architectures.
CO4	Develop and execute assembly language programs using 8085 microprocessor instruction set, addressing modes, and programming model.

UNIT-1	12Hours
Number Systems: Decimal, Binary, Hexadecimal, Octal Number System Conversions, Binary Arithmetic, Complements- r's complement, (r-1)'s complement, Addition and subtraction of BCD, Octal Arithmetic, Hexadecimal Arithmetic, Binary Codes, Decimal Codes, Error detecting and correcting codes, ASCII, EBCDIC, UNICODE, Digital Logic Circuits: Digital Computers, Logic Gates, Universal Gates, Boolean algebra, Map Simplification.	
UNIT-2	11Hours
Combinational Circuits- Half Adder and Full Adder, Flip-Flops- SR Flip- Flop, D Flip-Flop, J-K Flip-Flop, T Flip-Flop, Sequential Circuits- Flip-Flop input equations, State Table, State Diagram and problems. Digital Components: Integrated Circuits, Decoders-3-to-8-line decoder, NAND gate Decoder, Octal to Binary Encoder, Multiplexers- 4-to-1 line Multiplexer, Registers- 4 bit register with parallel load, Shift Registers- Bidirectional shift register with parallel load, Binary Counters-4-bit synchronous binary counter.	
UNIT-3	11Hours
Basic Computer Organization and Design: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input- Output Interrupt, Complete Computer Description, Design of Basic Computer, Design of Accumulator logic. Central Processing Unit: Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer (RISC), CISC Vs RISC.	
UNIT-4	11Hours



Introduction to 8085 Assembly language programming: Architecture of 8085, Pin Configuration, The 8085-programming model, Instruction classification, Instruction, data formats, and storage. How to write assemble and execute a simple program, overview of 8085 instruction set. Introduction to 8085 Instructions: Instruction classification of 8085 based on word length and functions, Data Transfer operations, Arithmetic operations, Logic Operations, Branch operations, Writing Assembly language programs, Addressing modes of 8085.

Text Book

- 1 M. Morris Mano- “Computer System Architecture”, 3rd Edition Pearson India, 2019.
- 2 Ramesh Gaonkar- “Microprocessor Architecture, Programming and Applications with the 8085”, 5th Edition, Penram International Publishing (India) Private Limited,2007.
- 3 Andrew S. Tanenbaum, Todd Austin –“Structured Computer Organization”, PHI /Pearson 6th Edition,2013

Reference Books

- 1 William Stallings- “Computer Organization and Architecture”, Pearson/PHI, 6th Edition,2007.
- 2 Andrew S. Tanenbaum-“ Structured Computer Organization”, PHI /Pearson 4th Edition,1998.
- 3 M.V. Subramanyam, “Switching Theory and Logic Design”, Laxmi Publications (P) Ltd,2011.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs1-8) and Program Specific Outcome (PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	-	-	-	-	1	-	1	-	-	-	-
CO2	3	3	2	2	-	-	1	-	1	-	-	-	-
CO3	3	2	2	2	-	-	2	1	2	1	-	-	-
CO4	3	3	3	3	1	-	2	2	2	2	-	-	-



Course Code: 24BCA12P	Course Title: Programming in C Practical
Course Credits: 02	Hours/Week: 04
Total Contact Hours: 52	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03 Hours

CO1: Apply fundamental programming concepts to solve simple computational problems. This includes using basic data types, variables, arithmetic operations, and control structures to create functional C programs.

CO2: Implement programs that use arrays, strings, and pointers to efficiently manipulate and manage data. This involves writing code to handle collections of data and understand memory management.

CO3: Design and develop modular programs using functions, structures, and unions to create well-organized and reusable code. This demonstrates an understanding of breaking down complex problems into smaller, manageable components.

CO4: Analyze and debug C programs by identifying and correcting logical and syntax errors. This involves using problem-solving skills to ensure program correctness and efficiency.

Part A:

1. Write a C Program to read radius of a circle and to find area and circumference
2. Write a C Program to read three numbers and find the biggest of three
3. Write a C Program to demonstrate library functions in math.h
4. Write a C Program to check for prime
5. Write a C Program to generate n primes
6. Write a C Program to read a number, find the sum of the digits, reverse the number and check it for palindrome
7. Write a C Program to read numbers from keyboard continuously till the user presses 999 and to find the sum of only positive numbers
8. Write a C Program to read percentage of marks and to display appropriate message (Demonstration of else-if ladder)
9. Write a C Program to find the roots of quadratic equation (demonstration of switch-case statement)
10. Write a C program to read marks scored by n students and find the average of marks (Demonstration of single dimensional array)
11. Write a C Program to remove Duplicate Element in a single dimensional Array
12. Program to perform addition and subtraction of Matrices

**Part B:**

1. Write a C Program to find the length of a string without using built in function
2. Write a C Program to demonstrate string functions.
3. Write a C Program to demonstrate pointers in C
4. Write a C Program to check a number for prime by defining isprime() function
5. Write a C Program to read, display and to find the trace of a square matrix
6. Write a C Program to read, display and add two m x n matrices using functions
7. Write a C Program to read, display and multiply two m x n matrices using functions
8. Write a C Program to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters.
9. Write a C Program to Reverse a String using Pointer
10. Write a C Program to Swap Two Numbers using Pointers
11. Write a C Program to demonstrate student structure to read & display records of n students.
12. Write a C Program to demonstrate the difference between structure & union.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	2	1	1	-	-	1	3	2	2	-	-
CO2	3	2	3	2	1	-	-	1	3	3	2	2	-
CO3	3	2	3	2	2	-	1	2	3	3	2	3	-
CO4	3	3	2	2	1	1	1	2	3	2	3	3	2



Course Code: 24BCA13P	Course Title: Computer Architecture Practical
Course Credits: 02	Hours/Week: 04
Total Contact Hours: 52	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03 Hours

CO1: Understand the 8085 microprocessor architecture, instruction set, and addressing modes.

CO2: Apply 8085 assembly language instructions to perform arithmetic, logical, and bit manipulation operations.

CO3: Analyze and implement assembly programs for solving computational problems like sorting, searching, palindrome, and number system-based tasks.

CO4: Develop and execute modular assembly programs for real-time applications such as code conversions, Fibonacci series, and arithmetic operations on multi-byte data.

Part A

1. Write an 8085 Program to swap two 8-bit numbers.
2. a. Write a Program to find the largest of two numbers
b. Write an 8085 Program to find the smallest of two numbers
3. Write an 8085 Program to find whether an 8-bit number is positive, negative or zero. If positive display EE, if negative display FF, if zero display DD.
4. Write an 8085 Program to check whether 4th bit of a number is zero or one. If 4th bit is 1 display FF, if 4th bit is 0 display DD.
5. Write an 8085 Program to calculate the sum of first ten natural numbers.
6. Write an assembly language program in 8085 microprocessors to find the sum of digits of an 8-bit number.
7. Write an 8085 Program to find the reverse of an 8-bit number
8. Write an 8085 Program to check whether 1-byte number is a palindrome or not. If it is a palindrome display FF otherwise display DD.
9. Write an 8085 Program to check whether a number is ODD or EVEN. If Even no. display DD, if odd no. display FF.
10. Write an 8085 program to count a number of ones in the given 8-bit number.

Part B

11. Write an 8085 program to find Addition & Subtraction of two 8 –bit HEX numbers.
12. Write an 8085 program to find Addition of two 16 –bit numbers.
13. Write an 8085 program to find Subtraction of two 16 –bit numbers.
14. Write an 8085 program for Swapping of two 16-bit numbers.
15. Write an 8085 program to implement 2 out of 5 codes
16. Write an 8085 program to generate Fibonacci series
17. Write an 8085 program to find the first ten terms of odd and even numbers.



18. Write an 8085 program to find 4-Digit BCD addition.
19. Write an 8085 program to find Multiplication of 2-digit BCD numbers.
20. Write an 8085 program to find division of two 8-bit numbers.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	1	1	-	-	1	-	3	2	2	1	1
CO2	3	3	2	2	-	-	1	-	3	3	2	1	1
CO3	3	3	3	2	1	-	2	1	3	3	3	2	2
CO4	3	3	3	3	2	1	2	2	3	3	3	2	3



II SEMSTER

Course Code: 24BCA21	Course Title: Data Structure
Course Credits: 03	Hours/Week: 04
Total Contact Hours: 45	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 3Hours

Course Outcome	
CO1	Understand the fundamentals of data structures, abstract data types, algorithm complexity, and string operations.
CO2	Apply arrays and linked lists to perform insertion, deletion, traversal, searching, and sorting operations.
CO3	Implement stacks, queues, and recursion to solve computational problems and evaluate expressions.
CO4	Analyze and apply advanced data structures such as trees, graphs, and hashing techniques for efficient data storage, retrieval, and problem solving.

UNIT I	11 Hours
Introduction and Overview: Definition, Elementary data organization, Data Structures, data Structures operations, Abstract data types, algorithms complexity, time-space trade off. Preliminaries: Mathematical notations and functions, Algorithmic notations, control structures, Complexity of algorithms, asymptotic notations for complexity of algorithms. Introduction to Strings, Storing String, Character Data Types, String Operations, word processing, Introduction to pattern matching algorithms.	
UNIT II	11 Hours
Arrays: Definition, Linear arrays, arrays as ADT, Representation of Linear Arrays in Memory, Traversing Linear arrays, Inserting and deleting, multi-dimensional arrays, Matrices and Sparse matrices, searching and sorting techniques using array. Linked list: Definition, Representation of Singly Linked List in memory, Traversing a Singly linked list, Searching in a Singly linked list, Memory allocation, Garbage collection, Insertion into a singly linked list, Deletion from a singly linked list; Doubly linked list, Header linked list, Circular linked list.	
UNIT III	11 Hours



Stacks: Definition, Array representation of stacks, Linked representation of stacks, Stack as ADT, Arithmetic Expressions: Polish Notation, Conversion of infix expression to postfix expression, Evaluation of Post fix expression, Application of Stacks, Recursion, Towers of Hanoi, Implementation of recursive procedures by stack. Queues: Definition, Array representation of queue, Linked list representation of queues. Types of queue: Simple queue, Circular queue, Double-ended queue, Priority queue, Operations on Queues, Applications of queues.

UNIT IV	12 Hours
Binary Trees: Definitions, Tree Search, Traversal of Binary Tree, Tree Sort, Building a Binary Search Tree, Height Balance: AVL Trees, Contiguous Representation of Binary Trees: Heaps, Red Black Tree: Insertion and Deletion, External Searching: B-Trees, Applications of Trees. Graphs: Mathematical Back ground, Computer Representation, Graph Traversal. Hashing: Hash Table ADT, understanding Hashing, Components of Hashing, Hash Table, Hash Function, Hashing Techniques, collisions, collision resolution techniques.	

Text Book	
1	Seymour Lipschutz, “Data Structures with C”, Schaum’s outLines, Tata Mc Graw Hill, 2011.
2	Robert Kruse, C.L.Tondo, Bruce Leung, Shashi Mogalla, “Data Structures and Program Design using C”, Pearson Education, 2009

Reference Books	
1	Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, Second Edition, Pearson Education, 2013
2	Forouzan, “A Structured Programming Approach using C”, 2nd Edition, Cengage Learning India, 2008.

**Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs1-8)**

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	1	-	-	-	1	-	2	-	-	-	-
CO2	2	3	2	1	-	-	-	-	3	2	-	-	-
CO3	2	3	2	1	1	-	-	-	2	2	2	-	-
CO4	2	3	3	2	1	-	2	2	3	2	3	-	2



Course Code: 24BCA22	Course Title: Object Oriented Programming Using JAVA
Course Credits: 03	Hours/Week: 04
Total Contact Hours: 45	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 3 Hours

Course Outcome

CO1 - Understand the principles of object-oriented programming and apply Java basics such as data types, operators, control structures, arrays, and classes to build simple programs.

CO2 - Apply the concepts of inheritance, polymorphism, interfaces, and packages to develop modular and reusable Java applications.

CO3 - Develop interactive applications using event handling, graphical user interface (GUI) components, applets, and string manipulation techniques.

CO4 - Implement exception handling, multithreading, collections, and advanced Java features to design efficient, secure, and concurrent applications.

UNIT-1	12 Hours
Introduction: Basics of object-oriented programming, comparison of procedure-oriented and object-oriented programming paradigms; Difference between C and Java Programming languages; Features of Java; Objects and classes in Java, Structure of a Java program; Data Types, variables and operators in java; Control structures- Branching and looping; Methods & Constructors in java; Java Development Kit (JDK); Built-in classes in Java; Math, Character, String, String Buffer and Scanner; Wrapper classes; The abstract, static and final classes; Casting objects; The instance of operator; Usage of this keyword; Arrays in Java.	
UNIT-2	11 Hours
Inheritance: Super and subclasses; visibility modifiers; Types of Inheritance- single, multiple, hierarchical and hybrid inheritance; the interface concept in Java, P o l y m o r p h i s m : Compile time and run time polymorphisms – Method overloading and method overriding. Package: Types of packages; the util, awt and swing packages; Creating and importing user-defined packages. I/O programming: Standard I/O streams in Java; Types of streams – Based on the type of Operations and the type of file.	
UNIT -3	11 Hours



Event handling: Major events in Java; Two Event Handling mechanisms- Event classes and Event Listener Interfaces; Mouse and keyboard events; GUI: Panels; Frames; Layout managers – Flow, border and grid layouts; Buttons; Checkboxes; Radio buttons; Labels; Text fields; Text areas; Combo boxes; Scroll bars; Sliders; Menu, Dialog boxes. Applet programming: Comparison of applets and applications; Applet life cycle; Developing and running applets. String handling: String construction, string length, special string operations, character extraction, string comparison, modifying string and string buffers.

UNIT-4

11 Hours

Exception handling: Types of Java exception – checked and unchecked exceptions; Usage of try-catch-finally blocks. Multithreading: comparison of multithreading and multitasking; Life cycle of a thread; two ways of creating thread – by extending the Thread class and by implementing the Runnable Interface, Thread synchronization. Advanced concepts: Collections in Java; Introduction to JavaBeans and Java security manager, Importance of generic programming in java with examples.

Text Book

- | | |
|---|--|
| 1 | E. Balagurusamy, Programming with JAVA, McGraw Hill, New Delhi, 2007 |
| 2 | Herbert Schildt, Java A Beginner's Guide – Create, Compile, and Run Java Programs Today, Sixth Edition, Oracle Press, 2014 |

Reference Books

- | | |
|---|---|
| 1 | Ken Arnold, James Gosling, "The Java Programming Language, Fourth Edition, Addison Wisely, 2005 |
| 2 | R Herbert Schildt, 'The Complete Reference Java, 7 th Edition, McGraw Hill, 2007 |

**Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)**

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	2	1	-	-	2	-	3	2	2	-	1
CO2	2	3	3	2	-	-	2	2	3	3	2	1	2
CO3	2	3	3	3	1	-	2	2	2	3	3	1	2
CO4	2	3	3	3	1	1	2	3	3	3	3	2	3



Course Code: 24BCA23	Course Title: Operating Systems
Course Credits: 05	Hours/Week: 05
Total Contact Hours: 45	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 3Hours

Course Outcomes

CO1: Explain the structure and functions of operating systems, including process, memory, storage, and file management.

CO2: Apply scheduling algorithms, synchronization techniques, and deadlock handling methods to solve process management problems.

CO3: Analyze memory management strategies, virtual memory concepts, and file system implementations for efficient resource utilization.

CO4: Demonstrate the use of Linux commands for file operations, process control, and disk management in a practical environment.

UNIT –1	11Hours
Introduction: Computer System Organization, Architecture, Structure, Operations, Process Management, Memory Management, Storage Management. Operating System Structures: Services, System Calls, Types, Operating System Structure, System Boot. Processes: Process Concept, Scheduling, Operations, Inter-process Communication. Multithreaded Programming: Multithreading Models	
UNIT-II	12Hours
Process Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization Examples. Process Scheduling: Criteria, Scheduling Algorithms, Multi-Processor Scheduling, Real-time CPU Scheduling. Deadlocks: System model, Characterization, Methods for handling deadlocks, Deadlock Prevention, Avoidance, Detection and Recovery from deadlock.	
UNIT – III	11 Hours
Memory Management Strategies: Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table. Virtual Memory Management: Demand Paging; Copy-on-Write, Page Replacement; Allocation of Frames, File System: File Concept, Access Methods, Directory and Disk Structure, Protection. File-system Implementation: Structure, File- System and Directory Implementation, Allocation Methods, Free Space Management. Mass-Storage Structure:	



Overview, Disk Scheduling, Disk Management.

UNIT – IV

11Hours

Introduction to Linux Programming: Linux system Architecture, Linux Command format, Linux Internal and External Commands, Directory Commands, File related commands, Disk related commands, General Utilities

Text Books

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne; “Operating Systems Concepts”, 9th Edition, 2016 India, Wiley.
2. William Stallings, "Operating Systems and Design Principles", Pearson, 3rd Edition, 2018

Reference Books

D M Dhamdhere; operating systems - A concept Based Approach, 3rd Edition, Tata McGraw - Hill.

1. Sumitabha Das: "I-INIX Concepts and Applications", 4th Edition, Tata McGraw Hill' 2006
2. MG Venkateshmurthy, "uNIX and Shell Programming", Pearson Edition Asia'2005



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	3	-	-	-	-	-	-	3	-	3	-	1
CO2	2	3	2	-	-	-	1	-	3	-	3	-	1
CO3	2	3	1	-	-	-	1	-	2	1	3	-	1
CO4	2	1	-	-	-	-	3	-	2	1	1	-	3



Course Code: 24BCA21P	Course Title: Data Structure Practical
Course Credits: 02	Hours/Week: 04
Total Contact Hours:52	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03 Hours

CO1: Understand the fundamentals of data structures, abstract data types, algorithm complexity, and string operations.
CO2: Apply arrays, linked lists, stacks, queues, and recursion to perform insertion, deletion, traversal, searching, and sorting operations.
CO3: Analyze and implement advanced data structures such as trees, heaps, graphs, and hashing techniques for efficient storage, retrieval, and problem solving.
CO4: Design, test, and evaluate data structure–based solutions using real-time examples, recursion, and expression evaluation for computational applications.

NOTE: For all the programs write the output, flowchart and number of basic operations performed.

1. Write a program to search for an element in an array using binary and linear search.
2. Write a program to sort list of n numbers using Bubble Sort algorithms.
3. Perform the Insertion and Selection Sort on the input {75,8,1,16,48,3,7,0} and display the output in descending order.
4. Write a program to insert the elements {61,16,8,27} into singly linked list and delete 8,61,27 from the list. Display your list after each insertion and deletion.
5. Write a program to insert the elements {45, 34, 10, 63,3} into linear queue and delete three elements from the list. Display your list after each insertion and deletion.
6. Write a program to simulate the working of Circular queue using an array.
7. Write a program to insert the elements {61,16,8,27} into ordered singly linked list and delete 8,61,27 from the list. Display your list after each insertion and deletion.
8. Write a program for Tower of Hanoi problem using recursion.
9. Write recursive program to find GCD of 3 numbers.
10. Write a program to demonstrate working of stack using linked list.
11. Write a program to convert an infix expression $x^y/(5*z)+2$ to its postfix expression
12. Write a program to evaluate a postfix expression $5\ 3+8\ 2 - *$.
13. Write a program to create a binary tree with the elements {18,15,40,50,30,17,41} after creation insert 45 and 19 into tree and delete 15,17 and 41 from tree. Display the tree on each insertion and deletion operation.
14. Write a program to create binary search tree with the elements {2,5,1,3,9,0,6} and perform inorder, preorder and post order traversal.
15. Write a program to Sort the following elements using heap sort {9.16,32,8,4,1,5,8,0}.
16. Given $S1=\{\text{"Flowers"}\}$; $S2=\{\text{"are beautiful"}\}$ I. Find the length of S1 II. Concatenate S1 and S2 III. Extract the substring "low" from S1 IV. Find "are" in S2 and replace it with "is" .
17. Write a program to implement adjacency matrix of a graph.
18. Write a program to insert/retrieve an entry into hash/ from a hash table with open addressing



using linear probing.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	1	1	-	-	1	-	3	2	2	1	1
CO2	3	3	2	2	-	-	1	-	3	3	2	1	1
CO3	3	3	3	2	1	-	2	1	3	3	3	2	2
CO4	3	3	3	3	2	1	2	2	3	3	3	2	3



Course Code: 24BCA22P	Course Title: Object Oriented Programming Using JAVA Practical
Course Credits: 02	Hours/Week: 04
Total Contact Hours: 52	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03 Hours

Course Outcomes

CO1: Apply fundamental Java concepts including data types, variables, operators, control structures, arrays, and string operations to solve simple programming problems.
CO2: Implement object-oriented programming features such as constructors, inheritance, interfaces, polymorphism, and packages to develop modular and reusable programs.
CO3: Demonstrate exception handling, applet lifecycle, and multithreading mechanisms to build reliable and interactive Java applications.
CO4: Design and execute Java programs to solve real-world problems using mathematical logic, algorithms, and efficient coding practices.

1. Java program to display “Hello World” and display the size of all the data types.
2. Java program to implement the usage of static, local and global variables.
3. Java program to implement string operations string length, string concatenate, substring
4. Java program to find the maximum of three numbers
5. Java program to check whether the number is odd or even.
6. Java program to implement default and parameterized constructors.
7. Java program to implement an array of objects.
8. Java program to implement Single Inheritance
9. Java program to implement Multiple Inheritance using Interface
10. Java program to implement the Life cycle of the applet
11. Java program to demonstrate a division by zero exception
12. Java program to add two integers and two float numbers. When no arguments are supplied give a default value to calculate the sum. Use method overloading.
13. Java program that demonstrates run-time polymorphism.
14. Java program to catch negative array size Exception. This exception is caused when the array is initialized to negative values.
15. Java program to handle null pointer exception and use the “finally” method to display a message to the user.
16. Java program to import user-defined packages



17. Java program to check whether a number is palindrome or not
18. Java program to find the factorial of a list of numbers reading input as command line argument.
19. Java program to display all prime numbers between two limits.
20. Java program to create a thread using Runnable Interface.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	2	1	-	-	1	-	3	2	2	-	1
CO2	3	3	3	2	1	1	1	-	3	3	3	1	2
CO3	2	3	3	2	1	2	2	-	2	3	3	1	2
CO4	3	3	3	3	2	2	2	2	3	3	3	2	3



Course Code: 24BCA23P	Course Title: LINUX and Shell Programming Practical
Course Credits: 02	Hours/Week: 04
Total Contact Hours: 52	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03 Hours

Course Outcome:

CO1: Write and execute shell scripts to perform arithmetic, string, and file operations in Linux.
CO2: Develop shell scripts to implement control structures, loops, and case statements for solving computational problems.
CO3: Apply file handling and directory management techniques using shell scripting to manipulate and manage data.
CO4: Demonstrate process management concepts such as orphan and zombie processes using shell scripts.

1. Write a shell script to print all the prime numbers between M to N ($M < N$).
2. Write a shell script to reverse a given number and check whether it is a palindrome.
3. Write a shell script to find the sum of digits of a given number using loops and without using loops.
4. Write a shell script to implement 10 Linux commands using case.
5. Write a Shell script that displays list of all the files in the current directory to which the user has read, write and execute permissions?
6. Write a shell script to copy a file within current directory
7. Write a shell script to copy file between two directories
8. Write a Shell script to create two data files and compare them to display unique and common entries.
9. Write a shell script to count the number of vowels in a string.
10. Write a shell script to convert uppercase characters to lowercase and vice versa.
11. Write a shell script to accept a word and perform pattern matching in a given file.
12. Write a shell script to find factorial of a number
13. Write a Menu driven program to demonstrate zombie process and orphan process.

**Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)**

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	3	3	3	-	-	2	-	3	3	2	-	2
CO2	2	3	3	3	-	-	2	-	3	3	3	-	2
CO3	2	3	3	3	-	-	2	-	2	3	3	-	3
CO4	2	3	3	3	-	-	2	-	2	3	3	-	2

SEMESTER –III

Theory	24BCA31: Database Management System
Teaching Hours: 04 Hours/Week	Credits: 03
Duration of Exam: 03 Hours	Maximum Marks: 100 (Exam 80 + IA 20)

Course Outcomes:

CO1: Understand fundamental concepts of databases, data models, ER model, relational model, and keys/constraints.
CO2: Apply SQL and PL/SQL commands for data definition, manipulation, queries, joins, views, triggers, and analyze normalization techniques for better database design.
CO3: Analyze transaction management techniques including concurrency control, recovery, serializability, and indexing strategies to improve performance.
CO4: Evaluate NoSQL data models, MongoDB operations, and Big Data technologies (Hadoop, Cassandra) for modern applications.

Course Content:

UNIT I:	[11Hours]
Introduction to Databases: Definition of Data, Database, and DBMS, Overview of Database Applications, Advantages and Disadvantages of DBMS, Roles of Database Users and Administrators, Data Models: Introduction to Data Models, Types of Data Models (Hierarchical, Network, Relational, Object-oriented), Importance of Data Models in DBMS.	
Database Design: Keys: Primary Key, Candidate Key, Super Key, Foreign Key, Composite Key, Alternate Key, Unique Key, Surrogate Key, Constraints in a table: Primary Key, Foreign Key, Unique Key, NOT NULL, CHECK, Entity-Relationship (ER) Model, Entities and Entity Sets, Attributes and Relationships, ER Diagrams, Key Constraints and Weak Entity Sets, Extended ER Features, Introduction to the Relational Model and Relational Schema.	
UNIT II:	[12Hours]
Structured Query Language (SQL): SQL Basics: DDL and DML, Aggregate Functions (Min (), Max(), Sum(), Avg(), Count()), Logical operators (AND, OR, NOT), Predicates (Like, Between, Alias, Distinct), Clauses(Group By, Having, Order by, top/limit), Inner Join, Natural Join, Full Outer Join, Left Outer Join, Right outer Join, Equi Join, Views (Virtual Tables). Assertions and Triggers, Basics of PL/SQL Programming.	
Normalization and Database Design: Functional Dependencies: Armstrong's Axioms, Definition, Properties (Reflexivity, Augmentation, Transitivity), Types (Trivial, Non-Trivial, Partial and Full Functional Dependency), Closure of Functional Dependencies, Normal Forms (1NF, 2NF, 3NF, BCNF), Denormalization.	
UNIT III:	[11 Hours]
Transaction Management: ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock-Based Concurrency Control, Performance of Locking, Transaction Support in SQL, Introduction to Crash Recovery, 2PL, Serializability, and Recoverability, Introduction to Lock Management, Dealing with Deadlocks.	



Database Storage and Indexing: Data on External Storage, File Organizations and Indexing, Index Data Structures, Comparison of File Organizations, Indexes and Performance Tuning, Guidelines for Index Selection, Basic Examples of Index Selection.

UNIT IV:**[11 Hours]**

NoSQL Databases and Big Data: Introduction to NoSQL, Data Models: Document, Key value, Column family, Graph. Uses and Features of NO/SQL document databases. CAP theorem, BASE vs ACID, CRUD operations, MongoDB operators, Overview of Big Data Technologies: Hadoop, MongoDB, Cassandra.

Text Books

1. Elmasri and Navathe, “Fundamentals of Database Systems”, 7th Edition Addison- Wesley, 2016.
2. Raghu Ramakrishnan, Johannes Gehrke, “Database Management Systems”, third edition, McGraw – Hill, 2018.
3. Benjamin Rosenzweig, Elena Rakhimov, “Oracle PL/SQL by Example”, fifth edition, Prentice Hall, 2015
4. Brad Dayley, “NoSQL with MongoDB in 24 Hours”, 1st edition, Sams Publishing, 2024

Reference Books

1. Korth, Silbertz, Sudarshan,” Database System Concepts”, Seventh Edition, McGraw - Hill.(2019)
2. R.P. Mahapatra, Govind Verma, “Database Management Systems”, Khanna Publishing House, 2025.

Web Resources

1. <https://oracle-base.com/articles>
2. https://forums.oracle.com/ords/apexds/domain_dev-community/category/sql_and_pl_sql
3. <https://asktom.oracle.com/ords/f?p=100:1:0>

**Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)**

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	-	-	-	-	1	-	2	-	-	-	-
CO2	3	3	2	2	2	-	-	-	3	3	2	-	-
CO3	2	3	2	2	-	-	-	-	3	2	3	-	-
CO4	2	2	-	3	1	-	2	3	2	3	2	2	3

Theory	24BCA31: Probability and Statistics
Teaching Hours: 04 Hours/Week	Credits: 03
Duration of Exam: 03 Hours	Maximum Marks: 100 (Exam 80 + IA 20)

Course Outcomes:
CO1: Summarize and represent data using measures of central tendency and dispersion.
CO2: Compute correlation and regression to analyze relationships between variables.
CO3: Use probability concepts to solve real-world uncertainty problems.
CO4: Analyze and interpret various probability distributions, sampling and statistical inference.

Course Content:

UNIT I:	[11 Hours]
Basic concepts of Statistics, qualitative and quantitative data, classification of data, construction of frequency distribution, diagrammatic representation of data. Measures of Central Tendency: Arithmetic mean, median and mode—their properties Measures of Dispersion: Range, mean deviation, quartile deviation, variance and standard deviation.	
UNIT II:	[12 Hours]
Correlation: Definition, scatter diagram, types of correlation, measures—Karl Pearson’s correlation coefficient and Spearman’s rank correlation coefficient. Regression: Linear regression-fitting by least square method and interpretation.	
UNIT III:	[11 Hours]
Concepts of probability: Experiment and sample space, events and operations with events, probability of an event, basic probability rules, applications of probability rules, conditional probability. Random Variables: Discrete and continuous random variable, probability distribution of a random variable, probability mass function, probability density function, expectation and variance of a random variable. Standard Probability Distributions: Binomial probability distribution, Poisson probability distribution, Normal probability distribution.	
UNIT IV:	[11 Hours]
Sampling Distribution: Concept of Population and Sample, parameter and statistic, sampling distribution of sample mean and sample proportion. Statistical Inference: Estimation and Hypothesis Testing (only concept).	

Text Books
1. Manish Sharma, Amit Gupta, The Practice of Business Statistics, Khanna Book Publishing Company, 2010 (AICTE Recommended Textbook)
2. Das N. G., Statistical Methods, Combined Edition, Tata McGraw Hill, 2010.
3. Ross Sheldon M., Introduction to Probability and Statistics for Engineers and Scientists, 6 th Edition, Elsevier, 2021.
4. Miller Irwin and Miller Marylees, Mathematical Statistics with Applications, Seventh Edition, Pearson Education, 2005



Reference Books

1. Pal Nabendu and Sarkar Sahadeb, Statistics: Concepts and Applications, Second Edition, PHI, 2013
2. Montgomery Douglas and Runger George C., Applied Statistics and Probability for Engineers, Wiley, 2016.
3. Reena Garg, Engineering Mathematics, Khanna Publishing House, 2024.

Web Resources

1. <https://nptel.ac.in/courses/111106112>
2. <https://nptel.ac.in/courses/111105041>

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	2	-	-	2	2	3	-	2	-	2
CO2	3	3	2	2	-	-	2	2	3	2	3	-	2
CO3	3	3	-	2	-	-	3	2	3	-	3	-	3
CO4	3	3	-	2	-	-	3	2	3	-	3	-	3

Theory	24BCA33: Artificial Intelligence
Teaching Hours: 04 Hours/Week	Credits: 04
Duration of Exam: 03 Hours	Maximum Marks: 100 (Exam 80 + IA 20)

Course Outcomes
CO1: Understand the foundations of Artificial Intelligence, including intelligent agents, environments, and problem-solving approaches.
CO2: Apply advanced search strategies, adversarial game-playing algorithms, and evolutionary methods to solve well-defined problems.
CO3: Analyze logical reasoning techniques, planning methods, and uncertainty handling mechanisms to assess their effectiveness in decision-making.
CO4: Evaluate AI applications across domains such as machine learning, natural language processing, robotics, and expert systems, while reflecting on ethical and legal implications.

Course Content:

UNIT – I Introduction to AI	15 Hours
What is AI? Intelligent Agents: Agents and environment, the concept of Rationality, the nature of environment, the structure of Agents. Knowledge-Based Agents: Introduction to Knowledge Based Agents, The Wumpus World as an Example World. Problem-solving: Problem-solving agents.	
UNIT – II Advanced Search Techniques	15 Hours
Uninformed Search: DFS, BFS, Iterative Deepening Search. Informed Search: Best First Search, A* search, AO* search. Adversarial Search & Games: Two-player zero-sum games, Minimax Search, Alpha-Beta pruning. Constraints and Constraint Satisfaction Problems (CSPs), Backtracking search for CSP. Evolutionary Search Techniques: Introduction to evolutionary algorithms, Genetic algorithms, Applications of evolutionary search in AI.	
UNIT – III Logical Reasoning and Uncertainty	15 Hours
Logic: Propositional logic, First-order predicate logic, Propositional versus first-order inference, Unification and lifting. Inference: Forward chaining, Backward chaining, Resolution, Truth maintenance systems. Introduction to Planning: Blocks World problem, Strips; Handling Uncertainties: Non-monotonic reasoning, Probabilistic reasoning, Introduction to Fuzzy set theory.	

UNIT – IV Domains and Applications of AI **15 Hours**

Domains in AI: Introduction to Machine Learning, Computer Vision, Robotics, Natural Language Processing, Deep Neural Networks, and their Applications. Expert Systems: The architecture and role of expert systems include two case studies. Legal and Ethical Issues: Concerns related to AI.

**TEXT BOOKS:**

1. M.C. Trivedi, *A Classical Approach to Artificial Intelligence*, Khanna Book Publishing Company, 2024 (AICTE Recommended).
2. Nilsson Nils J., *Artificial Intelligence: A New Synthesis*, Morgan Kaufmann Publishers Inc., ISBN: 978-1-55860-467-4.
3. Dan W. Patterson, *Introduction to Artificial Intelligence & Expert Systems*, PHI Learning, 2010.
4. Rajiv Chopra, *Data Science with Artificial Intelligence, Machine Learning and Deep Learning*, Khanna Book Publishing Company, 2024.
5. Russell, S. and Norvig, P., *Artificial Intelligence – A Modern Approach*, 3rd Edition, Prentice Hall.

REFERENCE BOOKS:

1. M.C. Trivedi, *Introduction to AI and Machine Learning*, Khanna Book Publishing Company, 2024.
2. Van Hirtum, A. & Kolski, C., *Constraint Satisfaction Problems: Algorithms and Applications*, Springer, 2020.
3. Elaine Rich, Kevin Knight, Shivasankar B. Nair, *Artificial Intelligence*, 3rd Edition, McGraw Hill, 2019.
4. Rajiv Chopra, *Machine Learning and Machine Intelligence*, Khanna Book Publishing Company, 2024.

**Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs1-8)**

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	3	2	-	-	-	2	-	3	2	3	-	2
CO2	3	3	3	2	-	-	2	-	3	3	3	-	2
CO3	3	3	2	2	-	-	2	-	3	2	3	-	2
CO4	3	3	3	3	2	3	3	2	3	3	3	3	3



Practical	24BCA31: Database Management System Lab
Teaching Hours: 04 Hours/Week	Credits: 02
Duration of Exam: 03 Hours	Maximum Marks: 50(Exam 40 + IA 10)

Course outcome:

CO1: Apply DDL, DML, and TCL commands to create, manipulate, and manage database objects.

CO2: Formulate and execute SQL queries using constraints, aggregate functions, grouping, joins, set operators, and advanced queries (analytical, hierarchical, recursive).

CO3: Develop and test PL/SQL programs to solve computational and database-related problems.

CO4: Demonstrate proficiency in handling unstructured data using MongoDB queries for CRUD operations.

List of Practical:

Database Design:

Draw an ER Diagram of Registrar Office

Draw an ER Diagram of Hospital Management System

Reduce The ER diagram in question no 1 into tables

Reduce the ER diagram of question no 2 into tables

Consider the following Schema Supplier (SID, Sname, branch, city, phone) Part(PID, Pname, color, price)

Supplies(SID, PID, qty, date_supplied)

DDL Commands

1. Create the above tables
2. Add a new attribute state in supplier table
3. Remove attribute city from supplier table
4. Modify the data type of phone attribute
5. Change the name of attribute city to address
6. Change a table's name, supplier to sup
7. Use truncate to delete the contents of supplies table
8. Remove the part table from database

DML Commands

1. Insert at least 10 records in tables supplier, part and supplies



2. Show the contents in tables supplier, part and supplies
3. Find the name and city of all suppliers
4. Find the name and phoneno of all suppliers who stay in 'Delhi'
5. Find all distinct branches of suppliers
6. Delete the record of the supplier whose SID is 204001
7. Delete all records of supplier table
8. Delete all records of suppliers whose city starts with capital A.
9. Find the supplier names which have 'lk' in any position
10. Find the supplier name where 'R' is in the second position
11. Find the name of supplier whose name starts with 'V' and ends with 'A'
12. Change the city of all suppliers to 'BOMBAY'
13. Change the city of supplier 'Vandana' to 'Goa'

Queries with Constraints

1. Create the supplier table with Primary Key Constraint
2. Create supplies table with Foreign key Constraint
3. Create a part table with UNIQUE Constraint
4. Create supplier Table with Check Constraints
5. Create Supplier table with
6. Default Constraint Queries on TCL
7. Create Savepoints
8. Rollback to SavePoints
9. Use Commit to save on

Aggregate Functions:

1. Find the minimum, maximum, average and sum of costs of parts
2. Count the total number of parts present
3. Retrieve the average cost of all parts supplied by
4. 'Mike' Queries on GROUP BY, HAVING AND ORDER BY Clauses
5. Display total price of parts of each color
6. Find the branch and the number of suppliers in that branch for branches which have more than 2 suppliers
7. Find all parts sorted by pname in ascending order and cost in descending order
8. Find the branch and the number of suppliers in that branch

Queries on Analytical, Hierarchical, Recursive nature.

1. Find out the 5th highest earning employee details.
2. Which department has the highest number of employees with a salary above \$80,000, and what percentage of employees in that department have a salary above \$80,000
3. Retrieve employee table details using the hierarchy query and display that hierarchy path starting from the top level indicating if it is a leaf and there exists a cycle.
4. What is the average salary for employees in the top 2 departments with the highest average salary, and what is the hierarchy of departments and sub-departments for these top 2 departments?
5. Use recursion to retrieve the employee table and display the result in breadth first and depth first



order.

6. Write a recursive query to show the equivalent of level, connect_by_root and connect_by_path
7. Use recursion to retrieve the employee table and display the result in depth first order showing id, parent_id, level, root_id, path and leaf.

Queries on Operators

1. Find the pname, phoneno and cost of parts which have cost equal to or greater than 200 and less than or equal to 600.
2. Find the sname, SID and branch of suppliers who are in 'local' branch or 'global' branch
3. Find the pname, phoneno and cost of parts for which cost is between 200 and 600
4. Find the pname and color of parts, which has the word 'NET' anywhere in its pname.
5. Find the PID and pname of parts with pname either 'NUT' or 'BOLT'
6. List the suppliers who supplied parts on '1st may2000', '12 JAN 2021', '17 dec 2000', '10 Jan 2021'
7. Find all the distinct costs of parts

Join Operators

1. Perform Inner join on two tables
2. Perform Natural Join on two tables
3. Perform Left Outer Join on tables
4. Perform Right Outer join on tables
5. Perform Full Outer Join on tables

Set Theory Operators

1. Show the use of UNION operator with union compatibility
2. Show the use of intersect operator with union compatibility
3. Show the use of minus operator with union compatibility
4. Find the cartesian product of two tables

Queries on Set Theory Operators

1. List all parts except 'NUT' and 'BOLT' in ascending order of costs
2. display all parts that have not been supplied so far
3. To display the supplier names who have supplied 'green' part with cost 500 Rupees AND 'red' part with cost 400 Rupees.
4. To display the supplier names who have supplied 'green' part with cost 500 Rupees OR 'red' part with cost 400 Rupees.
5. To Display the name of suppliers who have supplied all parts that are 'red' in color.

PL/SQL Programs

1. Write a PL/SQL Code to add two numbers
2. Write a PL/SQL code for Fibonacci series
3. Write a PL/SQL Code for greatest of 3 numbers
4. Write a PL/SQL code for area and circumference of a circle



MongoDB Queries

1. Create a collection and insert documents into it using insertOne() and insertMany()
2. Select all documents in collection
3. Find the count of all suppliers
4. Find all records that have city = 'Delhi'
5. Retrieve all documents that have color equal to 'red' or 'green'
6. Retrieve all documents where part_name is 'P1' or price is less than 200.
7. Update the record of 'Geeta' ,set city = 'Bombay' and phoneno = '11223344'
8. Delete all records where price is greater than 5000
9. Display only the name and city of the supplier
10. Sort all suppliers on city and display only the first two records.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	2	2	-	-	2	2	3	2	2	-	2
CO2	3	3	2	2	-	-	2	2	3	2	3	-	2
CO3	2	3	3	2	-	-	2	2	2	3	3	-	2
CO4	2	2	2	3	-	-	2	3	2	3	2	-	2



Practical	24BCA31: Artificial intelligence Lab
Teaching Hours: 04 Hours/Week	Credits: 02
Duration of Exam: 03 Hours	Maximum Marks: 50(Exam 40 + IA 10)

CO1: Demonstrate basic Python programming concepts including input/output, collections, string operations, exception handling, and file handling.

CO2: Apply Python libraries and functions such as NumPy, Pandas, lambda, map, and regular expressions for problem solving.

CO3: Implement classical search and optimization algorithms such as DFS, BFS, Hill Climbing, Minimax, and Constraint Propagation.

CO4: Use Python to represent knowledge, build simple expert systems, and perform basic Natural Language Processing tasks like tokenization, stop-word removal, and stemming.

PART-A

1. Console input and output.
2. Implementation of Python Collections.
3. Create a simple calculator.
4. Demonstration of Lambda Functions and Map Function in Python
5. Demonstration of common String Functions in Python.
6. Demonstration of Exception handling
7. Create arrays using numpy and perform array operations.
8. Generate Random Numbers using NumPy
9. Create a NumPy array and demonstrate slicing using both positive and negative indexing.
10. Read and write into file.
11. Create DataFrames from an Excel sheet using Pandas and perform operations on DataFrames.
12. Create a DataFrame using a Python dictionary and perform basic operations such as displaying specific columns and finding the average of a numeric column.
13. Create a DataFrame with product details. Filter the products whose price is greater than 10 and sort the DataFrame by stock in descending order.
14. Matplotlib Visualization Suite

PART-B

1. Implement Depth-First Search (DFS) on a small graph.
2. Implement Breadth-First Search (BFS) on a small graph.
3. Implement a Hill Climbing search to find the peak in a numeric dataset.
4. Implement the Minimax Search Algorithm for a 2-Player Game Tree (3 Plies)
5. Use constraint propagation to solve a Magic Square puzzle.
6. Apply optimization techniques to find the maximum value in a list.
7. Represent and evaluate propositional logic expressions.
8. Implement a basic rule-based expert system for weather classification.



9. Using Python NLTK, perform the following Natural Language Processing (NLP) tasks for text content:

- a) Tokenizing
- b) Filtering Stop Words
- c) Stemming

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	2	2	-	-	2	-	3	2	2	-	2
CO2	3	2	2	3	-	-	2	2	3	3	2	-	2
CO3	2	3	2	2	-	-	2	2	2	2	3	-	2
CO4	2	3	2	2	-	-	3	2	2	2	3	-	3

**SEMESTER –IV**

Theory	24BCA33: Computer Networks
Teaching Hours: 04 Hours/Week	Credits: 03
Duration of Exam: 03 Hours	Maximum Marks: 100 (Exam 80 + IA 20)

Course Outcomes:

CO1: Understand the fundamentals of computer networks, models, topologies, transmission modes, and networking devices.

CO2: Apply addressing schemes, subnetting, VLANs, and data link/transport layer protocols to configure simple networks.

CO3: Analyze routing algorithms, TCP/UDP mechanisms, congestion control, and evaluate basic network security techniques.

CO4: Evaluate application layer protocols and explore emerging networking technologies such as SDN, NFV, and IoT.

UNIT I: Introduction to Computer Networks [11 Hours]

Overview of Computer Networks: Definition and Objectives, Applications and Examples Network Components and Architecture, Network Models: OSI Model: Layers and Functions, TCP/IP Model: Layers and Functions Comparison between OSI and TCP/IP Models.

Network Topologies: Physical vs. Logical Topologies, Common Topologies: Star, Ring, Bus, Mesh, Hybrid, Advantages and Disadvantages of Each Topology, Data Transmission: Analog vs. Digital Signals, Transmission Modes: Simplex, Half-Duplex, Full-Duplex, Bandwidth and Latency, Networking Devices: Routers, Switches, Hubs, Bridges, Gateways, Functions and Configurations of Each Device.

UNIT II: Data Link Layer and Networking Protocols [12 Hours]

Data Link Layer Fundamentals: Functions of the Data Link Layer, Framing, Error Detection, and Error Correction, Flow Control Mechanisms. Ethernet: Ethernet Standards and Frame Structure, MAC Addressing and ARP, Ethernet Switching: Basic Concepts and Methods

Network Protocols: Introduction to TCP/IP Protocol Suite, IP Addressing: IPv4 and IPv6 Subnetting and CIDR Notation Address Resolution Protocol (ARP): ARP Operation and Table, ARP Spoofing and Security Considerations, Virtual LANs (VLANs): Concept of VLANs, VLAN Tagging and Configuration, Benefits and Use Cases.

UNIT III: Network Layer and Layer Transport [11 Hours]

Network Layer: IP Routing: Static vs. Dynamic Routing, Routing Protocols: RIP, OSPF, BGP, Network Address Translation (NAT), Transport Layer: TCP vs. UDP: Characteristics and Use Cases, TCP Handshake and Connection Management, Flow Control and Congestion Control in TCP, Congestion Control Algorithms: Techniques: Slow Start, Congestion Avoidance, Fast Retransmit, Fast Recovery, TCP Variants: TCP Reno, TCP Vegas.

Network Security Fundamentals: Threats and Vulnerabilities, Basic Security Mechanisms: Firewalls, VPNs, Encryption.

UNIT IV: Application Layer and Emerging Technologies [11 Hours]



Application Layer Protocols: HTTP/HTTPS: Structure and Operation, FTP, SMTP, POP3, IMAP: Protocols and Uses, DNS: Domain Name System and Resolution. Network Applications: Web Browsing, Email Communication, File Transfer, Voice over IP (VoIP) and Streaming. Emerging Technologies: Software-Defined Networking (SDN), Network Function Virtualization (NFV), Internet of Things (IoT) and Its Impact on Networking

Text Books:

1. Andrew S. Tanenbaum, "Computer Networks", 5th Edition, Pearson Education, 2011.
2. James F. Kurose and Keith W. Ross, "Computer Networking: A Top-Down Approach", 8th Edition, Pearson, 2021.

Reference Books:

1. Behrouz A. Forouzan, "Data Communications and Networking", 5th Edition, McGraw-Hill Education, 2012.
2. Larry L. Peterson and Bruce S. Davie, "Computer Networks: A Systems Approach", 6th Edition, Morgan Kaufmann, 2019.
3. Bhavneet Sidhu, An Integrated Approach to Computer Networks, Khanna Publishing House, 2023.
4. Mastering PC Hardware & Networking, Khanna Publishing House, 2024.

Web Resources:

1. [Cisco Networking Academy - Online Courses and Resources](#)
2. [NetworkLessons.com - Tutorials on Various Networking Topics](#)



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	-	-	-	-	1	-	2	-	-	-	-
CO2	3	3	3	2	2	-	-	-	3	2	-	-	-
CO3	3	3	2	3	-	-	2	2	2	3	2	-	-
CO4	3	2	3	3	1	-	2	3	2	3	2	2	3



Theory	24BCA42: Design And Analysis of Algorithms
Teaching Hours: 04 Hours/Week	Credits: 03
Duration of Exam: 03 Hours	Maximum Marks: 100 (Exam 80 + IA 20)

Course Outcomes:

CO1: Understand the fundamentals of algorithm design, performance analysis, asymptotic notations, and recursive algorithms to measure efficiency.

CO2: Apply divide and conquer, greedy strategies, and sorting/searching techniques to solve computational problems with best/worst case analysis.

CO3: Analyze dynamic programming and graph algorithms to solve optimization problems like shortest paths, connectivity, and topological ordering.

CO4: Evaluate problem-solving strategies using backtracking and classify computational problems into P, NP, NP-Complete, and NP-Hard categories.

UNIT I:**[11 Hours]**

What is an algorithm? Design and performance analysis of algorithms, time complexity, space complexity. Asymptotic notations (O , Ω , Θ) to measure growth of a function and application to measure complexity of algorithms. Analysis of sequential search, bubble sort, selection sort, insertion sort, matrix multiplication. Recursion: Basic concept. Analysis of recursive algorithms, Master's theorem.

UNIT II:**[12 Hours]**

The Divide & Conquer Design Technique:

The general concept. Binary search, finding the maximum and minimum, merge sort, quick sort. Best and worst case analysis for the mentioned algorithms. Strassen's matrix multiplication.

Lower bound for comparison-based sorting. The Greedy Design Technique: The general concept. Applications to general Knapsack problem, finding minimum weight spanning trees: Prim's and Kruskal's algorithms, Dijkstra's algorithm for finding single source shortest paths problem.

UNIT III:**[11 Hours]**

The Dynamic Programming Design Technique: The general concept. Computation of Fibonacci series and Binomial coefficients, all pair shortest paths problem (Floyd-Warshall's algorithm), 0/1 Knapsack problem. Algorithms on Graphs: Breadth First Search, Depth First Search, finding connected components, depth first search of a directed graph, topological sorting.

UNIT IV:**[11 Hours]**

Limitations of Algorithmic Power: Backtracking Method: n-Queen problem; sum of subsets problem/ Hamiltonian circuit problem/vertex cover problem. Computational intractability: Overview of non-deterministic algorithms, P, NP, NP-Complete and NP-hard problems.

**Text Books**

1. Levitin Anany, Introduction to Design and Analysis of Algorithms, 3rd Edition, Pearson, 2012
2. Gajendra Sharma, Design and Analysis of Algorithms, Khanna Publishing House (AICTE Recommended Textbook)
3. Cormen Thomas H., Leiserson Charles E., Rivest Ronald L. and Stein Clifford, Introduction to Algorithms, PHI publication, 3rd Edition, 2009.
4. Horowitz Ellis, Sahni Sartaj and Rajasekaran Sanguthevar, Fundamentals of Computer Algorithms, University Press (I) Pvt. Ltd., 2012.

Reference Books

1. Aho Alfred V., Hopcroft John E. & Ullman Jeffrey D., The Design & Analysis of Computer Algorithms, Addison Wesley Publications, Boston, 1983.
2. Kleinberg Jon & Tardos Eva, Algorithm Design, Pearson Education, 2006.

Web Resources

<https://nptel.ac.in/courses/106101060>

<https://www.cs.umd.edu/~mount/451/Lects/451lects.pdf>

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	3	2	2	-	-	1	-	3	2	2	-	1
CO2	3	3	3	2	-	-	1	-	3	3	3	-	2
CO3	3	3	3	2	-	-	1	-	3	3	3	-	2
CO4	3	3	2	3	-	-	2	-	3	2	3	-	2



Theory	24BCA43: Software Engineering
Teaching Hours: 04 Hours/Week	Credits: 04
Duration of Exam: 03 Hours	Maximum Marks: 100 (Exam 80 + IA 20)

Course Outcomes
CO1: Explain software development life cycle models, agile principles, and compare traditional and agile methodologies.
CO2: Apply requirements engineering processes, project planning techniques, and risk management strategies in software development.
CO3: Develop software design models, UML diagrams, and apply testing strategies to ensure quality and reliability.
CO4: Analyze and evaluate software quality assurance practices, release management strategies, and maintenance processes for sustainable software systems.

Course Content:
UNIT I: [15 Hours]
The evolving role of software, changing nature of software, layered technology, a process framework, Process models: The waterfall model, incremental process models, evolutionary process models, the unified process. Agile software development: Agility Principles, Agile methods, Plan-driven and agile development, Extreme programming, Scrum, A Tool Set for the Agile Process.
UNIT II: [15 Hours]
Software Requirements Engineering: Functional and non-functional requirements, the software requirements document, Requirements specification, Requirements engineering processes, Requirements elicitation and analysis, Requirements validation, Requirements management. Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan. Project planning- Software pricing, Plan-driven development, Project scheduling, Agile planning, Estimation techniques.
UNIT III: [15 Hours]
Design: Design process and design quality, design concepts, the design model, software architecture, data design, architectural design, Basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.
Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging. Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.
UNIT IV: [15 Hours]
Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability. Release Management: Release planning, development and build plans, release strategies, risk management, and post-deployment monitoring. Product sustenance: Maintenance, updates, End of life, migration strategies.

**Text Books**

1. Software Engineering, N.S. Gill, Khanna Publishing House, 2023 (AICTE Recommended Textbook)
2. Software Engineering, Ian Somerville, 9th edition, Pearson education.
3. Software Engineering a Practitioner's Approach, 8th edition, Roger S Pressman, Bruce R. Maxim. McGraw Hill Education, 2015.

Reference Books

1. Stephen Schach, Software Engineering 7th ed, McGraw-Hill, 2007
2. Software Engineering: Principles and Practice Hans van Vliet

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	3	-	3	-	-	2	-	3	2	2	-	3
CO2	2	3	-	3	-	-	2	-	3	2	2	-	2
CO3	2	3	3	3	-	-	2	-	2	3	3	-	3
CO4	2	3	3	3	-	-	2	-	2	3	3	-	2



Practical	24BCA41: Computer Networks Lab
Teaching Hours: 04 Hours/Week	Credits: 02
Duration of Exam: 03 Hours	Maximum Marks: 50(Exam 40 + IA 10)

CO1: Understand and configure basic network settings, IP addressing, subnetting, VLANs, and routing/switching in real and simulated environments.

CO2: Apply networking protocols by implementing DNS resolution, HTTP client-server communication, and deploying simple web servers.

CO3: Analyze network performance and security using tools like Wireshark, ping, traceroute, iperf, firewalls, and VPNs.

CO4: Diagnose, troubleshoot, and resolve connectivity problems while applying systematic debugging techniques in network environments.

Lab Programs:

1. Configure Basic Network Settings:
 - a. IP Address Configuration
 - b. Subnet Mask and Gateway Settings
2. Implement Network Protocols:
 - a. Write a simple Python script to perform DNS resolution.
 - b. Implement a basic HTTP client-server application.
3. Network Simulation:
 - a. Use network simulation tools (e.g., Cisco Packet Tracer) to design and simulate network topologies.
 - b. Configure routers and switches in a simulated environment.
4. Performance Measurement:
 - a. Measure network performance using tools like `ping`, `traceroute`, and `iperf`.
 - b. Analyze network traffic using Wireshark.
5. Implement VLANs:
 - a. Configure VLANs on a switch and verify using simulation tools.
6. Set Up a Simple Web Server:
 - a. Deploy a basic web server and configure HTTP/HTTPS access.
7. Network Security Lab:
 - a. Implement basic firewall rules and VPN configurations.
 - b. Perform vulnerability scanning and analyze results.
8. Network Troubleshooting:
 - a. Diagnose and resolve common network issues.
 - b. Use troubleshooting commands and techniques to fix connectivity problems.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	1	3	-	-	1	-	3	1	2	-	-
CO2	2	3	2	2	-	-	1	-	3	3	3	-	-
CO3	2	3	3	3	-	1	2	-	2	2	3	1	1
CO4	1	3	1	2	-	-	1	-	2	2	1	-	-



Practical	24BCA42P: Design and Analysis of Algorithms Lab
Teaching Hours: 04 Hours/Week	Credits: 02
Duration of Exam: 03 Hours	Maximum Marks: 50(Exam 40 + IA 10)

CO1: Recall and describe the properties and fundamental concepts of various sorting, searching, and graph traversal algorithms. This includes recognizing the basic steps and characteristics of algorithms like Bubble Sort, Linear Search, and DFS.

CO2: Implement and execute fundamental algorithms to solve given computational problems. They will demonstrate the ability to translate algorithmic logic into code for algorithms such as Merge Sort and BFS.

CO3: Analyze and compare the time and space complexity of different algorithms. This involves evaluating the efficiency of various algorithms to determine their best use cases.

CO4: Design and develop a new algorithm or modify an existing one to solve a novel problem. This showcases the ability to apply theoretical knowledge to create original solutions, justifying design choices based on efficiency.

Lab Programs:

1. Solve the following problems, and find the complexity of the best and worst case:
2. Write a program to implement linear search algorithm.
3. Write a program to implement binary search algorithm.
4. Write a Program to sort a given set of numbers using Selection sort algorithm.
5. Write a program to sort a given set of numbers using Bubble sort algorithm.
6. Write a program to sort a given set of numbers using Insertion sort algorithm.
7. Write a program to sort a given set of numbers using Merge sort algorithm.
8. Write a program to implement matrix multiplication.
9. Write a program to find the factorial of a number using recursive function.
10. Write a program to find the factorial of a number using recursive function.
11. Write a program to find the minimum spanning tree of a given graph using Prim's algorithm.
12. Write a program to find the minimum spanning tree of a given graph using Kruskal's algorithm.
13. Write a Program to find the binomial co-efficient $C(n, k)$, [where n and k are integers and $n > k$] using brute force based algorithm and also dynamic programming based algorithm.
14. Write a program to implement BFS traversal algorithm.
15. Write a program to implement DFS traversal algorithm.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	-	-	-	-	-	3	-	-	-	-
CO2	3	3	2	2	-	-	-	-	3	3	3	-	-
CO3	3	3	2	-	-	-	2	-	3	-	3	-	2
CO4	2	3	3	-	-	-	3	-	2	2	3	-	3



Theory	24BCA34: Cyber Security
Teaching Hours : 02 Hours/Week	Credits : 02
Duration of Exam : 1.5Hours	Maximum Marks : 50(Exam 40 + IA 10)

Course Outcomes

CO1: Explain the fundamentals of cyber security, types of cyber threats, and core security concepts including CIA triad, firewalls, and malware.

CO2: Apply secure communication techniques, encryption methods, and safe internet practices to protect personal and organizational data.

CO3: Demonstrate system, personal, and device security practices, including authentication, password policies, antivirus use, and safe web and mobile app usage.

Course Content:**Unit I – Fundamentals of Cyber Security & Secure Communication [14 Hours]**

Introduction to Cyber Security: Importance of cyber safety in the digital world, Common types of cyber threats: viruses, worms, trojans, phishing, ransomware, fake websites, social engineering, Real-life examples of cyber attacks. Goals of Cyber Security: Confidentiality, Integrity, Availability (CIA Triad), Difference between threats, vulnerabilities, and risks. Security Concepts and Terminology: Firewall, antivirus, malware, hacking. Safe Internet Practices: Responsible digital behaviour, Secure browsing and identifying fake/unsafe websites, Email safety: detecting phishing and spam, Use of public Wi-Fi and avoiding data leaks. Introduction to Cryptography and Secure Communication: Purpose of encryption and decryption, Difference between symmetric and asymmetric encryption, Role of HTTPS and SSL, Two-Factor Authentication (2FA), OTP-based logins, Importance of secure messaging and email

Unit II – System, Personal, and Device Security Practices [14 Hours]

System and Device Security Basics: Operating System security basics, User authentication and access control, Password policies and best practices, Antivirus and anti-malware software. Web and App Security: Common web security issues: SQL injection, XSS, CSRF (overview only), Mobile app security – permissions and safe usage, Safe use of mobile devices, laptops, and social media, Software updates and patch management. Network and Online Protection: Introduction to firewalls and VPNs, Risks of pop-ups, cookies, and online tracking, Safe and responsible data sharing online



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	2	-	3	2	-	3	-	2	3	2
CO2	2	3	-	3	-	3	2	-	3	-	3	3	2
CO3	2	3	-	3	-	3	2	-	3	-	3	3	2

**III Semester (Elective)**

Elective	24BCAE1: Feature Engineering
Teaching Hours: 02 Hours/Week	Credits: 02
Duration of Exam: 1.5Hours	Maximum Marks: 50(Exam 40 + IA 10)

Course Outcomes

CO1: Define data types and explain the importance of features in machine learning.

CO2: Apply problem-solving techniques using agent-based approaches in sample AI environments.

CO3: Implement feature engineering methods for numerical and categorical data.

CO4: Analyze and compare feature selection and reduction techniques such as Filter, Wrapper, Hybrid methods, and PCA.

Course Content:**UNIT I: Introduction to Feature Engineering [15 Hours]**

Introduction to Data and Features: Importance of Features in Machine Learning. Data types and features: Numerical, Categorical, Ordinal, Discrete, Continuous, Interval and Ratio. Basic Feature Preprocessing: Handling Missing Data, Data Cleaning, Feature Scaling, Normalization, and Transformation.

UNIT II: Feature Engineering Techniques [15 Hours]

Techniques for Numerical Data: Binning and Discretization, Polynomial and Interaction Features. Categorical Data Techniques: One Hot Encoding, Label Encoding. Feature extraction vs. feature selection, Steps in feature selection. Feature Selection Methods: Filter, Wrapper, and Hybrid. Feature Reduction: Introduction and application of Principal Components Analysis.

Text Books

1. M.C. Trivedi, Data Science and Data Analytics Using Python Programming, Khanna Publishing House, 2024.
2. Zheng, Alice, & Casari, Amanda. (2018). Feature engineering for machine learning: Principles and techniques for data scientists. O'Reilly Media, Inc.
3. Kalita, J. K., Bhattacharyya, D. K., & Roy, S. (2023). Fundamentals of Data Science: Theory and Practice. Elsevier. ISBN-13: 9780323917780.

Reference Books:

1. Duda, R. O., Hart, P. E., Stork, D (2007). Pattern classification (2Ed), John Wiley & Sons, ISBN-13: 978-8126511167.
2. N. Bhaskar, Vasundhara, Machine Learning, Khanna Publishing House, 2024.
3. M.C. Trivedi, Deep Learning and Neural Network_MC Trivedi, Khanna Publishing House, 2024.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2			-	-	2	-	3	-	2	-	2
CO2	2	3	2	2	-	-	2	2	2	2	3	-	2
CO3	3	2	3	3	-	-	2	2	3	3	2	-	2
CO4	3	3	2	3	-	-	3	2	3	2	3	-	3



Elective	24BCAE1: Web Programming-I (Full Stack Development)
Teaching Hours: 02 Hours/Week	Credits: 02
Duration of Exam: 1.5Hours	Maximum Marks: 50(Exam 40 + IA 10)

Course Outcomes

CO1: Explain the structure and elements of HTML, CSS, and Bootstrap to design responsive web pages.
CO2: Apply JavaScript functions, events, and form validation techniques to enhance interactivity in web applications.
CO3: Analyze and implement client-server communication using AJAX, XML, and JSON for dynamic data handling.
CO4: Demonstrate the ability to integrate HTML, CSS, JavaScript, and Bootstrap to develop functional and user-friendly web applications.

Course Content:

UNIT I:	[15 hours]
Introduction to HTML: history of HTML, Objective, basic Structures of HTML, Header Tags, body tags, Paragraph Tags.	
Tags for FORM Creation, TABLE, FORM, TEXTAREA, SELECT, IMG, IFRAME FIELDSET, ANCHOR.	
Lists in HTML, Introduction to DIV tag, NAVBAR Design.	
Introduction to CSS, types, Selectors, and Responsiveness of a web page.	
Introduction to Bootstrap, downloads/linking, using classes of Bootstrap, understanding the Grid System in Bootstrap.	
Introduction to www, Protocols and Programs, Applications and development tools, web browsers, DNS, Web hosting Provider, Setting up of Windows/Linux/Unix web servers, Web hosting in cloud, Types of Web Hosting.	
UNIT II:	[15 hours]
Introduction to JavaScript: Functions and Events. Document Object model traversing using JavaScript. Output System in JavaScript i.e. Alert, throughput, Input box, Console. Variables and Arrays in JavaScript. Date and String handling in JavaScript.	
Manipulating CSS through JavaScript: Form Validation like Required validator, length validator, Pattern validator. Advanced JavaScript, Combining HTML, CSS and JavaScript events and buttons, controlling your browser.	
Introduction to AJAX: Purpose, advantages and disadvantages, AJAX based Web applications and alternatives of AJAX.	
Introduction to XML: uses, Key concepts, DTD schemas, XSL, XSLT, and XSL Elements and transforming with XSLT. Introduction to XHTML.	
JSON: Introduction to JSON, Keys and Values, Types of Values, Arrays, Objects	



Course Mapping

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	2	2	-	-	2	-	3	2	2	-	2
CO2	2	3	3	2	-	-	2	-	2	3	3	-	2
CO3	2	3	3	2	-	-	2	-	-	2	3	-	3
CO4	2	3	3	3	-	-	2	-	-	3	3	-	3

Text Books
1. Laura Lemay, Mastering HTML, CSS & Java Script Web Publishing, BPB Publications, 2016
2. Thomas A. Powell, The Complete Reference HTML & CSS, Fifth Edition, 2017
3. Reference Books
4. Silvio Moreto, Bootstrap 4 By Example, ebook, 2016.

Web Resources
1. www.javatpoint.com
2. www.w3schools.com
3. https://www.geeksforgeeks.org/web-technology/



VI Semester (Elective)

Elective	24BCAE2: Feature Engineering-II(AI&ML)
Teaching Hours: 02 Hours/Week	Credits: 02
Duration of Exam: 1.5Hours	Maximum Marks: 50(Exam 40 + IA 10)
Course Outcomes	
CO1: Explain the fundamental concepts, types, and applications of machine learning along with evaluation parameters.	
CO2: Apply supervised learning techniques such as regression and classification to solve real-world problems.	
CO3: Implement unsupervised learning methods including clustering algorithms and evaluate their performance.	
CO4: Analyze ethical considerations and case studies to demonstrate the practical applications of machine learning.	

Course Content:

UNIT I: Introduction to Machine Learning	[15 Hours]
Introduction: Definition, History and Application of Machine Learning, Types of Machine Learning: Supervised, Unsupervised, Semi-Supervised, and Reinforcement Learning. Labeled and Unlabelled Dataset. Supervised Learning Tasks: Regression vs. Classification, Learning Framework: Training, Validation and Testing of ML models. Performance Evaluation Parameters: Confusion matrix, Accuracy, Precision, Recall, F1 Score, and AUC.	
UNIT II: Supervised Learning and Unsupervised Learning	[15 Hours]
Regression: Linear and non-linear Regression, Logistic Regression. Classification: Naïve Bayes, K-Nearest Neighbors, Clustering: K-Means, Hierarchical Clustering, DBSCAN, Clustering Validation Measures. ML Applications: Ethical Considerations in Machine Learning, Case study and Real-world Applications.	

Text Books:
1. Rajiv Chopra (2024), Machine Learning and Machine Intelligence, Khanna Publishing House.
2. Jeeva Jose (2023), Introduction to Machine Learning, Khanna Publishing House.
3. Mitchell T. (1997). Machine Learning, First Edition, McGraw-Hill.
4. Kalita, J. K., Bhattacharyya, D. K., & Roy, S. (2023). Fundamentals of Data Science: Theory and Practice. Elsevier. ISBN9780323917780

Reference Books:
1. Flach, P. A. (2012). Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge University Press. ISBN: 9781107422223, 2012.
2. Duda, R. O., Hart, P. E., Stork, D (2007). Pattern classification (2Ed), John Wiley & Sons, ISBN-13: 978-8126511167.
3. Haykin S. (2009). Neural Networks and Learning Machines, Third Edition, PHI Learning.

**Course Articulation Matrix:** Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	3	-	2	-	-	2	-	3	2	2	-	3
CO2	2	3	3	3	-	-	2	-	3	3	2	-	3
CO3	2	3	3	3	-	-	2	-	2	3	3	-	2
CO4	2	2	-	3	-	-	3	3	2	2	3	-	2



Elective	24BCAE2: Web Programming-II (Full Stack Development)
Teaching Hours: 02 Hours/Week	Credits: 02
Duration of Exam: 1.5Hours	Maximum Marks: 50(Exam 40 + IA 10)

Course Outcomes

CO1: Demonstrate the use of PHP syntax, control structures, functions, strings, arrays, and exception handling in server-side programming.

CO2: Develop dynamic web applications by handling HTML forms, file management, and directory operations using PHP.

CO3: Implement session and cookie management techniques to maintain state and enhance web application security.

CO4: Apply database connectivity concepts with MySQL to perform CRUD operations and execute different types of joins in PHP-based applications.

Course Content:

UNIT I:	[15 hours]
Introduction to PHP: Server-Side Scripting – Advantages of PHP – Installation & Configuration. Evaluation of PHP, Basic Syntax, Defining variable and constant, PHP Data types, Operators and Expressions	
Decisions and Loop: Making Decisions, Doing Repetitive task with looping, Mixing Decisions and looping with HTML.	
Function: Define a function, Call by value and Call by reference, Recursive function.	
String: Creating and accessing, String Searching & Replacing String, Formatting String, String Related Library function	
Array: Anatomy of an Array, Creating index based and Associative array, Accessing array, Element Looping with Index based array, Looping, Some useful Library function	
Exception Handling: Understanding Exception and error, try, catch, throw. error tracking and debugging	
UNIT II:	[15 hours]
Handling HTML Form with PHP: Capturing Form, Data Dealing with Multi-value filed, and Generating File uploaded form, redirecting a form after submission.	
Working with file and Directories: Understanding file & directory, Opening and closing, a file, Copying, renaming and deleting a file, working with directories, Creating and deleting folder, File Uploading & Downloading.	
Session and Cookie: Introduction to Session Control, Session Functionality, Cookie, Setting Cookies with PHP. Using Cookies with Sessions, Modify Cookies, Deleting Cookies, Registering Session variables, Destroying the variables and Session.	
Database Connectivity with MySQL: Connection with MySQL Database, performing basic database operation (DML:Insert, Delete, Update, Select), Setting query parameter, Executing query Join (Cross joins, Inner joins, Outer Joins, Self joins.)	



Text Books
1. PHP and MySQL Web Development - Welling – (Pearson Education)-Fourth Edition
2. Learning PHP, MySQL, books by ‘ O’ riley Press
3. Core PHP Programming. Leon Atkinson, Prentice Hall.
4. Reference Books
5. Programming PHP. RasmusLerdorf, Kevin Tatroe. O'Reilly.
6. Learning PHP 5. David Sklar, O'Reilly.

Web Resources
1. www.w3schools.com
2. https://www.geeksforgeeks.org/web-technology/
3.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	3	-	3	-	-	2	-	3	2	2	-	3
CO2	2	3	3	3	-	-	2	-	3	3	3	-	2
CO3	2	3	3	3	-	-	2	-	2	3	3	-	3
CO4	2	3	3	3	-	-	2	-	2	3	3	-	2

**V SEMESTER**

Course Code: DSAI51	Course Title: ARTIFICIAL INTELLIGENCE
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 57	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcome

CO1: Understand the foundations of Artificial Intelligence, including intelligent agents, environments, and problem-solving approaches.
CO2: Apply advanced search strategies, adversarial game-playing algorithms, and evolutionary methods to solve well-defined problems.
CO3: Analyze logical reasoning techniques, planning methods, and uncertainty handling mechanisms to assess their effectiveness in decision-making.
CO4: Evaluate AI applications across domains such as machine learning, natural language processing, robotics, and expert systems, while reflecting on ethical and legal implications.

Content	Hours
Unit - 1 Introduction to AI	
What is AI? Intelligent Agents: Agents and environment, the concept of Rationality, the nature of environment, the structure of Agents. Knowledge-Based Agents: Introduction to Knowledge Based Agents, The Wumpus World as an Example World. Problem-solving: Problem-solving agents.	15
Unit - 2 Advanced Search Techniques	
Uninformed Search: DFS, BFS, Iterative Deepening Search. Informed Search: Best First Search, A* search, AO* search. Adversarial Search & Games: Two-player zero-sum games, Minimax Search, Alpha-Beta pruning. Constraints and Constraint Satisfaction Problems (CSPs), Backtracking search for CSP. Evolutionary Search Techniques: Introduction to evolutionary algorithms, Genetic algorithms, Applications of evolutionary search in AI.	15

Unit - 3 Logical Reasoning and Uncertainty
--



Logic: Propositional logic, First-order predicate logic, Propositional versus first-order inference, Unification and lifting. Inference: Forward chaining, Backward chaining, Resolution, Truth maintenance systems. Introduction to Planning: Blocks World problem, Strips; Handling Uncertainties: Non-monotonic reasoning, Probabilistic reasoning, Introduction to Fuzzy set theory.	15
Unit - 4 Domains and Applications of AI	
Domains in AI: Introduction to Machine Learning, Computer Vision, Robotics, Natural Language Processing, Deep Neural Networks, and their Applications. Expert Systems: The architecture and role of expert systems include two case studies. Legal and Ethical Issues: Concerns related to AI.	15

Text Book/References

1. Russell, S. and Norvig, P., “Artificial Intelligence - A Modern Approach”, 3rd edition, Prentice Hall
2. Nilsson Nils J, “Artificial Intelligence: A new Synthesis, Morgan Kaufmann Publishers Inc. San Francisco, CA, ISBN: 978-1-55-860467-4.
3. Dan W Patterson, “Introduction to Artificial Intelligence & Expert Systems”, PHI



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	3	2	-	-	-	2	-	3	2	3	-	2
CO2	3	3	3	2	-	-	2	-	3	3	3	-	2
CO3	3	3	2	2	-	-	2	-	3	2	3	-	2
CO4	3	3	3	3	2	3	3	2	3	3	3	3	3



Course Code: DSDA52	Course Title: Data Analytics
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcomes (COs):

- CO1:** Describe the evolution, types, and applications of data analytics in real-world scenarios.
- CO2:** Apply correlation and regression techniques to examine and predict relationships among variables.
- CO3:** Analyze statistical problems using probability distributions, estimation methods, and hypothesis testing.
- CO4:** Create and interpret interactive dashboards by transforming and visualizing data using Power BI.

Content	Hours
Unit - 1	
Introduction to Data Analytics 14 Hours Evolution of Data Analytics, Data Analytics Overview, Types of Data Analytics -Descriptive Analytics -Diagnostic Analytics - Predictive Analytics -Prescriptive Analytics, Importance and Benefits of Data Analytics. Different Applications of Analytics in Business, Text Analytics and Web Analytics, Skills for Business Analytics.	10
Unit - 2	
Correlation & Regression Correlation: Introduction, Meaning of Correlation, Types of correlation ,probable error, Karl pearson’s coefficient of correlation for individual series only, Spearman’s Rank correlation for individual series only. Regression: Introduction, definition, difference between correlation and regression, Simple linear regression, properties of regression coefficients, Regression equation x on y, Regression equation y on x , Simple Problems.	12

Unit - 3	
----------	--



Probability & Statistical Methods 14 Hours Sample Space, Types of Events, Measures of probability, conditional probability, Bayes' theorem, Random variable. Probability Distributions: Binomial, Poisson and Normal Distributions – Definitions, means, variances and applications of these distributions. Simple problems. Estimation and Hypothesis Testing- t-test, Analysis of variance (ANOVA) and Chi-square test.	12
Unit - 4	
Data Visualization: Introduction to data visualization, Visualization foundations. Power BI Introduction to Power BI -What is Business Intelligence (BI)?, Overview of Power BI and its components: Power BI Desktop, Service, and Mobile , Installing Power BI Desktop, Power BI Interface & Navigation, Data sources supported by Power BI. Data Loading & Transformation using Power Query: Understanding Power Query Editor, Loading data from various sources (Excel, Web, SQL), Cleaning and transforming data:Remove columns/rows, Rename columns, Merge and Append queries, Data types and formatting. Visualizations and Dashboards-Types of charts: Bar, Column, Pie, Line, Map, Matrix, Card, Gauge, etc., Creating interactive dashboards, Formatting visuals and tooltips, Using slicers and filters, Drill through and bookmarks.	11

Text Books:

1. Kumar, U.D. :Business Analytics – The Science of Data – Driven Decision Making, Wiley.
2. Dr Anil Maheshwari, Data Analytics Made Accessible, Publisher: Amazon.com Services LLC.
3. Johnson, R.A., Miller, I. and Freund, . :Probability and Statistics for Engineers, Pearson.

Reference Books:

1. Gert, H.N., Thorlund, L. and Thorlund, J :Business Analytics for Managers – Taking Business Intelligence Beyond Reporting, Wiley.
2. Data Analytics: Principles, Tools, and Practices: A Complete Guide for Advanced Data Analytics Using the Latest Trends, Tools, and Technologies by Dr. Gaurav Aroraa (Author), Chitra Lele (Author),



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	-	2	-	3	-	2	-	2	-	3
CO2	3	3	2	-	-	-	2	-	3	2	3	-	2
CO3	3	3	2	-	-	-	2	-	3	2	3	-	2
CO4	2	3	3	3	3	-	2	-	3	3	3	2	3



DSDA52P: Data Analytics Lab

Course outcome:

CO1: Apply probability distributions and statistical methods to analyze real-world datasets.

CO2: Perform hypothesis testing and regression analysis to derive meaningful insights from data.

CO3: Develop skills in data preprocessing, transformation, and integration using Power Query.

CO4: Create interactive dashboards and visualizations in Power BI for effective data-driven decision-making.

**1. From the given dataset players_info.csv ,**

- a) What is the probability distribution of genders among the players?
- b) What is the probability of each batting style?
- c) What is the probability of each bowling style?
- d) What is the probability distribution of player positions?
- e) What is the probability distribution of countries among the players?

2. 80% of all the visitors to Museum of Goa end up buying souvenirs from the souvenir shop at the museum. On the coming Sunday, if a random sample of 10 visitors is picked, Find the Probability that every visitor will end up buying from the souvenir shop. Find the Probability that a maximum of 7 visitors will buy souvenirs from the souvenir shop.

3. A testing agency wants to analyze the complexity of SAT exam 2022. They have collected the SAT scores of 1000 students in “sat_score.csv”. Calculate the probability that a student will score less than 800 in SAT exam. Calculate the probability that a student will score more than 1300 in SAT exam.

4. A Marketing services company reported that the typical American spends a mean of 144 minutes (2.4 hours) per day accessing the Internet via a mobile device. Select a sample of 30 friends and family members whose mobile access time is stored in a CSV file “InternetMobileTime.csv”. Is there evidence that the population mean time spent per day accessing the Internet via mobile device is different from 144 minutes? (Level of Significance $\alpha = 0.05$)

5. A hotel manager looks to enhance the initial expression that hotel guests have when they check in. Contributing to initial impressions is the time it takes to deliver a guest’s luggage to the room after check_in. A random sample of 20 deliveries on a particular day is selected in Wing A of the hotel and a random sample of 20 deliveries is selected in Wing B. The results are stored in “Luggage.csv”. Analyze the data and determine whether there is a difference between the mean delivery time in the 2 wings of the hotel.(Use $\alpha = 0.05$)

6. The file “Concrete.csv” contains the compressive strength in thousands of pounds/square inch, of 40 samples of concrete taken 2 and 7 days after pouring. At the 0.01 level of significance, is there evidence that the mean strength is lower at 2 days than at 7 days?

7. Two companies A and B were merged. After the first appraisal cycle post merger, employees originally belonging to company B have put an allegation that the management favours employees



who were originally a part of company A. At 95% confidence perform a hypothesis test to validate if the claim holds good. Promotion Status

Company P NP Total

8. Traffic management inspector in a city wants to understand whether carbon emissions from different cars are different. For this reason, the inspector has taken random samples from all registered cars on the road in that city and would like to test if the amount of carbon emission release depends on fuel type at 5% significance level. Dataset – AOVDData.csv

9. Find the relationship between the price of a laptop with other factors of the dataset “laptops.csv”.

Part - B

1. Install Power BI Desktop, explore the interface, and load data from an Excel file.
2. Use Power Query Editor to clean and transform a dataset.
3. Use Power Query Editor to combine data from multiple tables.
4. Visualize data using basic chart types - Bar Chart, Column Chart and Pie Chart by loading a dataset.
5. Use cards, maps, and matrices for advanced visuals by loading a dataset.
6. Enhance dashboard interactivity using slicers and filters.
7. Use drill-through pages and bookmarks for navigation and detail views.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	3	2	2	1	-	-	-	-	2	-	2	3
CO2	3	3	2	2	1	-	-	-	-	2	-	2	3
CO3	2	2	2	3	3	-	-	-	-	2	-	2	2
CO4	2	2	3	3	3	-	-	-	-	3	-	3	2



Course Code: DSWP53	Course Title: Web Programming
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcomes (COs):

CO1: Understand the fundamentals of the web, including internet concepts, browsers, servers, security, and the use of JavaScript with the Document Object Model (DOM).
CO2: Apply JavaScript and XML to create dynamic web documents, manipulate elements, and exchange data using style sheets and web services.
CO3: Develop interactive web applications using PHP constructs such as variables, operators, control flow, functions, and arrays.
CO4: Implement advanced PHP features including file handling, exception handling, cookies, and database connectivity to design robust and data-driven web applications.

Content	Hours
Unit - 1	
Fundamentals of Web: Internet – World Wide Web - Web Browsers - Web Servers – URLs – MIME – Internet Security - The Web Programmers Toolbox. Java Script and HTML Documents: The JavaScript execution environment - The Document Object Model - Element access in JavaScript - Events and event handling - Handling events from the Body elements, Button elements, Text box and Password elements - The DOM 2 event model - The navigator object - DOM tree traversal and modification.	15
Unit - 2	
Dynamic Documents with JavaScript: Introduction to dynamic documents - Positioning elements - Moving elements - Element visibility - Changing colours and fonts - Dynamic content - Stacking elements - Locating the mouse cursor - Reacting to a mouse click - Slow movement of elements - Dragging and dropping elements. XML: Introduction – Syntax - Document structure - Document Type definitions - Namespaces - XML schemas - Displaying raw XML documents - Displaying XML documents with CSS - XSLT style sheets - XML Processors - Web services.	15



Unit – 3	
Introduction to PHP: The Structure of PHP-Using Comments -Basic Syntax -Variables Operators -Variable Assignment -Multiple-Line Commands -Variable Typing -Constants Predefined Constants -The Difference Between the echo and print Commands -Functions Variable Scope, Expressions and Control Flow in PHP: Operators -Operator Precedence - Associativity Relational Operators - Conditionals: The if Statement -The else Statement - The elseif Statement -The switch Statement - The ? Operator - Looping: while Loops - do...while Loops for Loops - Breaking Out of a Loop-The continue Statement.	15
Unit - 4	
PHP Functions and Objects: PHP Functions - Defining a Function - Returning a Value Returning an Array - Do Not Pass Arguments by Reference - Returning Global Variables. PHP Arrays: Numerically Indexed Arrays - Associative Arrays - Assignment Using the array Keyword - The foreach...as Loop -Multidimensional Arrays - Using Array Functions-Date and Time Functions. File Handling: Checking Whether a File Exists - Creating a File - Reading from Files - Copying Files - Moving a File - Deleting a File - Updating Files - Locking Files for - Multiple Accesses Reading an Entire File - Uploading Files. Exception Handling, Cookies and connecting to database	15

TEXT BOOK
1. Robert W Sebesta, “Programming the World Wide Web”, 4th Edition, Pearson Education, 2008.
2. Learning PHP, MySQL & JavaScript With jQuery, CSS & HTML5 by Robin Nixon, Published by O’Reilly 2015.

Reference Books
1. M.Deitel, P.J.Deitel, A.B.Goldberg, “Internet & World Wide Web How to program”, 3rd Edition, Pearson Education / PHI, 2004.
2. Chris Bates, “Web Programming Building Internet Applications”, 3rd Edition, Wiley India, 2006.
3. Xue Bai et al, “The Web Warrior Guide to Web Programming”, Thomson, 2003.
4. PHP A Beginner’s Guide by VikramVaswani, by The McGraw-Hill, 2009



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	-	-	1	2	-	-	3	2	-	-	-
CO2	2	3	3	2	1	-	-	-	3	3	2	-	-
CO3	2	3	3	2	1	-	-	2	3	3	2	1	-
CO4	2	3	3	3	1	1	-	3	3	3	2	1	2

**DSWP53P: WEB PROGRAMMING LAB****Course Outcomes (COs):**

CO1: Apply HTML and JavaScript to design and validate interactive web forms.
CO2: Develop client-side programs using JavaScript functions, arrays, and dynamic effects.
CO3: Implement PHP programs for solving basic computational problems and file handling.
CO4: Design simple database-driven applications in PHP using data storage, updating, deletion, cookies, and exception handling.

1. Create a form with the elements of Textboxes, Radio buttons, Checkboxes, and so on. Write JavaScript code to validate the format in email, and mobile number in 10 characters, If a textbox has been left empty, popup an alert indicating when email, mobile number and textbox has been left empty.
2. Develop an HTML Form, which accepts any Mathematical expression. Write JavaScript code to Evaluate the expression and Display the result.
3. Create a page with dynamic effects. Write the code to include layers and basic animation.
4. Write a JavaScript code to find the sum of N natural Numbers. (Use userdefined function)
5. Write a JavaScript code block using arrays and generate the current date in words, this should include the day, month and year.
6. Create a form for Student information. Write JavaScript code to find Total, Average, Result and Grade.
7. Create a form for Employee information. Write JavaScript code to find DA, HRA, PF, TAX, Gross pay, Deduction and Net pay.
8. Write a program in PHP to change background color based on day of the week using if else if statements and using arrays.
9. Write a simple program in PHP for i) generating Prime number ii) generate Fibonacci series.
10. Write a PHP program to remove duplicates from a sorted list
11. Write a PHP Script to print the following pattern on the Screen:

```
*****  
*****  
***  
**  
*
```



12. Write a simple program in PHP for Searching of data by different criteria
13. Write a function in PHP to generate captcha code
14. Write a Program to store and read image from Database.
15. Write a program in PHP to read and write file using form control.
16. Write a program in PHP to add, update and delete using student database.
17. Write a program in PHP to Validate Input
18. Write a program in PHP for setting and retrieving a cookie
19. Write a PHP program to Create a simple webpage of a college.
20. Write a program in PHP for exception handling for i) divide by zero ii) checking date format.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	2	1	-	-	1	-	3	2	2	-	1
CO2	2	3	2	2	-	-	1	-	2	3	2	-	2
CO3	2	3	3	2	1	1	1	-	3	3	3	1	2
CO4	3	3	3	3	2	2	2	2	3	3	3	2	3



Course Code: DSEDM54	Course Title: Data Mining
Course Credits: 03	Hours/Week: 03
Total Contact Hours: 60	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcomes (COs):

CO1: Understand fundamental concepts of data mining, including classification, regression, clustering, summarization, association rules, and decision trees, along with their applications and social implications.
CO2: Apply statistical, distance-based, and decision-tree algorithms such as regression, Bayesian classification, k-Nearest Neighbors, ID3, C4.5, and CART to solve data mining problems.
CO3: Analyze clustering techniques including hierarchical, partitional, and similarity-based approaches, and evaluate outliers in large datasets..
CO4: Design and implement association rule mining using Apriori, sampling, partitioning, and parallel/distributed algorithms to discover useful patterns and knowledge from data.

Content	Hours
Unit - 1	
Basic Data Mining Tasks: [Some things on data warehousing, ETL Tools] Classification – Regression - Time Series Analysis- Prediction - Clustering – Summarization - Association Rules - Sequence Discovery. Data Mining Versus Knowledge Discovery in Databases - The Development of Data Mining - Data Mining Issues - Data Mining Metrics - Social Implications of Data Mining - Data Mining from a Database Perspective. Data Mining Techniques: Statistical Perspective on Data Mining - Similarity Measures - Decision Trees.	12
Unit - 2	
Introduction - Statistical-Based Algorithms: Regression – Bayesian Classification. Distance- Based Algorithms: Simple Approach - K Nearest Neighbors. Decision Tree-Based Algorithms: ID3 – C4.5 - CART - Scalable DT techniques.	12

Unit - 3



Introduction - Similarity and Distance Measures – Outliers. Hierarchical Algorithms: Agglomerative Algorithms - Divisive Clustering. Partitional Algorithms: Minimum Spanning Tree - Squared Error Clustering Algorithm - K -Means Clustering - Nearest Neighbor Algorithm.	12
Unit - 4	
Introduction – Large Itemsets – Basic Algorithms: Apriori Algorithm – Sampling algorithm – Partitioning. Parallel and Distributed algorithms (be specific): Data Parallelism – Task Parallelism. Comparing approaches – Incremental rules.	12

Text Book:

1. Margaret H Dunham, “Data Mining Introductory and Advanced Topics”, Pearson Education, 2012

References:

1. Jiawei Han and Micheline Kamber, “Data Mining - Concepts and Techniques”, Third Edition, Elsevier, 2012
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison Wesley, Second edition, 2018.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	1	-	-	2	1	-	3	2	2	1	2
CO2	3	3	3	2	-	-	-	2	3	3	3	1	2
CO3	2	3	3	2	-	-	1	2	3	3	3	1	2
CO4	2	3	3	3	1	-	1	3	3	3	3	2	2



Course Code: SEC51	Course Title: Cyber Crimes, Cyber Laws and Intellectual Property Rights
Course Credits: 03	Hours/Week: 03
Total Contact Hours: 48	Formative Assessment Marks: 25
Exam Marks: 25	Exam Duration: 1 hrs

Course Outcomes (COs):

CO1: Identify various types of cybercrimes and apply relevant provisions of the IT Act to real-world cases.
CO2: Demonstrate procedures for reporting cybercrimes both online and offline, and analyze phishing attempts for preventive action.
CO3: Configure privacy, security, and payment settings on social media and e-commerce platforms, and evaluate mechanisms for reporting misuse.
CO4: Examine intellectual property rights (IPR) issues in cyberspace and interpret legal procedures for patents, copyrights, trademarks, and geographical indications.

Content	Hours
Unit – 1	
Cybercrime & laws - Identify types of cybercrimes Prepare checklist for reporting cyber-crime at Cybercrime Police Station. Prepare checklist for reporting cyber-crime online. Identify phishing emails, Analyze cybercrime cases and identify section applicable (as per IT Act), Discuss Data protection laws in India	12
Unit – 2	
Social media and E-commerce Security - Basic checklist, privacy and security settings for popular social media platforms. Reporting and redressal mechanism for violations and misuse of social media platforms. Configure security settings in Mobile Wallets and UPIs. Prepare checklist for secure net banking.	12



Unit – 3	
Digital Devices Security, Tools and Technologies for Cyber Security - Setting, configuring and managing three password policy in the computer (BIOS, Administrator and Standard User). Setting and configuring two factor authentication in the Mobile phone. Security patch management and updates in Computers and Mobiles. Managing Application permissions in Mobile phone. Installation and configuration of computer Anti-virus. Wi-Fi security management in computer and mobile.	12
Unit – 4	
IPR - PR issues in Cyber Space ,Identify liabilities in case of infringement of copyrights/trademarks/patents using Cases , Procedure for registration of patents, copyrights, trademarks and GI , Recognize Geographical Indicators and their significance , Traditional knowledge and IPR., Discuss Landmark judgements on trademark and domain names issues	12

References

1. Sunit Belapure and Nina Godbole, “Cyber Security: Understanding Cyber Crimes, Computer Forensics And Legal Perspectives”, Wiley India Pvt Ltd, ISBN: 978-81- 265-21791, Publish Date 2013
2. Dr. Surya Prakash Tripathi, Ritendra Goyal, Praveen Kumar Shukla, KLSI. “Introduction to information security and cyber laws”. Dreamtech Pre ss. ISBN: 9789351194736, 2015
3. Duggal Pavan, Legal Framework on Electronic Commerce and Intellectual Property Rights in Cyberspace Hardcover – 2014



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	2	3	-	1	-	3	-	-	3	-	3	2	-
CO2	2	3	-	-	2	3	-	-	3	-	3	2	-
CO3	-	2	-	-	1	3	3	-	-	-	2	3	3
CO4	-	2	-	-	2	3	3	-	-	-	2	3	3



Course Code: VOCQT55	Course Title: Quantitative Techniques
Course Credits: 03	Hours/Week: 03
Total Contact Hours: 48	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcomes (Cos):

CO1: Understand the quantitative approaches to decision-making and general methods for solving quantitative models.
CO2: Calculate the minimum transportation cost and optimize resource allocation.
CO3: Apply game theory and decision-making under uncertainty to formulate and solve strategic problems.
CO4: Understand and apply queuing theory and simulation to solve real-world problems in various business domains

Content	Hours
Unit – 1	
Numbers Property – Simplification – Divisibility – HCF and LCM – Decimal Fractions – Square roots and Cube Roots – Logarithms – Antilogarithms - Surds and indices - Permutation and Combination – Probability – Odd man out series - Number series - letter series – codes – Relationships – classification.	12
Unit – 2	
Time and work – Problems on Ages – Calendar – Clock – Pipes and Cistern – Time and Distance – Problems of Train – Boats and Streams. Area – Volume and surface Areas – Heights and Distances – Data Interpretation: Tabulation – Bar Graphs – Pie Charts – Line Graphs. Data Interpretation - Sources, acquisition and interpretation of data; Quantitative and qualitative data; Graphical representation and mapping of data.	12



Unit – 3	
Simple Interest – Compound Interest – Stocks and Shares – True Discount – Banker's discount. Averages – Percentage – Profit and Loss - Ratio and Proposition – Partnership – Allegation and mixture – Chain rule. Understanding the structure of arguments; Evaluating and distinguishing deductive and inductive reasoning; Verbal analogies: Word analogy Applied analogy; Verbal classification; Reasoning Logical Diagrams: Simple diagrammatic relationship, multi diagrammatic relationship; Venn diagram; Analytical Reasoning.	12
Unit – 4	
Teaching: Nature, objectives, characteristics and basic requirements; Learner's characteristics; Factors affecting teaching; Methods of teaching; Teaching aids; Evaluation systems. Research Aptitude: Meaning, characteristics and types; Steps of research; Methods of research; Research Ethics; Paper, article, workshop, seminar, conference and symposium; Thesis writing: its characteristics and format. Reading Comprehension: A passage to be set with questions to be answered. Communication: Nature, characteristics, types, barriers and effective classroom communication.	12

Reference
1. R.S. Aggarwal, Quantitative Aptitude, S. Chand & Company, New Delhi, 2012
2. Govind Prasad Singh and Rakesh Kumar, Text Book of Quickest Mathematics (for all Competitive Examinations), Kiran Prakashan, 2012.
3.R.S. Aggarwal, Objective Arithmetic, S. Chand & Company, New Delhi, 2005.
4. Dr. Lal, Jain, Dr. K. C. Vashistha, “U.G.C.- NET/JRF/SET Teaching & Research Aptitude”, Upkar Prakashan, 2010.
5. “UGC NET/SLET: Teaching & Research Aptitude”, Bright Publications, 2010



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	-	-	-	-	-	3	-	2	-	-
CO2	2	3	-	-	-	-	-	1	2	-	3	-	-
CO3	2	3	-	-	-	1	1	2	2	-	3	-	2
CO4	2	3	-	-	-	1	1	2	2	-	3	-	3

**SEMESTER VI**

Course Code: DSML61	Course Title: Machine Learning
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 48	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcomes (COs):

CO1. Understand the fundamentals of machine learning, its types, challenges, applications, and the role of Python libraries and tools.
CO2. Apply data preparation techniques such as visualization, preprocessing, feature selection, and model training for machine learning tasks.
CO3. Analyze supervised learning methods including classification, regression, k-Nearest Neighbors, linear models, Naive Bayes, and decision trees to evaluate their performance on real datasets.
CO4. Design and implement unsupervised learning algorithms such as K-Means, DBSCAN, and clustering-based techniques for segmentation, preprocessing, and semi-supervised learning applications.

Content	Hours
Unit – 1	
Fundamentals of Machine Learning Introduction to Machine Learning: What is Machine Learning? Why Use Machine Learning? Types of Machine Learning Systems, Main Challenges of Machine Learning, Applications of Machine Learning. Why Python, scikit-learn, Essential Libraries and Tools.	12
Unit – 2	
Data Preparation: Working with Real Data, look at the Big Picture, Get the Data, Discover and Visualize the Data to Gain Insights, Prepare the Data for Machine Learning Algorithms, Select and Train a Model.	12
Unit – 3	



Supervised Learning Classification and Regression, Some Sample Datasets, k-Nearest Neighbours, Linear Models, Naive Bayes Classifiers, Decision Trees.	12
--	----

Unit – 4	
Unsupervised Learning Clustering, K-Means, Limits of K-Means, using clustering for image segmentation, Using Clustering for Preprocessing, Using Clustering for Semi- Supervised Learning, DBSCAN, Other Clustering Algorithms.	12

Textbook:

1. Andreas . C. Müller and S. Guido, "Introduction to Machine Learning with Python," O'Reilly, 2017 (Unit-1)
2. Amanda . Casari and Alice . Zheng, "Feature Engineering for Machine Learning," O'Reilly Media, Inc., 2018, p. 218. (Unit-2)
3. A. Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow," O'Reilly Media, Inc., 2022.
4. Ian Goodfellow, Yoshua. Bengio, and Aaron. Courville, "Deep Learning," MIT Press, 2016. (Unit-4)

Reference Books

1. S. Rashka and V. Mirdzhalili, "Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2," Packt, Birmingham and Mumbai, 2020.
2. S. Shalev-Shwartz and S. Ben-David, "Understanding Machine Learning: From Theory to Algorithms," Cambridge University Press, 2014.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	-	1	2	1	-	3	2	-	-	2
CO2	3	3	3	2	-	-	-	2	3	3	2	-	-
CO3	2	3	3	2	1	-	1	2	3	3	3	-	-
CO4	2	3	3	3	1	-	1	3	3	3	3	2	3



DSML61P: Machine Learning Lab

Course Outcomes (COs):

CO1: Apply Python libraries such as NumPy, pandas, and Matplotlib to load, explore, and visualize datasets.
CO2: Preprocess datasets by handling missing values, encoding categorical variables, and applying feature scaling techniques.
CO3: Implement supervised learning algorithms such as k-Nearest Neighbors, Linear Regression, and Decision Trees using scikit-learn.
CO4: Apply unsupervised learning techniques such as K-Means clustering and evaluate results through visualization.

List of Programs

1. Install and set up Python and essential libraries like NumPy and pandas.
2. Introduce scikit-learn as a machine learning library.
3. Install and set up scikit-learn and other necessary tools.
4. Write a program to Load and explore the dataset of .CVS and excel files using pandas.
5. Write a program to Visualize the dataset to gain insights using Matplotlib or Seaborn by plotting scatter plots, bar charts.
6. Write a program to Handle missing data, encode categorical variables, and perform feature scaling.
7. Write a program to implement a k-Nearest Neighbours (k-NN) classifier using scikitlearn and Train the classifier on the dataset and evaluate its performance.
8. Write a program to implement a linear regression model for regression tasks and Train the model on a dataset with continuous target variables.
9. Write a program to implement a decision tree classifier using scikit-learn and visualize the decision tree and understand its splits.
10. Write a program to Implement K-Means clustering and Visualize clusters.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	2	2	-	-	1	-	3	2	2	-	1
CO2	3	3	2	2	-	1	2	-	3	3	3	1	2
CO3	3	3	3	3	1	1	2	2	3	3	3	1	3
CO4	3	3	3	3	1	1	2	3	3	3	3	2	3



Course Code: DSMAD62	Course Title: Mobile Application Development
Course Credits: 04	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcomes (COs):

CO1: Understand the fundamentals of mobile technologies, Android architecture, application components, and development environment.
CO2: Apply Android activities, fragments, intents, and user interface components to design interactive mobile applications.
CO3: Analyze and implement data storage options, content providers, and device features for enhanced mobile app functionality.
CO 4: Evaluate and prepare Android applications for deployment by integrating web services, ensuring performance, and adhering to publishing standards.

Content	Hours
Unit - 1	
Introduction : Brief History of mobile technologies, Different mobile technologies Key Mobile Application Services-Introducing Android, The Android Application Components, Exploring the Development Environment, -Obtaining the Required Tools Launching Your First Android Application-Exploring the IDE-Debugging Your Application-Publishing Your Application Using Activities - Fragments and Intents in Android : Working with activities, Using Intents, Fragments, Using the Intent Object to Invoke Built-in Application	15
Unit - 2	



Working with the User Interface Using views Understanding the Components of a Screen- Adapting to Display Orientation-Managing Changes to Screen Orientation- Utilizing the Action Bar-Creating the User Interface Programmatically Listening for UI Notification Using Basic Views-Using Picker Views -Using List Views to Display Long Lists Understanding Specialized Fragments - Using Image Views to Display Pictures -Using Menus with Views Using WebView- Saving and Loading User Preferences-Persisting Data to Files-Creating and Using Databases.	15
--	----

Unit – 3	
Designing User interface Designing by declaration, creating the opening screen, using alternate resources, implementing an about box, applying a theme, adding a menu, adding settings, debugging with log messages, debugging with debugger	15
Unit - 4	
Creating Your Own Content Providers -Using the Content Provider, SMS Messaging - Sending Email-Displaying Maps- Getting Location Data- Monitoring a Location Putting SQL to work Introducing SQLite, In and Out of SQLite, Hello Database, Data Binding, using content provider, implementing content provider. Reading/writing local data, Accessing the Internal File system, Accessing the SD card. Preparing app for publishing, Deploying APK files, uploading in Market, Consuming Web Services Using HTTP-Consuming JSON Services- Creating Your Own Services Binding Activities to Services -Understanding Threading.	15

Text Books:

- 1..Wei-Meng Lee, *Beginning android 4 application Development*, John Wiley & sons, Inc, 2012.
2. Jerome DiMarzio, “*Beginning Android Programming with Android Studio*”, 4th Edition

Reference Books:

1. Grant Allen, *Beginning Android 4*, Apress,2012.
2. Pradeep Kothari, “*Android Application Development (With Kitkat Support)*”, Black Book 2014



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	3	2	2	2	-	-	2	-	3	2	2	-	2
CO2	3	3	3	3	2	-	2	2	3	3	3	2	2
CO3	3	3	3	3	-	-	2	2	3	3	3	2	2
CO4	3	2	3	3	2	2	3	3	3	3	3	3	3



DSMAD62P: Mobile Application Development Lab

CO1: Demonstrate the use of UI controls and views in basic Android applications.

CO2: Develop applications that manage user interaction through activities, intents, menus, and screen orientation.

CO3: Apply techniques for storing, reading, and writing data using local files and SQLite databases.

CO4: Implement Android applications that utilize device services such as SMS, email, and location-based features, and deploy them effectively.

- 1 Creating “Hello world” Application.
- 2 Creating an application that displays message based on the screen orientation.
- 3 Create an application to develop Login window using UI controls.
- 4 Create an application to implement new activity using explicit intent, implicit intent and content provider.
- 5 Create an application that displays custom designed Opening Screen.
- 6 Create an UI with all views.
- 8 Create menu in Application
- 9 Read/ write the Local data.
- 10 Create / Read / Write data with database (SQLite).
- 11 Create an application to send SMS and receive SMS
- 12 Create an application to send an e-mail.
- 13 Display Map based on the Current/given location.
- 14 Create a sample application with login module(check user name and password)
On successful login change Textview “Login Successful”. On login fail alert using Toast “login fail”
- 15 Learn to deploy Android applications.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5
CO1	2	2	3	3	-	-	2	2	2	3	2	-	2
CO2	2	3	3	3	-	-	2	2	2	3	3	-	2
CO3	2	3	3	3	-	-	2	2	2	3	3	-	2
CO4	2	3	3	3	-	-	2	3	2	3	3	-	2



Course Code: VOCECD64	Course Title: Electronic Content Design
Course Credits: 04	Hours/Week: 03
Total Contact Hours: 48	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcome:

CO1: Explain the fundamentals of e-learning, learning management systems, and emerging trends in digital education.
CO2: Apply instructional models like ADDIE and ASSURE to design and develop reusable e-content using appropriate authoring tools.
CO3: Analyze principles of visual design, user experience (UX), and multimedia content creation for effective digital learning materials.
CO4: Demonstrate skills in HTML, CSS, CMS platforms, social media content planning, SEO, and ethical practices in digital content development.

Content	Hours
Unit - 1	
Introduction to E-learning- Definition, history, benefits and drawbacks of online learning, best practices of online learning, future of e learning. Overview of LMS. Technologies used in e-learning, Online course, tools to create an online course, need of the millennial learners, 21st century skills and E-learning trends.	12
Unit - 2	
E-content. Designing and Development of E-content. Standards of E-content. Learning Objects and Re-usability of E-content. Phases of e-content development, various instructional models- ADDIE and ASSURE instructional model. An overview of Content Authoring Tools.	12
Unit - 3	



Principles of Visual Design- Visual hierarchy, typography, and colour theory, User Experience (UX) Design-UX principles, wireframing, and prototyping, Graphic Design for Digital Media-Image editing, resolution, and file formats.	12
--	----

Unit - 4	
HTML and CSS basics, designing for different screen sizes and devices, Multimedia Content Creation- Creating multimedia elements (images, audio, video), Social media platforms and content planning, Develop a social media content calendar, Content Management Systems(CMS),introduction to CMS platforms (e.g., WordPress), search Engine Optimization (SEO),SEO basics and best practices, Copyright and Ethics in Digital Content, Copyright laws, fair use, and ethical considerations, Content Evaluation and Feedback, Usability testing and feedback collection	12

Books and References
1 Diane Elkins et al , E-Learning Fundamentals: A PRACTICAL GUIDE, . (2015). ISBN: 9781562869472, Pages: 176.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	3	-	3	-	-	2	-	3	2	2	-	3
CO2	2	3	-	3	-	-	2	-	3	2	2	-	2
CO3	2	3	3	3	-	-	2	-	2	3	3	-	3
CO4	2	3	3	3	-	-	2	-	2	3	3	-	2



Course Code: DSEST63	Course Title: Software Testing
Course Credits: 04	Hours/Week: 03
Total Contact Hours: 48	Formative Assessment Marks: 40
Exam Marks: 60	Exam Duration: 2 & ½ Hours

Course Outcomes (COs):

CO1: Describe fundamental software testing concepts, life cycle, and techniques such as boundary value, robustness, and random testing.
CO2: Apply equivalence class testing, decision table testing, and data flow testing to design effective test cases for given problems.
CO3: Analyze system and integration testing strategies, including thread, interaction, and client-server testing, to evaluate software quality.
CO4: Demonstrate object-oriented and GUI testing techniques to validate software functionality.

Content	Hours
Unit – 1	
Introduction: Basic definitions, A testing life cycle, Test Cases, Fundamental approaches to apply Test Cases, Levels of Testing, Examples: The NextDate function, Triangle problem and The Commission Problem and The ATM (Simple Automatic Teller Machine) problem. Boundary Value Testing: Generalizing Boundary Value Analysis, Limitations of Boundary Value Analysis, Robustness Testing, Worst-Case Testing, Special Value Testing, Test cases for the Triangle problem, Test cases for the NextDate function, Test cases for the Commission Problem, Random Testing and Guidelines for Boundary Value Testing.	12
Unit – 2	
Equivalence Class Testing: Equivalence Classes, Weak Normal Vs Strong Normal Equivalence Class Testing, Weak Robust Vs Strong Robust Equivalence Class Testing, Equivalence Class Test Cases for the Triangle Problem, Equivalence Class Test Cases for the Next Date Function and Equivalence Class Test Cases for the Commission Problem, Guidelines for Equivalence Class Testing. Decision Table Based Testing: Decision tables, Test cases for the triangle problem, Test cases for the Next Date function, Test cases for the commission problem, Guidelines and observations. Data flow Testing: Definition Use Testing, Example- The Commission Problem, Slice-Based Testing, Guidelines and Observations.	12



Unit – 3	
Levels of Testing: The SATM System, Structural and Behavioural Insights. Integration Testing: A Closer Look at the SATM System, Decomposition-Based Integration, Top Down Vs Bottom-Up Integration, Sandwich Integration, Call Graph-Based Integration, Pair wise Integration, Neighborhood Integration, Path-Based Integration. System Testing: Threads, Basic concepts for requirements specification, Finding threads, Structural strategies and functional strategies for thread testing, Interaction Testing: A Taxonomy of Interactions, Static Interaction in a Single Processor, Static Interaction in Multiple Processors, Dynamic Interaction in a Single Processor, Dynamic Interaction in Multiple Processors, Client-Server Testing.	12
Unit – 4	
Object Oriented Testing: Issues in Object Oriented Testing, Implication of Composition and Encapsulation, Implications of Inheritance, Implications of Polymorphism, GUI Testing, Object-Oriented Integration Testing. Exploratory Testing: The context-driven school, Exploring exploratory testing, Exploring a familiar example, Exploratory and context-driven testing observations. Model-Based Testing: Testing based on models, Appropriate models, Use case-based testing, Commercial tool support for model-based testing. Test-Driven Development: Test-then-code cycles, Automated test execution, Java and JUnit example, Remaining questions, Pros, cons, and open questions of TDD, Retrospective on MDD versus TDD, Software Testing Excellence: Craftmanship, Best practice of software testing, Top 10 best practices for software testing excellence.	12

Text Book
Paul C. Jorgensen: Software Testing, A Craftsman’s Approach, 3rd Edition, 2013.
Reference Books
Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, 1st edition, John Wiley & Sons, 2011.
2. Brian Marrick: The Craft of Software Testing, 1st edition, Pearson, 2012.
3. Srinivasan Desikan, Gopalaswamy Ramesh: Software testing Principles and Practices, 1 st Edition, Pearson, 2012.
4. Aditya P Mathur: Foundations of Software Testing, Pearson, 2008.



Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (PO1-8) and Program specific outcome(PSOs1-5)

Course Outcome (COs)	Program Outcomes (POs)								Program Specific Outcomes (PSOs)				
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	-	-	2	2	-	2	-	2	-	2
CO2	3	3	2	-	-	2	2	-	3	2	3	-	2
CO3	3	3	3	2	-	2	2	-	3	2	3	-	3
CO4	2	3	3	3	2	2	2	-	3	3	3	2	2