Department of Computer and System Sciences
Siksha Bhavana, Visva-Bharati
Santiniketan - 731235

B.Sc. (Computer Science) Course Structure
(Under National Education Policy - 2020)

(From 2023-2024 Academic Session)

SEMESTER - I

Type	Subject Code	Subject	Credit	Hours
Major	MJCS11T MJCS11P	Introduction to Programming (Theory) Introduction to Programming (Lab)	3 1	45 30
	MJCS12T MJCS12P	Digital Electronics (Theory) Digital Electronics (Lab)	3 1	45 30
Minor	MNCS01T MNCS01P	Basics of Programming (Theory) Basics of Programming (Lab)	3 1	45 30
Multi	MDCST	Fundamentals of Computer Science (Theory)	3	45
SEC	SECCS11P	UNIX/LINUX Programming (Lab)	3	90
AECC		English/Modern Indian Language1	2	30
CVAC		Tagore Studies	3	45
		Total	23	435

SEMESTER - II

Type	Subject Code	Subject	Credit	Hours
Major	MJCS21T MJCS21P	Data Structures and Algorithms (Theory) Data Structures and Algorithms (Lab)	3 1	45 30
	MJCS22T	Discrete Structures (Theory)	4	60
Minor	MNCS01T MNCS01P	Basics of Programming (Theory) Basics of Programming (Lab)	3 1	45 30
Multi	MDCST	Fundamentals of Computer Science (Theory)	3	45
SEC	SECCS21P	Programming in Python (Lab)	3	90
AECC		English/Modern Indian Language2	2	30
CVAC		Environmental Studies	3	45
		Total	23	420

Year I	Summer Internship	4	120

SEMESTER - III

Type	Subject Code	Subject	Credit	Hours
Major	MJCS31T MJCS31P	Design and Analysis of Algorithms (Theory) Design and Analysis of Algorithms (Lab)	3 1	45 30
	MJCS32T MJCS32P	Computer Organization and Architecture (Theory) Computer Organization and Architecture (Lab)	3 1	45 30
Minor	MNCS02T MNCS02P	Data Structures (Theory) Data Structures (Lab)	3 1	45 30
Multi	MDCST	Fundamentals of Computer Science (Theory)	3	45
SEC	SECCS31P	Web Programming (Lab)	3	90
AECC		Modern Indian Language/English1	2	30
		Total	20	390

SEMESTER - IV

Type	Subject Code	Subject	Credit	Hours
Major	MJCS41T MJCS41P	Object-Oriented Programming (Theory) Object-Oriented Programming (Lab)	3 1	45 30
	MJCS42T MJCS42P	Software Engineering (Theory) Software Engineering (Lab)	3 1	45 30
	MJCS43T MJCS43P	Operating Systems (Theory) Operating Systems (Lab)	3 1	45 30
	MJCS44T MJCS44P	Computational Statistics (Theory) Computational Statistics (Lab)	3 1	45 30
Minor	MNCS02T MNCS02P	Data Structures (Theory) Data Structures (Lab)	3 1	45 30
AECC		Modern Indian Language/English2	2	30
		Total	**22**	**405**

Year II	**Summer Internship**	**4**	**120**

SEMESTER - V

Type	Subject Code	Subject	Credit	Hours
Major	MJCS51T MJCS51P	Database Management Systems (Theory) Database Management Systems (Lab)	3 1	45 30
	MJCS52T	Theory of Computation (Theory)	4	60
	MJCS53T MJCS53P	Digital Communication (Theory) Digital Communication (Lab)	3 1	45 30
Minor	MNCS03T MNCS03P	Programming in JAVA (Theory) Programming in JAVA (Lab)	3 1	45 30
		Total	**16**	**285**

SEMESTER - VI

Type	Subject Code	Subject	Credit	Hours
Major	MJCS61T MJCS61P	Computer Graphics (Theory) Computer Graphics (Lab)	3 1	45 30
	MJCS62T MJCS62P	Computer Networks (Theory) Computer Networks (Lab)	3 1	45 30
	MJCS63T MJCS63P	Compiler Construction (Theory) Compiler Construction (Lab)	3 1	45 30
Minor	MNCS03T MNCS03P	Programming in JAVA (Theory) Programming in JAVA (Lab)	3 1	45 30
		Total	16	300

Year III	Summer Internship	4	120

SEMESTER - VII

Type	Subject Code	Subject	Credit	Hours
Major	MJCS71T MJCS71P	Advanced Algorithms (Theory) Advanced Algorithms (Lab)	3 1	45 30
	MJCS72T MJCS72P	Artificial Intelligence (Theory) Artificial Intelligence (Lab)	3 1	45 30
	MJCS73T MJCS73P	Soft Computing (Theory) Soft Computing (Lab)	3 1	45 30
Minor	MNCS04T MNCS04P	Basics of Python Programming (Theory) Basics of Python Programming (Lab)	3 1	45 30
Research/ Course in lieu of Research	CSR71	Research Project I	4	75
	CSEC7XYT CSEC7XYP	Elective I (Theory) Elective I (Lab)	3 1	45 30
		Total	20	375

SEMESTER - VIII

Type	Subject Code	Subject	Credit	Hours
Major	MJCS81T MJCS81P	Image Processing (Theory) Image Processing (Lab)	3 1	45 30
	MJCS82T MJCS82P	Machine Learning (Theory) Machine Learning (Lab)	3 1	45 30
Minor	MNCS04T MNCS04P	Basics of Python Programming (Theory) Basics of Python Programming (Lab)	3 1	45 30
Research /Course in lieu of Research	CSR81	Research Project II	8	150
	CSEC8AYT CSEC8AYP	Elective II (Theory) Elective II (Lab)	3 1	45 30
	CSEC8BZT CSEC8BZP	Elective III (Theory) Elective III (Lab)	3 1	45 30
		Total	20	375

Elective Courses in lieu of Research - 4 Credits (3T+1P)

I. List for Elective I (Choose any one)

Type	Subject Code	Subject	Credit	Hours
Course in lieu of Research	CSEC700T	Cyber Security (Theory)	3	45
	CSEC700P	Cyber Security (Lab)	1	30
	CSEC701T	Human-Computer Interaction (Theory)	3	45
	CSEC701P	Human-Computer Interaction (Lab)	1	30
	CSEC702T	Imprecise Mathematics (Theory)	3	45
	CSEC702P	Imprecise Mathematics (Lab)	1	30
	CSEC703T	Information Theory (Theory)	3	45
	CSEC703P	Information Theory (Lab)	1	30
	CSEC704T	Internet of Things (Theory)	3	45
	CSEC704P	Internet of Things (Lab)	1	30
	CSEC705T	Modelling and Simulation (Theory)	3	45
	CSEC705P	Modelling and Simulation (Lab)	1	30
	CSEC706T	Numerical Methods (Theory)	3	45
	CSEC706P	Numerical Methods (Lab)	1	30
	CSEC707T	Operations Research (Theory)	3	45
	CSEC707P	Operations Research (Lab)	1	30
	CSEC708T	Quantum Computation and Quantum Information (Theory)	3	45
	CSEC708P	Quantum Computation and Quantum Information (Lab)	1	30
	CSEC709T	VLSI Design (Theory)	3	45
	CSEC709P	VLSI Design (Lab)	1	30

II. List for Elective II (Choose any one)

Type	Subject Code	Subject	Credit	Hours
Course in lieu of Research	CSEC810T	Algorithmic Graph Theory (Theory)	3	45
	CSEC810P	Algorithmic Graph Theory (Lab)	1	30
	CSEC811T	Android Programming (Theory)	3	45
	CSEC811P	Android Programming (Lab)	1	30
	CSEC812T	Bioinformatics (Theory)	3	45
	CSEC812P	Bioinformatics (Lab)	1	30
	CSEC813T	Cloud Computing (Theory)	3	45
	CSEC813P	Cloud Computing (Lab)	1	30
	CSEC810T	Cryptography and Network Security (Theory)	3	45
	CSEC810P	Cryptography and Network Security (Lab)	1	30
	CSEC815T	Introduction to Data Sciences (Theory)	3	45
	CSEC815P	Introduction to Data Sciences (Lab)	1	30
	CSEC816T	Parallel Algorithms (Theory)	3	45
	CSEC816P	Parallel Algorithms (Lab)	1	30
	CSEC817T	Pattern Recognition (Theory)	3	45
	CSEC817P	Pattern Recognition (Lab)	1	30
	CSEC818T	Randomized Algorithms (Theory)	3	45
	CSEC818P	Randomized Algorithms (Lab)	1	30
	CSEC819T	Wireless Networks (Theory)	3	45
	CSEC819P	Wireless Networks (Lab)	1	30

III. List for Elective III (Choose any one)

Type	Subject Code	Subject	Credit	Hours
Course in lieu of Research	CSEC820T	Advanced Graphics (Theory)	3	45
	CSEC820P	Advanced Graphics (Lab)	1	30
	CSEC821T	Approximation Algorithms (Theory)	3	45
	CSEC821P	Approximation Algorithms (Lab)	1	30
	CSEC822T	Big Data Analytics (Theory)	3	45
	CSEC822P	Big Data Analytics (Lab)	1	30
	CSEC823T	Complex Networks (Theory)	3	45
	CSEC823P	Complex Networks (Lab)	1	30
	CSEC824T	Computer Vision (Theory)	3	45
	CSEC824P	Computer Vision (Lab)	1	30
	CSEC825T	Data Mining (Theory)	3	45
	CSEC825P	Data Mining (Lab)	1	30
	CSEC826T	Deep Learning (Theory)	3	45
	CSEC826P	Deep Learning (Lab)	1	30
	CSEC827T	Ethics of AI (Theory)	3	45
	CSEC827P	Ethics of AI (Lab)	1	30
	CSEC828T	Natural Language Processing (Theory)	3	45
	CSEC828P	Natural Language Processing (Lab)	1	30
	CSEC829T	Quantum Algorithms (Theory)	3	45
	CSEC829P	Quantum Algorithms (Lab)	1	30

List of the Subjects related to Indian Knowledge System

Paper code	Name of the Paper	Semester	Credit
CVAC	Tagore Studies	Semester I	3
MJCS22T	Discrete Structures (Theory)	Semester II	2
MJCS31T	Design and Analysis of Algorithms (Theory)	Semester III	1
MJCS52T	Theory of Computation (Theory)	Semester V	1
MJCS63T	Compiler Construction (Theory)	Semester VI	1
CSEC827T	Ethics of AI (Theory)	Semester VIII	1
CSEC828T	Natural Language Processing (Theory)	Semester VIII	1

SEMESTER - I

MJCS11T: Introduction to Programming (Theory)

Overview of Procedural programming using C; Compilation and execution of simple C programs.
Data types, constants and keywords: Declaring, definition and initialization of variables of different data types, typecasting; named constants; keywords.
Operators and expressions: Arithmetic, logical and bitwise operators and expressions; unary and binary operators; precedence and associativity.
Statements: Conditional statements (if-else, switch-case constructs), loops (while, for and do-while loops), Nested statements, scopes of variables in nested statements.
Arrays: Declaration, definition, initialization of one-dimensional and two-dimensional arrays, accessing elements, row-major and column-major representations, string as character array.
Functions: Utility of functions, call by value and call by reference, return types, void functions; prototype declaration and definition of functions, extern declarations; recursive function calls.
Derived data types: Utility of derived data types, Structures and unions, declaration and definitions; access and manipulation of member variables; array of structures.
Pointers: understanding concept of pointers as references, different types of declaration and definitions of pointer variables and their usage; revisit of call-by-reference; variable argument list to functions; command-line arguments in main(); self-referential structure and its application in linked-list.
Memory allocation in C: Static and dynamic allocations. Usage of malloc and calloc functions, free functions.
File I/O: opening and closing a file, read/write/fseek functions.
C preprocessing: preprocessor directives and usages; specifying different types of macros with arguments, defining macros using compile time flags.

References:

1. The C Programming Language - Brian W. Kernighan and Dennis M. Ritchie; Prentice Hall.
2. Programming in ANSI C - E. Balagurusamy; McGraw Hill.
3. Let Us C - Y. Kanetkar; BPB Publications.

MJCS11P: Introduction to Programming (Lab)

Based on Paper **MJCS11T**.

MJCS12T: Digital Electronics (Theory)

Binary, Octal and Hexadecimal number systems.
Binary arithmetic, complements and representation of multi-byte numbers.
Boolean algebra and Logic functions, minimization of boolean functions using algebraic, sum-of-products and product-of-sums, Karnaugh maps and Quine-McClusky methods. Realization using logic gates.
Combinational Functions and Circuits: Realization of logical expression using different logic gates and comparative performance, Functions – code conversion, Decoding, comparison, multiplexing/demultiplexing, addder and subtractor. Design of combinational circuits with combinational ICs.
Sequential Circuits: Structure of sequential circuits, flip-flops – SR, JK, Toggle, Master-slave JK flip-flops; excitation tables, Conversions, clocking aspects, timing and triggering conditions.
Shift registers and synchronous counters: Ripple counters, modulo-n counters, pseudo-random sequence generators using feedback paths.

References:

1. Digital Electronics - M. Morris Mano; PHI.
2. Digital Computer Electronics - Albert P. Malvino, Jerald A. Brown; McGraw Hill.
3. Digital Integrated Electronics - Herbert Taub, Donald Schilling; McGraw Hill.

MJCS12P: Digital Electronics (Lab)

Based on Paper **MJCS12T**.

MNCS01T: Basics of Programming (Theory)

Introduction to C, familiarization with compilation and execution of simple C programs.
Data types, constants and keywords: Declaration, definition and initialization of variables of different data types, named constants; keywords.
Operators and expressions: Arithmetic, logical and bitwise operators and expressions; unary and binary operators.
Statements: Conditional statements (if-else, switch-case constructs), loops (while, for and do-while loops), Nested statements, scopes of variables in nested statements.

Arrays: Declaration, definition, initialization of one-dimensional and two-dimensional arrays, accessing elements, string as character array.
Functions: Utility of functions, call by value and call by reference, return types, void functions; prototype declaration and definition of functions.
Derived data types: Utility of derived data types, Structures and Unions, declaration and definitions; access and manipulation of member variables; array of structures.
Pointers: Understanding concept of pointers as references, different types of declaration and definitions of pointer variables and uses.
Memory allocation in C: Static and dynamic allocations. malloc, calloc and free functions.
File I/O: opening and closing a file, read/write/fseek functions.

References:

1. The C Programming Language - Brian W. Kernighan and Dennis M. Ritchie; Prentice Hall.
2. Programming in ANSI C - E. Balagurusamy; McGraw Hill.
3. Let Us C - Y. Kanetkar; BPB Publications.

MNCS01P: Basics of Programming (Lab)

Based on Paper **MNCS01T**.

MDCST: Fundamentals of Computer Science (Theory)

Introduction to Computers: Computer Characteristics, Concept of Hardware, Software; Evolution of computer and Generations, Types of Computers - General Purpose and Special Purpose Computer, Applications of Computer in Various Fields with modern day examples.
Structure and Working of Computer: Functional Block Diagram of Computer. CPU, ALU, Memory Unit, Bus Structure of Digital Computer – Address, Data and Control Bus; motherboards and components in a typical motherboard.
Computer Memory: Memory Concept, Memory Cell, Memory Organisation, Semiconductor Memory – RAM, ROM, PROM, EPROM, Secondary Storage Devices – Hard Drives, SSDs.
Input/Output Devices: Input Device – Keyboard, Mouse, Scanner, MICR, OMR. Output Devices – VDU, Printers – Dot Matrix, Daisy-wheel, Inkjet, Laser, Line Printers and Plotters.
Operating System: Evolution of Operating System. Functions of Operating System. Types of Operating Systems. Brief demonstration of Windows and Linux Operating Systems; various productivity tools.
Networking: Basic Elements of a Communication System, Data Transmission Media, Topologies,

LAN, MAN, WAN, Internet connectivity options.

References:

1. Computer Fundamentals - P. K. Sinha; BPB Publications.
2. Introduction to Information Technology - V. Rajaraman; PHI.
3. Fundamental of Information Technology - Chetan Shrivastava; Kalyani Publishers.
4. Computers Today - Suresh K Basandra; Galgotia Publications.

SECCS11P: UNIX/LINUX Programming (Lab)

Programming related tools: familiarity with gcc, gdb; C-preprocessing, usage of compile time flags and include paths; making static and dynamic libraries; make utility and makefiles.
System commands: Process and job control commands; file and folder permissions - user, group and other permissions, chmod, chown; text processing and pattern matching – concept of regular expressions; input-output redirections; usage of pipe.
Basic shell programming: shell variables; quoting; string substitution operators; pattern matching and extended pattern matching; Flow control and loops; typed variables; functions.

References:

1. Learning the Bash Shell - Cameron Newham; O'Reilly.
2. Unix programming environment - Kernighan and Pike; PHI.
3. Unix And Shell Programming: A Textbook - Behrouz A. Forouzan, Richard F. Gilberg; Cengage Learning India.
4. Unix: Concepts and Applications - Sumitabha Das; McGraw Hill Education, India.

SEMESTER - II

MJCS21T: Data Structures and Algorithms (Theory)

Concept of Data Types, Abstract Data Type or ADT.
Algorithm: Definition and Properties of an Algorithm, Time Complexity, Order Notation.
Arrays and Linked Lists: Single and Multi-dimensional Arrays; Singly, Doubly and Circular Lists; Skip Lists, Sparse Matrices, Polynomial Representation.
Stacks: Definition, Representation and Uses. Push and Pop Operations. Infix, Prefix and Postfix Notations, Conversion and Evaluation.
Queues: Definition, Representation and Uses. Linear and Circular Queue. enqueue, dequeue, Deque, Priority Queues.
Recursion: Developing Recursive definitions of simple problems and their implementation; Advantages and Limitations of Recursions.
Sorting: Selection Sort, Insertion Sort, Bubble Sort, Shell Sort, Comparison of Sorting Techniques.
Searching: Linear search, Binary search, Tree as a Data Structure, Binary Tree, Tree Traversal, Binary Search Tree - Insertion, Deletion, and Searching in a BST, AVL Tree.

References:

1. Fundamentals of Data Structure in C - E. Horowitz, S. Sahni, S. Anderson-Freed; Computer Science Press.
2. Data Structures & Algorithms - Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman; Pearson Education India.
3. Data Structures Using C - Aaron M. Tenenbaum, Y Langsam, Moshe J. Augenstein; Pearson Education.
4. Data Structures and Program Design in C - Robert L. Kruse, Bruce P. Leung, C.L.Tondo, Shashi Mogalla; Pearson Education.

MJCS21P: Data Structures and Algorithms (Lab)

Based on Paper **MJCS21T**.

MJCS22T: Discrete Structures (Theory)

Introduction: Sets - Finite and Infinite sets, Uncountably Infinite Sets; Relations and Functions, Properties of Binary Relations, Closure, Partial Order Relations; counting - Pigeonhole Principle, Permutation and Combination; Mathematical Induction, Principles of Inclusion and Exclusion. *The concept of zero and its integration into place-value system (number system), Counting and enumeration strategies in Pingala's Chandas Shastra, Arithmetic combinatorics and mental calculation techniques from Vedic Mathematics (e.g., Ekādhikena, Antyayoreva)*
Discrete Numeric Functions: Discrete Numeric Functions (DNF) and their properties, Representation of a DNF using a Generating Function, Applications of Generating Functions, Asymptotic Growth of a DNF and Related Notations.
Recurrences: Representation of a DNF by a Recurrence Relation, Linear Recurrence Relations with Constant Coefficients: Finding Homogeneous Solutions, Particular Solutions, and General Solutions, Solution of Linear Recurrence Relation and Simultaneous Linear Recurrence Relation with Constant Coefficients using Generating Functions.
Boolean Algebra: Lattice and it's Properties, Algebraic Structure induced by a Lattice, Distributive, Complemented, Boolean Lattice, Boolean Algebra, Properties of Boolean Algebra, Boolean Variables and Functions.
Graph Theory: Basic Terminology, Models and Types, Graph Representation, Basic Definitions; Graph Isomorphism, Connectivity, Euler and Hamiltonian Cycle, Circuits, Cliques, Independent Sets, Vertex Covers, Matching, Cut Set, Cut Vertex, Connectivity; Planar Graphs and their properties, Graph Coloring, Trees and their properties.
Prepositional Logic: Logical Connectives, Well-Formed Formulas, Tautologies, Equivalences, Inference Theory. *Nyaya system of inference (Anumana), five-part syllogism (Panchāvayava) and its mapping to formal logical arguments.*

References:

1. Elements of Discrete Mathematics - C. L. Liu, D.P. Mohapatra; McGraw Hill.
2. Discrete Mathematics and Its Applications, Kenneth H. Rosen; McGraw Hill.
3. Graph Theory with Applications to Engineering and Computer Science - N. Deo; Dover Publication.
4. Discrete Mathematical Structures with Applications to Computer Science - J. P. Tremblay and R. P. Manohar; McGraw Hill.

MNCS01T: Basics of Programming (Theory)

Introduction to C, familiarization with compilation and execution of simple C programs.

Data types, constants and keywords: Declaration, definition and initialization of variables of different data types, named constants; keywords.
Operators and expressions: Arithmetic, logical and bitwise operators and expressions; unary and binary operators.
Statements: Conditional statements (if-else, switch-case constructs), loops (while, for and do-while loops), Nested statements, scopes of variables in nested statements.
Arrays: Declaration, definition, initialization of one-dimensional and two-dimensional arrays, accessing elements, string as character array.
Functions: Utility of functions, call by value and call by reference, return types, void functions; prototype declaration and definition of functions.
Derived data types: Utility of derived data types, Structures and Unions, declaration and definitions; access and manipulation of member variables; array of structures.
Pointers: Understanding concept of pointers as references, different types of declaration and definitions of pointer variables and uses.
Memory allocation in C: Static and dynamic allocations. malloc, calloc and free functions.
File I/O: opening and closing a file, read/write/fseek functions.

References:

1. The C Programming Language - Brian W. Kernighan and Dennis M. Ritchie; Prentice Hall.
2. Programming in ANSI C - E. Balagurusamy; McGraw Hill.
3. Let Us C - Y. Kanetkar; BPB Publications.

MNCS01P: Basics of Programming (Lab)

Based on Paper **MNCS01T**.

MDCST: Fundamentals of Computer Science (Theory)

Introduction to Computers: Computer Characteristics, Concept of Hardware, Software; Evolution of computer and Generations, Types of Computers - General Purpose and Special Purpose Computer, Applications of Computer in Various Fields with modern day examples.
Structure and Working of Computer: Functional Block Diagram of Computer. CPU, ALU, Memory Unit, Bus Structure of Digital Computer – Address, Data and Control Bus; motherboards and components in a typical motherboard.
Computer Memory: Memory Concept, Memory Cell, Memory Organisation, Semiconductor Memory – RAM, ROM, PROM, EPROM, Secondary Storage Devices – Hard Drives, SSDs.

Input/Output Devices: Input Device – Keyboard, Mouse, Scanner, MICR, OMR. Output Devices – VDU, Printers – Dot Matrix, Daisy-wheel, Inkjet, Laser, Line Printers and Plotters.
Operating System: Evolution of Operating System. Functions of Operating System. Types of Operating Systems. Brief demonstration of Windows and Linux Operating Systems; various productivity tools.
Networking: Basic Elements of a Communication System, Data Transmission Media, Topologies, LAN, MAN, WAN, Internet connectivity options.

References:

1. Computer Fundamentals - P. K. Sinha; BPB Publications.
2. Introduction to Information Technology - V. Rajaraman; PHI.
3. Fundamental of Information Technology - Chetan Shrivastava; Kalyani Publishers.
4. Computers Today - Suresh K Basandra; Galgotia Publications.

SECCS21P: Programming in Python (Lab)

Setting up Python and IDEs, Running basic Python programs, Working with variables and data types, Implementing Conditional statements (if, elif, else); Looping (for, while) and iteration techniques.
Functions: parameters, return values, and scope; Lambda functions and built-in functions, Recursion and recursive functions.
Working with Data Structures and Comprehensions: Lists, tuples, sets, and dictionaries; List comprehensions and generator expressions; Advanced dictionary techniques; Manipulating strings and regular expressions.
Reading from and writing to files, Handling exceptions and errors, Using context managers for file operations, Working with text and binary files.
Implementing classes and objects, Creating inheritance and polymorphism examples, Practicing encapsulation and abstraction principles.
Object-Oriented Programming (OOP): Introduction to OOP principles and concepts; Defining classes and creating objects; Inheritance, composition, and polymorphism; Advanced OOP topics: abstract classes, interfaces.
Working with external APIs and web services, Serializing and deserializing data with JSON, XML, and Pickle, Visualizing data using Matplotlib and Seaborn.
Introduction to data visualization with libraries like Matplotlib and Seaborn; Introduction to databases, SQLite, and basic SQL operations.

References:

1. Python: The Complete Reference - Martin C. Brown, McGraw Hill Education.
2. Think Python: How to Think Like a Computer Scientist - Allen B. Downey; O'Reilly.
3. Introduction to Computation and Programming Using Python: With Application to Understanding Data - John V Guttag; MIT Press.
4. Fluent Python: Clear, Concise, and Effective Programming - Luciano Ramalho; O'Reilly.

Table 1: Descriptors for qualifications at levels 4.5 on the NHEQF

Element of the Descriptor	NHEQF level descriptors relating to undergraduate certificate
Knowledge and understanding	KU1: Knowledge of facts, concepts, principles, theories, and processes in broad multidisciplinary learning contexts within the chosen fields of learning in broad multidisciplinary learning, KU2: Understanding of the linkages between the learning areas within and across the chosen fields of study, KU3: Procedural knowledge required for performing skilled or paraprofessional tasks associated with the chosen fields of learning.
General, technical and professional skills required to perform and accomplish tasks	GT1: A range of cognitive and technical skills required for accomplishing assigned tasks relating to the chosen fields of learning in the context of broad multidisciplinary contexts. GT2: Cognitive skills required to identify, analyze and synthesize information from a range of sources. GT3: Cognitive and technical skills required for selecting and using relevant methods, tools, and materials to assess the appropriateness of approaches to solving problems associated with the chosen fields of learning.
Application of knowledge and skills	AK1: Apply the acquired operational or technical and theoretical knowledge, and a range of cognitive and practical skills to select and use basic methods, tools, materials, and information to generate solutions to specific problems relating to the chosen fields of learning.
Generic learning outcomes	GL1: Listen carefully, read texts related to the chosen fields of study analytically, and present information in a clear and concise manner to different groups/audiences. GL2: Express thoughts and ideas effectively in writing and orally and present the results/findings of the experiments carried out in a clear and concise manner to different groups. GL3: Meet one's own learning needs relating to the chosen fields of learning. GL4: Pursue self-directed and self-managed learning to upgrade the knowledge and skills required for a higher level of education and training. GL5: Gather and interpret relevant quantitative and qualitative data to identify problems. GL6: Critically evaluate principles and theories associated with the chosen fields of learning. GL7: Make judgment and take decisions, based on analysis of data and evidence, for formulating responses to issues/problems associated with the chosen fields of learning, requiring the exercise of some personal responsibility for action and outputs/outcomes.
Constitutional, humanistic, ethical, and moral values	CH1: Practice constitutional, humanistic, ethical, and moral values in one's life, and practice these values in real-life situations. CH2: Put forward convincing arguments to respond to the ethical and moral issues associated with the chosen fields of learning.
Employability and job-ready skills, and entrepreneurship skills and capabilities/qualities and mindset	EJ1: Knowledge and a basket of essential skills, required to perform effectively in a defined job relating to the chosen fields of study. EJ2: Ability to exercise responsibility for the completion of assigned tasks and for the outputs of own work, and to take some responsibility for group work and output as a member of the group.

Table 2: Level 4.5 (Year 1)

		SEMESTER-I							SEMESTER-II						
Elements of the Descriptor		MJCS11	MJCS12	MNCS01	MDCST	SECCS11P	AECC	CVAC	MJCS21	MJCS22	MNCS01	MDCST	SECCS21P	AECC	CVAC
Knowledge and understanding	KU1	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	KU2	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	KU3	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
General, technical and professional skills required to perform and accomplish tasks	GT1	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	GT2	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	GT3	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
Application of knowledge and skills	AK1	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
Generic learning outcomes	GL1		✓	✓	✓				✓	✓	✓	✓			
	GL2	✓	✓	✓		✓				✓	✓		✓		
	GL3	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	GL4	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	GL5	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	GL6	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	GL7	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
Constitutional, humanistic, ethical, and moral values	CH1						✓	✓	✓	✓	✓	✓	✓	✓	✓
	CH2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Employability and job-ready skills, and entrepreneurship skills and capabilities/qualities and mindset	EJ1	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		
	EJ2	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		

SEMESTER - III

MJCS31T: Design and Analysis of Algorithms (Theory)

Introduction: Properties of an algorithm, Growth of Functions, Asymptotic Notations, Time and Space Complexity.
Sorting and Order Statistics: Review of Sorting Algorithms, Merge sort, Quicksort, Heapsort, Sorting in Linear Time, Medians and Order Statistics.
Tree: Height-Balanced Binary Tree, Red-Black Tree and Various operations on them.
Dynamic Programming: Matrix-Chain Multiplication, Longest Common Subsequence. *Recursive reasoning in Panini's grammar and dynamic construction of derivations, structure of optimal rule selection in Sanskrit parsing.*
Greedy Algorithms: Activity selection problem, Huffman codes and Fractional knapsack problem. Union-Find Algorithm, Greedy Algorithms on Matroid.
Divide-and-Conquer: Strassen's Matrix Multiplication, Merge of Two Sorted Arrays. *Techniques from Chandas Shastra (Pingala) for combinatorial generation of metrical patterns using recursive divide-and-conquer logic.*
Graph Algorithms: Breadth-First Search (BFS) and Depth-First Search (DFS), Minimum Spanning Tree, Shortest Path Algorithms - Dijkstra's, Bellman-Ford, Floyd-Warshall Algorithm, Transitive Closure.

Hashing: Hash Tables, Hash Functions and Collisions, Collision Resolution, Linear Probing, Quadratic Probing, Double Hashing; separate chaining and coalesced hashing.

References:

1. Introduction to Algorithms - Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein; The MIT Press.
2. The Art of Computer Programming, Volumes 1-4B - Donald E. Knuth, Addison-Wesley.
3. Algorithm Design - Jon Kleinberg and Éva Tardos; Addison-Wesley.

MJCS31P: Design and Analysis of Algorithms (Lab)

Based on Paper **MJCS31T**.

MJCS32T: Computer Organization and Architecture (Theory)

Introduction to computer system, Functional Units of Computers and their organization; CPU, Memory, Input-Output System and interconnection buses/system buses.
Computer Memory System Overview, Memory hierarchy, Cache Memory, Cache Memory Principles, Property of locality of reference, performance of cache, hit ratio, miss penalty, Operation Multiple Caches.
Internal Memory: Semiconductor Main memory, RAM, DRAM, SRAM, ROM, PROM, EPROM, EEPROM.
External Memory: Magnetic Memory, Optical Memory, Flash memory.
Input/Output System: External devices, I/O modules; Programmed I/O, Interrupt-driven I/O, Direct Memory Access (DMA), DMA controller, Cycle stealing DMA, burst mode DMA.
Central Processing Unit (CPU): Arithmetic Logic Unit (ALU), Control Unit and CPU memory.
Intel 8085 Microprocessor (as a case study): Set of general and special registers, flags; Various Addressing modes of Intel 8085, Pin configuration, Interrupts, Instruction cycles, control and timing Diagram; Single-byte, two-byte and three-byte instructions; 0-address, 1-address, 2-address and 3-address instructions.

References:

1. Computer Architecture & Organization - M. Morris Mano; PHI.
2. Computer Organization - Carl Hamacher; McGraw Hill.
3. Computer Organization & Architecture - William Stallings; Pearson Education.

MJCS32P: Computer Organization and Architecture (Lab)

Based on Paper **MJCS32T**.

MNCS02T: Data Structures (Theory)

Concept of Data Types, Abstract Data Type or ADT.
Arrays: Single and Multi-dimensional Arrays, Sparse Matrices.
Linked Lists: Singly and Doubly Lists, Insertion and Deletion of Nodes.
Stacks: Definition, Representation and Uses. Push and Pop Operations. Infix, Prefix and Postfix Notations, Conversion and Evaluation.
Queues: Definitions, Representation and Uses. enqueue and dequeue Operations.
Recursion: Recursive definitions of simple problems and their implementation.
Sorting: Bubble Sort, Selection Sort, Insertion Sort.
Searching: Linear Search, Binary Search, Introduction to a Tree as a Data Structure, Binary Tree, Tree Traversal, Binary Search Tree - insertion, deletion.

References:

1. Fundamentals of Data Structure in C - E. Horowitz, S. Sahni, S. Anderson-Freed; Computer Science Press.
2. Data Structures & Algorithms - Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman; Pearson Education India.
3. Data Structures Using C - Aaron M. Tenenbaum, Y Langsam, Moshe J. Augenstein; Pearson Education.
4. Data Structures and Program Design in C - Robert L. Kruse, Bruce P. Leung, C.L.Tondo, Shashi Mogalla; Pearson Education.

MNCS02P: Data Structures (Lab)

Based on Paper **MNCS02T**.

MDCST: Fundamentals of Computer Science (Theory)

Introduction to Computers: Computer Characteristics, Concept of Hardware, Software; Evolution of computer and Generations, Types of Computers - General Purpose and Special Purpose Computer, Applications of Computer in Various Fields with modern day examples.
Structure and Working of Computer: Functional Block Diagram of Computer. CPU, ALU, Memory Unit, Bus Structure of Digital Computer – Address, Data and Control Bus; motherboards and components in a typical motherboard.
Computer Memory: Memory Concept, Memory Cell, Memory Organisation, Semiconductor Memory – RAM, ROM, PROM, EPROM, Secondary Storage Devices – Hard Drives, SSDs.
Input/Output Devices: Input Device – Keyboard, Mouse, Scanner, MICR, OMR. Output Devices – VDU, Printers – Dot Matrix, Daisy-wheel, Inkjet, Laser, Line Printers and Plotters.
Operating System: Evolution of Operating System. Functions of Operating System. Types of Operating Systems. Brief demonstration of Windows and Linux Operating Systems; various productivity tools.
Networking: Basic Elements of a Communication System, Data Transmission Media, Topologies, LAN, MAN, WAN, Internet connectivity options.

References:

1. Computer Fundamentals - P. K. Sinha; BPB Publications.
2. Introduction to Information Technology - V. Rajaraman; PHI.
3. Fundamental of Information Technology - Chetan Shrivastava; Kalyani Publishers.
4. Computers Today - Suresh K Basandra; Galgotia Publications.

SECCS31P: Web Programming (Lab)

Introduction to Web Development: Overview of web technologies and the client-server model. Introduction to HTML, CSS, and JavaScript; Setting up a development environment (text editors, browsers, developer tools).
HTML: Understanding HTML structure: elements, tags, attributes; Creating headings, paragraphs, lists, and links; Semantic HTML and accessibility best practices; HTML5 elements and semantic tags; Accessibility considerations and ARIA roles; Web forms and input validation; Integrating audio, video, and other media elements.
CSS: Introduction to CSS: selectors, properties, values; Styling text, backgrounds, borders, and boxes; Layout techniques: positioning, floating, flexbox basics; Introduction to responsive design and media queries.
JavaScript: Basics of JavaScript: variables, data types, operators; Control structures: conditionals and loops; Functions and scope; Handling events and user interactions; ES6+ features: arrow functions, destructuring, async/await, etc; Modular JavaScript and module bundlers (e.g. Webpack);

Introduction to TypeScript for type-safe JavaScript.
DOM Manipulation: Introduction to the Document Object Model (DOM); Modifying HTML and CSS through JavaScript; Creating dynamic content and interactive web pages.
Front-End Frameworks: Introduction to front-end frameworks (e.g. React, Vue, Angular); Building components and user interfaces; State management and props/props binding (for React); State management with Redux or MobX; Introduction to component libraries (e.g. Material-UI, Ant Design).
Back-End Development and APIs: Introduction to server-side programming with Node.js and Express; Building RESTful APIs and handling HTTP requests; Authentication and authorization using JWT; Introduction to GraphQL as an alternative API architecture.
Databases and Data Management: Relational databases (e.g. MySQL, PostgreSQL) and NoSQL databases (e.g. MongoDB); ORM (Object-Relational Mapping) libraries and database migrations; Data validation and sanitization.
Web Security and Performance: Common web security threats and secure coding practices; HTTPS, CORS, and Cross-Site Scripting (XSS) prevention; Performance optimization techniques (e.g. lazy loading, minification); Introduction to Progressive Web Apps (PWAs) and Service Workers.
Deployment and DevOps: Deploying web applications using cloud platforms (e.g. AWS, Heroku, Netlify); Continuous integration and continuous deployment (CI/CD) pipelines; Containerization with Docker and container orchestration (e.g. Kubernetes).

References:

1. HTML and CSS: Design and Build Websites - Jon Duckett, Wiley.
2. CSS Secrets: Better Solutions to Everyday Web Design Problems - Lea Verou; O'Reilly Media.
3. Eloquent Javascript 3E: A Modern Introduction to Programming - Marijn Haverbeke; No Starch Press.
4. React – Up & Running: Building Web Applications - Stoyan Stefanov; O'Reilly Media.
5. Node.js Design Patterns: Design and implement production-grade Node.js applications using proven patterns and techniques - Mario Casciaro, Luciano Mammino; Packt Publishing Limited.
6. MongoDB: The Definitive Guide - Shannon Bradshaw, Eoin Brazil, and Kristina Chodorow; O'Reilly.
7. Web Security for Developers: Real Threats, Practical Defense - Malcolm McDonald; No Starch Press.
8. High Performance Web Sites: Essential Knowledge for Front-end Engineers - Steve Souders; O'Reilly Media.

9. Docker Deep Dive - Nigel Poulton; O'Reilly Media.

10. The Phoenix Project: A Novel about It, Devops, and Helping Your Business - Gene Kim; IT Revolution Press.

SEMESTER - IV

MJCS41T: Object-Oriented Programming (Theory)

Introduction to C and C++: History of C and C++, Overview of Procedural Programming and Object-Oriented Programming, Using main() function, Compiling and Executing Simple Programs in C++.
Data Types and Expressions, Loops.
Functions and Arrays: Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Function's parameters.
Pointers and References in C++: Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers vs. References, Declaring and initializing references, Using references as function arguments and function return values.
Memory Allocation in C++: Differentiating between static and dynamic memory allocation, use of new and delete operators, storage of variables in static and dynamic memory allocation.
File I/O, Preprocessor Directives: Opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes), Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files.
Using Classes in C++: Principles of Object-Oriented Programming, Defining & Using Classes, Class Constructors, Constructor Overloading, Function overloading in classes, Class Variables and Functions, Objects as parameters, Specifying the Protected and Private Access, Copy Constructors, Overview of Template classes and their use.
Overview of Function Overloading and Operator Overloading: Need of Overloading functions and operators, Overloading functions by number and type of arguments, Looking at an operator as a function call, Overloading Operators (including assignment operators, unary operators).
Inheritance, Polymorphism and Exception Handling: Introduction to Inheritance (Multi-Level Inheritance, Multiple Inheritance), Polymorphism (Virtual Functions, Pure Virtual Functions), Basics Exceptional Handling (using catch and throw, multiple catch statements), Catching all exceptions, Restricting exceptions.

References:

1. C++: The Complete Reference - Herbtz Schildt; McGraw Hill.
2. Programming: Principles and Practice Using C++ - Bjarne Stroustrup; Addison-Wesley.
3. Programming with C++ (Schaum's Outlines) - John R. Hubbard; McGraw-Hill Education.

MJCS41P: Object-Oriented Programming (Lab)

Based on Paper **MJCS41T**.

MJCS42T: Software Engineering (Theory)

Introduction: The Evolving Role of Software, Software Characteristics, Changing Nature of Software, Software Engineering as a Layered Technology, Software Process Framework, Framework and Umbrella Activities, Process Models, Capability Maturity Model Integration (CMMI).
Requirement Analysis: Software Requirement Analysis, Initiating Requirement Engineering Process, Requirement Analysis and Modeling Techniques, Flow Oriented Modeling, Need for SRS, Characteristics and Components of SRS.
Software Project Management: Estimation in Project Planning Process, Project Scheduling.
Risk Management: Software Risks, Risk Identification, Risk Projection and Risk Refinement, RMMM Plan.
Quality Management: Quality Concepts, Software Quality Assurance, Software Reviews,Metrics for Process and Projects.
Design Engineering: Design Concepts, Architectural Design Elements, Software Architecture,Data Design at the Architectural Level and Component Level, Mapping of Data Flow into Software Architecture, Modeling Component Level Design.
Testing Strategies & Tactics: Software Testing Fundamentals, Strategic Approach to Software Testing, Test Strategies for Conventional Software, Validation Testing, System testing, Black-Box Testing, White-Box Testing and their type, Basis Path Testing.

References:

1. Software Engineering: A Practitioner's Approach - Roger Pressman; McGraw Hill Indian Edition.
2. Software Engineering - Ian Sommerville; Pearson Education.
3. Fundamentals of Software Engineering - Rajib Mall; PHI.

MJCS42P: Software Engineering (Lab)

Based on Paper **MJCS42T**.

MJCS43T: Operating Systems (Theory)

Introduction: Functions of operating systems, Kernels, System calls, System programs, Types of operating systems.
Process Management: Process abstraction, System calls for process management, Process execution mechanisms, Scheduling policies, Threads, Threading issues, Thread libraries, Concurrent processes, Locks, Semaphores, Classical problems in synchronization, Inter-process communication.
Deadlocks: Characterization, Prevention, Avoidance, Detection, Recovery.
Memory Management: Physical and virtual address space, Memory allocation strategies (fixed and variable partitions), Paging, Segmentation, Virtual memory, Page Replacement.
File and I/O Management: Concept of a file, Directory structure, File systems, File allocation methods, Disks scheduling, Device controllers and device drivers.

References:

1. Operating Systems Concepts - A Silberschatz, P.B. Galvin, G. Gagne; Wiley Edition.
2. Modern Operating Systems - Andrew S. Tanenbaum, Herbert Bos; Pearson.
3. Operating Systems, Internals and Design Principles - W. Stallings; Prentice Hall.

MJCS43P: Operating Systems (Lab)

Based on Paper **MJCS43T**.

MJCS44T: Computational Statistics (Theory)

Statistical properties of a given data set: plotting as scatter plot, histogram and line plot; mean and variance; frequencies in bins; plot of frequency vs bins; central tendency; concepts of skewness and curtosis.
Linear relationship between two sets of data points: best-fit straight line and deviations; prediction of newer points based on best-fit line; linear correlation and Pearson's linear correlation coefficient, monotonicity and Spearman's rank coefficient; concept of covariance.
Recap of probability: definition, joint and conditional probability, Bayes' criteria.
Random variables: Concept, definition, discrete and continuous radom variables, PMF, PDF and CDF functions, examples of common types of distributions, plots, joint PDF and CDF of two random variables. Expected values of random variables and their functions, concept of moments and characteristic functions.
Basics of queuing theory as application of random variables: Little's theorem, M/M/1 and M/M/n

queuing system, arrival, departure and waiting time and queue-size statistics.

References:

1. A First Course on Probability - Sheldon Ross; Prentice Hall India.
2. Probability and Statistics (Schaum's Outlines) - Murray R. Spiegel, John J. Schiller, R. Alu Srinivasan; McGraw Hill.
3. Fundamentals of Statistics - A. M. Goon, M. K. Gupta, B. Dasgupta; World Press Pvt Ltd.

MJCS44P: Computational Statistics (Lab)

Based on Paper **MJCS44T**.

MNCS02T: Data Structures (Theory)

Concept of Data Types, Abstract Data Type or ADT.
Arrays: Single and Multi-dimensional Arrays, Sparse Matrices.
Linked Lists: Singly and Doubly Lists, Insertion and Deletion of Nodes.
Stacks: Definition, Representation and Uses. Push and Pop Operations. Infix, Prefix and Postfix Notations, Conversion and Evaluation.
Queues: Definitions, Representation and Uses. enqueue and dequeue Operations.
Recursion: Recursive definitions of simple problems and their implementation.
Sorting: Bubble Sort, Selection Sort, Insertion Sort.
Searching: Linear Search, Binary Search, Introduction to a Tree as a Data Structure, Binary Tree, Tree Traversal, Binary Search Tree - insertion, deletion.

References:

1. Fundamentals of Data Structure in C - E. Horowitz, S. Sahni, S. Anderson-Freed; Computer Science Press.
2. Data Structures & Algorithms - Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman; Pearson Education India.
3. Data Structures Using C - Aaron M. Tenenbaum, Y Langsam, Moshe J. Augenstein; Pearson Education.
4. Data Structures and Program Design in C - Robert L. Kruse, Bruce P. Leung, C.L.Tondo, Shashi Mogalla; Pearson Education.

MNCS02P: Data Structures (Lab)

Based on Paper **MNCS02T**.

Table 3: Descriptors for qualifications at levels 5 on the NHEQF

Element of the Descriptor	NHEQF level descriptors
Knowledge and understanding	KU1: Theoretical and technical knowledge in broad multidisciplinary contexts within the chosen fields of learning. KU2: Deeper knowledge and understanding of one of the learning areas and its underlying principles and theories. KU3: Procedural knowledge required for performing skilled or paraprofessional tasks associated with the chosen fields of learning.
Skills required to perform and accomplish tasks	SR1: Cognitive and technical skills required for performing and accomplishing complex tasks relating to the chosen fields of learning, SR2: Cognitive and technical skills required to analyze and synthesize ideas and information from a range of sources and act on information to generate solutions to specific problems associated with the chosen fields of learning.
Application of knowledge and skills	AK1: Apply the acquired specialized or theoretical knowledge, and a range of cognitive and practical skills to gather quantitative and qualitative data. AK2: Select and apply basic methods, tools, materials, and information to formulate solutions to problems related to the chosen field(s) of learning.
Generic learning outcomes	GL1: Listen carefully, read texts related to the chosen fields of learning analytically, and present complex information in a clear and concise manner to different groups/audiences. GL2: Communicate in writing and orally the information, arguments, and results of the experiments and studies conducted accurately and effectively to specialist and non-specialist audiences. GL3: Meet one's own learning needs relating to the chosen field(s) of learning, work/vocation, and an area of professional practice. GL4: Pursue self-paced and self-directed learning to upgrade knowledge and skills required for pursuing a higher level of education and training. GL5: Critically evaluate the essential theories, policies, and practices by following a scientific approach to knowledge development. GL6: Make judgement and take decision, based on the analysis and evaluation of information, for determining solutions to a variety of unpredictable problems associated with the chosen fields of learning, taking responsibility for the nature and quality of outputs.
Constitutional, humanistic, ethical, and moral values	CH1: Embrace the constitutional, humanistic, ethical, and moral values, practice these values in life, and take a position regarding these values. CH2: Formulate arguments in support of actions to address issues relating the ethical and moral issues relating to the chosen fields of learning, including environmental and sustainable development issues, from multiple perspectives.
Employability and job-ready skills, and entrepreneurship skills and capabilities or qualities and mindset	EJ1: Take up job/employment relating to the chosen fields of study or professional practice requiring the exercise of full personal responsibility for the completion of tasks and for the outputs of own work, and full responsibility for the group task/ work as a member of the group/team. EJ2: Exercise self-management within the guidelines of study and work contexts EJ3: Supervise the routine work of others, taking some responsibility for the evaluation and improvement of work or study activities.

Table 4: Level 5.0 (Year 2)

		SEMESTER-III						SEMESTER-IV					
Elements of the Descriptor		MJCS31	MJCS32	MNCS02	MDCST	SECCS31P	AECC	MJCS41	MJCS42	MJCS43	MJCS44	MNCS02	AECC
Knowledge and understanding	KU1	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	KU2	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	KU3	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
Skills required to perform and accomplish tasks	SR1	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	SR2	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
Application of knowledge and skills	AK1	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	AK2	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
Generic learning outcomes	GL1	✓	✓	✓	✓			✓	✓	✓	✓	✓	
	GL2	✓	✓	✓		✓		✓	✓	✓	✓	✓	
	GL3	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	GL4	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	GL5	✓	✓	✓				✓	✓	✓	✓	✓	
	GL6	✓	✓	✓		✓		✓	✓	✓	✓	✓	
Constitutional, humanistic, ethical, and moral values	CH1						✓						✓
	CH2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Employability and job-ready skills, and entrepreneurship skills and capabilities/qualities and mindset	EJ1	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	EJ2	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	EJ3	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	

SEMESTER - V

MJCS51T: Database Management Systems (Theory)

Introduction: Characteristics of database approach, Data models, Database system architecture and data independence.
Relational data model: Relational model concepts, Keys and other relational constraints, SQL queries.
Entity-Relationship (E-R) Model: Entity sets, Relationship sets, Attributes, Constraints, E-R diagrams, Mapping E-R model to relational model.
Formal Relational Query Languages: Relational algebra, Tuple relational calculus, Domain relational calculus.
Relational Database Design: Normalization concepts (1NF, 2NF, 3NF, BCNF), Functional dependency, Closure of a set of functional dependencies, Closure of attribute sets, Canonical cover, Lossless decomposition, Dependency preservation.
File Structure and Indexing: Overview of file organizations, Files of unordered and ordered records, Indexing concepts (primary, secondary and clustering indices), Multilevel indexing, B+ tree indexing.

Transaction Processing: ACID properties, Overview of concurrency control.

References:

1. Database System Concepts - A. Silberschatz, H.F. Korth, S. Sudarshan; McGraw Hill.
2. Fundamentals of Database Systems - R. Elmasri, S.B. Navathe; Pearson Education.
3. Database Management Systems - R. Ramakrishanan, J. Gehrke; McGraw Hill.
4. MySQL - Reference Manual.

MJCS51P: Database Management Systems (Lab)

Based on Paper **MJCS51T**.

MJCS52T: Theory of Computation (Theory)

Finite automata and regular languages: *Panini's Ashtadhyayi and rule-based generative grammar, Finite-state modeling of Sanskrit morphology.* Regular grammars and regular languages, Finite automata, Transition graphs, Equivalence of deterministic and non-deterministic finite automata, Minimization of finite automata, Regular expressions, Equivalence of regular languages, finite automata and regular expression, Pumping lemma for regular languages, Properties of regular languages.
Context free languages: Context free grammars and context free languages, Parse trees, Ambiguities in grammars and languages, Pushdown automata, Equivalance of Pushdown automata and Context free languages, Deterministic Pushdown automata, Pumping lemma for context free languages, Properties of context free languages, Normal forms – Chomsky normal form and Greibach normal form. *Chandas Shastra and metrical pattern generation, Binary representations of syllabic meters and their relation to context-free structures.*
Turing machines and hierarchy of formal languages: Turing machine as a model of computation, Universal Turing machine, Turing's thesis, Recursive and recursively enumerable languages, Context sensitive grammars and bounded linear automata, Chomsky hierarchy, decidability, Turing machine halting problem.

References:

1. An Introduction to Formal Language and Automata - P. Linz; Jones and Bartlett Publishers Inc.

2. Introduction to Automata Theory, Language & Computation – Hopcroft, Motwani, Ullman; Pearson Education.

3. Elements of the Theory of Computation - Lewis & Papadimitriou; PHI.

MJCS53T: Digital Communication (Theory)

Introduction: Earliest digital communication system, types of transmission, Mapping between bits to signals - digital and analog waveforms.
Conversion from Analog to Digital signals: Sampling, Nyquist criteria and rate, quantization, quantization noise, PCM and PWM, Delta modulation.
Digital waveforms: Binary waveforms line coding techniques - NRZ-L and Manchester coding, differential encoding - NRZ-I, differential Manchester coding, Ternary waveforms - AMI, B8ZS, HDB3 waveforms.
Frequency characteristics of pulses: Frequency characteristics of signals, Fourier series and fourier transforms for periodic and non-periodic signals, concept of Inter-Symbol-Interference, raised cosine waveforms.
Analog waveforms: Wireless channels and concept of bandwidth, binary ASK, FSK and PSK modulation and demodulation schemes - M-ary PAM, PSK and FSK modulation and demodulation schemes, signal constellations, differential coding and modulation/demodulation.
Noise: AWGN characteristics, Bit-error rate, Signal to noise ratio, Shannon's channel capacity theorem.
Framing and error detection: Frame-level and bit-level framing, bit-stuffing; length based framing; parity check bits and CRC - overheads and performance bounds, error-correcting codes.

References:

1. Data Communications and Networking - Behrouz A Forouzan; McGraw Hill Education.

2. Principles of Communication Systems - Herbut Taub, Donald L. Schilling, Goutam Saha; McGraw Hill Education.

3. Digital Communications - John G Proakis; McGraw Hill.

4. Digital Communications - Simon Haykin; Wiley Student Edition.

MJCS53P: Digital Communication (Lab)

Based on Paper **MJCS53T**.

MNCS03T: Programming in JAVA (Theory)

Introduction to Java: Java Architecture and Features; Semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords, Data Types, Operators, Expressions, Constructs, Java Methods.
Arrays, Strings and I/O: One Dimensional and Multi-dimensional arrays, referencing Arrays Dynamically, usage of String class, String Buffer Classes.
Simple I/O using System.out and the Scanner class, Reading/Writing from console and files.
Object-Oriented Programming Overview: Principles of Object-Oriented Programming, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables and Methods, Objects as parameters, final Classes, Object class, Garbage Collection.
Inheritance: Single Level and Multilevel inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.
Exception Handling: Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions.

References:

1. Java™ Programming Language, The (Java Series) - Ken Arnold, James Gosling, David Holmes; Addison-Wesley.
2. Core Java 2 Volume 1 - Cay S. Horstmann, Gary Cornell; Prentice Hall.
3. Core Java 2 Volume 2 - Advanced Features - Cay S. Horstmann, Gary Cornell; Prentice Hall.
4. Java: How to Program - Paul Deitel, Harvey Deitel; Prentice Hall.
5. Head First Java - Kathy Sierra, Bert Bates, Trisha Gee; O'Rielly Media, Inc.

MNCS03P: Programming in JAVA (Lab)

Based on Paper **MNCS03T**.

SEMESTER - VI

MJCS61T: Computer Graphics (Theory)

Introduction: Concept of digitization of pictures using pixels, pixel co-ordinate systems, grayscale and color models, display systems, frame buffers.
Drawing basic 2D shapes: Points and lines, line-drawing algorithms, circle generating algorithms, ellipse and other conics, polynomial and spline curves, polygons, filling algorithms - scan-line fill, color fill, pattern fill, soft-fill.
2D geometric transformations: Translation, rotation, scaling, composite transformations, matrix representation and computational efficiency, reflection, shear,transformation between different co-ordinate systems.
Viewing and clipping: Viewing coordinate reference frame, window to viewport coordinate transformation, 2D viewing functions, Clipping operations - line clipping, polygon clipping, curve clipping, text clipping, exterior clipping.
Graphical user interfaces: User dialogs, windows and widgets, interactive picture construction techniques - positioning methods, constraints, grids, rubber-and methods, dragging, painting and drawing.

References:

1. Computer Graphics with OpenGL - Donald Hearn and M. Pauline Baker; Pearson.
2. Procedural Elements for Computer Graphics - David F. Rogers; Tata Mc-Graw Hill Education.
3. Mathematical Elements for Computer Graphics - David F. Rogers, J. Alan Adams; Tata Mc-Graw Hill Education.

MJCS61P: Computer Graphics (Lab)

Based on Paper **MJCS61T**.

MJCS62T: Computer Networks (Theory)

Introduction: Topologies of data networks - bus, star, ring, mesh and hybrid; concept of multiplexing over shared physical links.
Multiplexing Techniques : Circuit-Switching and Packet switching concepts - FDM, TDM, OFDM,

statistical TDMA.
Multiple Access Techniques: Contention based techniques - unslotted/slotted Aloha, CSMA-CD, CSMA-CA, Token-ring and DQDB.
Data Link Control Protocols - HDLC operations, ARQ types - Stop-and-wait/ Go-Back-N/Selective-Repeat/Hybrid type I/II/III.
Interconnecting physical networks: Switches, bridges, routers.
IP Functions: IPv4 addressing - Class A/B/C addressing, subnetting, private IP addresses, NAT and Proxies, dynamic IP address allocation and DHCP, CIDR addresses, supernetting, IPv4 header parameters, IPv4 Options, routing and forwading, fragmentation, ICMP, IPv6 addressing.
Transport Layer: Concept of ports, UDP functions, TCP - Connection management, reliability, flow and congestion control mechanisms, UDP and TCP client-server socket programs.
DNS - Nameservers, iterative and recursive queries, types of queries and resolutions, overview of Dynamic DNS.

References:

1. Data Communications and Networking - Behrouz A Forouzan, McGraw Hill Education.
2. TCP/IP Illstrated(Volume 1) - Richard Stevens; Addison-Wesley.
3. Computer Networks - Andrew S Tanenbaum; Pearson Education.

MJCS62P: Computer Networks (Lab)

Based on Paper **MJCS62T**.

MJCS63T: Compiler Construction (Theory)

Introduction: Compilers and Translators, Structure of a Compiler, Compiler writing tools, Lexical and syntactic structure of a language.
Lexical analysis: Finite automata, Regular expression, Lexical analyzer generator. *Lexical processing techniques in Sanskrit — identification of roots (dhātus) and affixes (pratyaya); rule-based sandhi processing as early lexical analysis.*
Syntax Analysis: Notion of top-down and bottom-up parsing, LL parsing, Operator-precedence parsing, LR parsing (SLR, LALR, and CLR parsing), Syntax Directed Translation, Parser generator.
Semantic Analysis: Declaration processing, Type checking, Symbol tables. *Mapping to karaka roles and syntactic-semantic interfaces in Paninian grammar; function tagging based on semantic dependencies.*

Intermediate Code Generation: Run-time environments, translation of language constructs.
Code Generation: Flow-graphs, Register allocation, Code-generation algorithms.
Error handling and recovery. *Meta-rules (paribhāṣās) and exception handling in ancient Indian grammars; systematic conflict resolution using meta-linguistic strategies*
Code optimization techniques. *Rule minimization and transformation efficiency in Panini's sutra design as a model of linguistic and logical optimization.*

References:

1. Principles of Compiler Design - Alfred V. Aho, R Sethi Jeffrey D. Ullman, Monica S Lam; Pearson Education.
2. Compiler Design in C - Allen I. Holub; Prentice Hall of India.
3. Compiler Construction: Principles & Practice - Kenneth C. Louden; Thomson Learning.
4. The Theory and Practice of Compiler Writing - Jean-Paul Tremblay and Paul G. Sorenson, McGraw Hill.

MJCS63P: Compiler Construction (Lab)

Based on Paper **MJCS63T**.

MNCS03T: Programming in JAVA (Theory)

Introduction to Java: Java Architecture and Features; Semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords, Data Types, Operators, Expressions, Constructs, Java Methods.
Arrays, Strings and I/O: One Dimensional and Multi-dimensional arrays, referencing Arrays Dynamically, usage of String class, String Buffer Classes.
Simple I/O using System.out and the Scanner class, Reading/Writing from console and files.
Object-Oriented Programming Overview: Principles of Object-Oriented Programming, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables and Methods, Objects as parameters, final Classes, Object class, Garbage Collection.
Inheritance: Single Level and Multilevel inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.
Exception Handling: Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions.

References:

1. Java™ Programming Language, The (Java Series) - Ken Arnold, James Gosling, David Holmes; Addison-Wesley.
2. Core Java 2 Volume 1 - Cay S. Horstmann, Gary Cornell; Prentice Hall.
3. Core Java 2 Volume 2 - Advanced Features - Cay S. Horstmann, Gary Cornell; Prentice Hall.
4. Java: How to Program - Paul Deitel, Harvey Deitel; Prentice Hall.
5. Head First Java - Kathy Sierra, Bert Bates, Trisha Gee; O'Rielly Media, Inc.

MNCS03P: Programming in JAVA (Lab)

Based on Paper **MNCS03T**.

Table 5: Descriptors for qualifications at levels 5.5 on the NHEQF

Element of the descriptor	NHEQF level descriptors
Knowledge and understanding	KU1: Comprehensive, factual, theoretical, and specialized knowledge in broad multidisciplinary contexts with depth in the underlying principles and theories relating to one or more fields of learning. KU2: Knowledge of the current and emerging issues and developments within the chosen field(s) of learning. KU3: Procedural knowledge required for performing and accomplishing professional tasks associated with the chosen fields of learning.
General, technical and professional skills required to perform and accomplish tasks	GT1: Cognitive and technical skills required for performing and accomplishing complex tasks relating to the chosen fields of learning. GT2: Cognitive and technical skills required to evaluate and analyze complex ideas. GT3: Cognitive and technical skills required to generate solutions to specific problems associated with the chosen fields of learning.
Application of knowledge and skills	AK1: Apply the acquired specialized technical or theoretical knowledge, and cognitive and practical skills to gather and analyze quantitative/ qualitative data to assess the appropriateness of different approaches to solving problems. AK2: Employ the right approach to generate solutions to problems related to the chosen fields of learning.
Generic learning outcomes	GL1: Listen carefully, to read text related to the chosen fields of learning analytically and present complex information in a clear and concise manner to different groups/audiences. GL2: Communicate in writing and orally the constructs and methodologies adopted for the studies undertaken relating to the chosen fields of learning. GL3: Make coherent arguments to support the findings/results of the study undertaken to specialist and non-specialist audiences. GL4: Meet one's own learning needs relating to the chosen field(s) of learning. GL5: Pursue self-paced and self-directed learning to upgrade knowledge and skills that will help adapt to changing demands of the workplace and pursue higher level of education and training. GL6: Critically evaluate evidence for taking actions to generate solutions to specific problems associated with the chosen fields of learning based on empirical evidence. GL7: Make judgement and take decisions based on the analysis and evaluation of information for formulating responses to problems, including real-life problems. GL8: Exercise judgement across a broad range of functions based on empirical evidence, for determining personal and/or group actions to generate solutions to specific problems associated with the chosen fields of learning.
Constitutional, humanistic, ethical, and moral values	CH1: Embrace constitutional, humanistic, ethical, and moral values, and practice these values in life. CH2: Identify ethical issues related to the chosen fields of study. CH3: Formulate coherent arguments about ethical and moral issues, including environmental and sustainable development issues, from multiple perspectives. CH4: Follow ethical practices in all aspects of research and development, including avoiding unethical practices such as fabrication, falsification or misrepresentation of data or committing plagiarism.
Employability and job-ready skills, and entrepreneurship skills and capabilities or qualities and mindset	EJ1: Knowledge and essential skills set and competence that are necessary to take up a professional job relating to the chosen field of learning and professional practice. EJ2: Entrepreneurship skills and mindset required for setting up and running an economic enterprise or pursuing self-employment requiring the exercise of full personal responsibility for the outputs of own work, and full responsibility for the output of the group. EJ3: The ability to exercise management and supervision in the contexts of work or study activities involving unpredictable work processes and working environments.

Table 6: Level 5.5 (Year 3)

		SEMESTER-V				SEMESTER-VI			
Elements of the Descriptor		MJCS51	MJCS52T	MJCS53	MNCS03	MJCS61	MJCS62	MJCS63	MNCS03
Knowledge and understanding	KU1	✓	✓	✓	✓	✓	✓	✓	✓
	KU2	✓	✓	✓	✓	✓	✓	✓	✓
	KU3	✓	✓	✓	✓	✓	✓	✓	✓
General, technical and professional skills required to perform and accomplish tasks	GT1	✓	✓	✓	✓	✓	✓	✓	✓
	GT2	✓	✓	✓	✓	✓	✓	✓	✓
	GT3	✓	✓	✓	✓	✓	✓	✓	✓
Application of knowledge and skills	AK1	✓	✓	✓	✓	✓	✓	✓	✓
	AK2	✓	✓	✓	✓	✓	✓	✓	✓
Generic learning outcomes	GL1	✓	✓	✓	✓	✓	✓	✓	✓
	GL2	✓	✓	✓	✓	✓	✓	✓	✓
	GL3	✓	✓	✓	✓	✓	✓	✓	✓
	GL4	✓	✓	✓	✓	✓	✓	✓	✓
	GL5	✓	✓	✓	✓	✓	✓	✓	✓
	GL6	✓	✓	✓	✓	✓	✓	✓	✓
	GL7	✓	✓	✓	✓	✓	✓	✓	✓
	GL8	✓	✓	✓	✓	✓	✓	✓	✓
Constitutional, humanistic, ethical, and moral values	CH1								
	CH2	✓	✓	✓	✓	✓	✓	✓	✓
	CH3	✓	✓	✓	✓	✓	✓	✓	✓
	CH4	✓	✓	✓	✓	✓	✓	✓	✓
Employability and job-ready skills, and entrepreneurship skills and capabilities/qualities and mindset	EJ1	✓	✓	✓	✓	✓	✓	✓	✓
	EJ2	✓	✓	✓	✓	✓	✓	✓	✓
	EJ3	✓	✓	✓	✓	✓	✓	✓	✓

SEMESTER - VII

MJCS71T: Advanced Algorithms (Theory)

Amortized analysis of algorithms: Different techniques.
Advanced data structures: Red black trees, Binomial heaps, Fibonacci heaps and their applications.
Graph algorithms: Strongly connected components of a directed graph; Max flow algorithms, Bipartite and general matching algorithms.

Matrix operations: Strassen's algorithm, Matrix inversion algorithm.
Polynomial and FFT: FFT algorithms.
Number theoretic algorithms: Primality testing, Integer factorization.
String matching: Rabin Karp algorithm, KMP algorithms.
NP-Completeness: Notion of polynomial time reducibility, Class P, NP, NP-Hard, NP-Complete, Proving NP-Completeness of basic problems.

References:

1. Design and Analysis of Computer Algorithms - Aho, Hopcroft, and Ullman; Addison-Wesly.
2. Introduction to Algorithms - T H Cormen, C E Leiserson, R L Rivest, and C Stein; PHI.
3. Algorithm Design - J Kleinberg and Eva Tardos; Pearson Education.

MJCS71P: Advanced Algorithms (Lab)

Based on Paper **MJCS71T**.

MJCS72T: Artificial Intelligence (Theory)

Introduction: Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent approaches to AI, Introduction to Intelligent Agents, their structures, behavior and environment.
Problem Solving and Searching Techniques: Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its variations, Heuristics Search Techniques - Best First Search, A* algorithm, Constraint Satisfaction Problem, Means-End Analysis, Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms.
Knowledge Representation: Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs.
Programming in Logic (PROLOG).
Dealing with Uncertainty and Inconsistencies: Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations.
Understanding Natural Languages: Parsing Techniques, Context-Free and Transformational Grammars, Recursive and Augmented Transition Nets.

References:

1. Introduction to A.I and Expert Systems - DAN.W. Patterson; Prentice Hall of India.

2. Artificial Intelligence: A Modern Approach - Stuart J. Russell and Peter Norvig; Prentice Hall.

3. Artificial Intelligence – Rich & Knight; Tata McGraw Hill.

MJCS72P: Artificial Intelligence (Lab)

Based on Paper **MJCS72T**.

MJCS73T: Soft Computing (Theory)

Notion of uncertainty, imprecision, ambiguity, vagueness, likelihood. Concept of Fuzziness.
Fuzzy sets, Fuzzy relations, Fuzzy logic.
Applications of Fuzzy mathematics.
Nature inspired computing: Introduction to genetic algorithm, its uses in search and optimization.
Role of genetic operators in the design of genetic algorithm (selection, crossover, mutation).
Significance of parameter setting in deciding the performance of a genetic scheme.
Applicability of GA in single objective optimization.
Multiobjective optimization.
Concept of Neural Networks: Inspiration and lessons from the brain, introduction to biological neural networks, Models of artificial neurons, threshold logic, binary neurons and pattern dichotomizers, perceptron, it learning rule and convergence.
Multilayered perceptron, learning algorithms, function approximation, generalization, regularization networks, Radial Basis Function (RBF) network and learning. VC-dimension, support vector machines (regression and classification), activation function, learning, contrastive divergence.

References:

1. Fuzzy Logic with Engineering Applications - Ross and Ross; Wiley Student Edition.

2. Neural Networks and learning machines - Simon Haykin; Pearson.

3. Genetic Algorithms in Search Optimization and Machine Learning - D. E. Goldberg; Addison-Wesley.

MJCS73P: Soft Computing (Lab)

Based on Paper **MJCS73T**.

MNCS04T: Basics of Python Programming (Theory)

Setting up Python and IDEs, Running basic Python programs.
Implementing Conditional statements (if, elif, else); Looping (for, while) and iteration techniques; Functions: parameters, return values, and scope; Lambda functions and built-in functions, Recursion and recursive functions.
Working with Data Structures and Comprehensions: Lists, tuples, sets, and dictionaries; List comprehensions and generator expressions; Advanced dictionary techniques; Manipulating strings and regular expressions.
Reading from and writing to files, Handling exceptions and errors, Using context managers for file operations, Working with text and binary files.
Implementing classes and objects, Creating inheritance and polymorphism examples, Practicing encapsulation and abstraction principles.
Object-Oriented Programming (OOP): Introduction to OOP principles and concepts; Defining classes and creating objects; Inheritance, composition, and polymorphism; Advanced OOP topics: abstract classes, interfaces.
Working with external APIs and web services, Serializing and deserializing data with JSON, XML, and Pickle, Visualizing data using Matplotlib and Seaborn.
Introduction to data visualization with libraries like Matplotlib and Seaborn; Introduction to databases, SQLite, and basic SQL operations.

References:

1. Python: The Complete Reference - Martin C. Brown; McGraw Hill Education.
2. Think Python: How to Think Like a Computer Scientist - Allen B. Downey; O'Reilly.
3. Introduction to Computation and Programming Using Python: With Application to Understanding Data - John V Guttag; MIT Press.
4. Fluent Python: Clear, Concise, and Effective Programming - Luciano Ramalho; O'Reilly.

MNCS04P: Basics of Python Programming (Lab)

Based on Paper **MNCS04T**.

CSR71: Research Project I

CSEC7XYT: Elective I (Theory)

CSEC7XYP: Elective I (Lab)

Based on Paper **CSEC7XYT**.

SEMESTER - VIII

MJCS81T: Image Processing (Theory)

Introduction: Image processing systems and its applications.
Image formation: Geometric and photometric models; Digitization - sampling, quantization; Image definition and its representation, neighborhood metrics; Point spread function and its properties.
Intensity transformations and spatial filtering: Enhancement, contrast stretching, histogram specification, local contrast enhancement; Smoothing, linear and order statistic filtering, sharpening, spatial convolution, Gaussian smoothing, DoG, LoG; Fuzzy techniques for intensity transformations and spatial filtering.
Segmentation : Pixel classification; Grey level thresholding, global/local thresholding.
Registration : Monomodal/multimodal image registration; Global/local registration; Transform and similarity measures for registration; Intensity/pixel interpolation.
Color image processing: Fundamentals of different color models - RGB, CMY, HSI, YCbCr, Lab; False color; Pseudocolor; Enhancement.

References:

1. Digital Image Processing - R. C. Gonzalez and R. E. Woods; Prentice Hall.
2. Image Processing: The Fundamentals - Maria Petrou and Panagiota Bosdogianni; John Wiley & Sons, Ltd.
3. Digital Image Processing and Analysis - B. Chanda and D. Dutta Majumder; Prentice Hall of India, New Delhi.
4. Fundamentals of Digital Image Processing - A. Jain; Prentice Hall of India, New Delhi.

MJCS81P: Image Processing (Lab)

Based on Paper **MJCS81T**.

MJCS82T: Machine Learning (Theory)

Introduction to Machine Learning: Overview of machine learning: definitions, types, applications; Machine learning pipeline: data preprocessing, feature engineering, model training, evaluation, deployment.

Supervised Learning: Regression: linear regression, polynomial regression; Classification: logistic regression, Naive Bayes, Decision Tree, k-nearest neighbors (KNN), support vector machines (SVM); Model evaluation metrics: accuracy, precision, recall, F1-score, ROC curves.
Unsupervised Learning: Clustering: k-means, hierarchical clustering; Dimensionality reduction: Principal Component Analysis (PCA); Anomaly detection techniques; Model evaluation metrics: Cluster Validity Index.
Neural Networks and Deep Learning: Introduction to artificial neural networks; Deep learning architectures: convolutional neural networks (CNN), recurrent neural networks (RNN), Transfer learning and pre-trained models.
Reinforcement Learning: Basics of reinforcement learning; Markov Decision Processes (MDPs); Q-learning and policy gradient methods.

References:

1. Pattern Recognition and Machine Learning - Christopher M. Bishop; Springer.
2. Introduction to Machine Learning - Alpaydin Ethem; PHI Learning.
3. Deep Learning - Aaron Courville, Ian Goodfellow, Yoshua Bengio; MIT Press.
4. Reinforcement Learning: An Introduction - Richard S. Sutton, Andrew G. Barto, Francis Bach; MIT Press.

MJCS82P: Machine Learning (Lab)

Based on Paper **MJCS82T**.

MNCS04T: Basics of Python Programming (Theory)

Setting up Python and IDEs, Running basic Python programs.
Implementing Conditional statements (if, elif, else); Looping (for, while) and iteration techniques; Functions: parameters, return values, and scope; Lambda functions and built-in functions, Recursion and recursive functions.
Working with Data Structures and Comprehensions: Lists, tuples, sets, and dictionaries; List comprehensions and generator expressions; Advanced dictionary techniques; Manipulating strings and regular expressions.
Reading from and writing to files, Handling exceptions and errors, Using context managers for file operations, Working with text and binary files.
Implementing classes and objects, Creating inheritance and polymorphism examples, Practicing encapsulation and abstraction principles.

Object-Oriented Programming (OOP): Introduction to OOP principles and concepts; Defining classes and creating objects; Inheritance, composition, and polymorphism; Advanced OOP topics: abstract classes, interfaces.
Working with external APIs and web services, Serializing and deserializing data with JSON, XML, and Pickle, Visualizing data using Matplotlib and Seaborn.
Introduction to data visualization with libraries like Matplotlib and Seaborn; Introduction to databases, SQLite, and basic SQL operations.

References:

1. Python: The Complete Reference - Martin C. Brown; McGraw Hill Education.
2. Think Python: How to Think Like a Computer Scientist - Allen B. Downey; O'Reilly.
3. Introduction to Computation and Programming Using Python: With Application to Understanding Data - John V Guttag; MIT Press.
4. Fluent Python: Clear, Concise, and Effective Programming - Luciano Ramalho; O'Reilly.

MNCS04P: Basics of Python Programming (Lab)

Based on Paper **MNCS04T**.

CSR81: Research Project II

CSEC8AYT: Elective II (Theory)

CSEC8AYP: Elective II (Lab)

Based on Paper **CSEC8AYT**.

CSEC8BZT: Elective III (Theory)

CSEC8BZP: Elective III (Lab)

Based on Paper **CSEC8BZT**.

I. List for Elective I (Choose any one)

CSEC700T: Cyber Security

Introduction: Overview of Cyber Security, Challenges and Constraints, Cyber of internet governance; Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage; Need for a Comprehensive Cyber Security Policy.
Cyber Security Vulnerabilities: Vulnerabilities in software, System administration, Complex Network Architectures, Open Access, Weak Authentication, Unprotected Broadband communications.
Cyber Security Safeguards: Overview, Access control, Audit, Authentication, Biometrics, Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Security policy, Threat Management.
Securing Web Application, Services and Servers: Basic security for HTTP Applications and Services, Basic Security for SOAP Services Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges.
Application Layer security: PGP, S/MIME.
Cyber Laws: Cyber Security Regulations, Roles of International Law, Cyber Security Standards, Cyber Security Policy 2013; overview of cyber forensics.

References:

1. Cyber Security and Cyber Laws - Alfred Basta, Nadine Basta, et al; Cengage India Private Limited.
2. Cyber Security - Nina Godbole and Sunit Belapure; Wiley India.
3. Introduction to Information Security and Cyber Laws - Surya Prakash Tripathi, Ritendra Goel, et al; Dreamtech Press.

CSEC701T: Human-Computer Interaction

Introduction: Objective and overview, Historical evolution of the field.
Interactive system design (theory and practice): Concept of usability definition and elaboration, HCI and software engineering, GUI design and aesthetics, Prototyping techniques.
Model-based design and evaluation: Basic idea, types of models, GOMS family of models (KLM and CMN-GOMS), Fitts' law and HickHyman's law, Case studies of Model-based design.
Guidelines in HCI: Shneiderman's eight golden rules, Norman's seven principles, Norman's model of interaction, Nielsen's ten heuristics with example of its use, Heuristic evaluation, Contextual inquiry, Cognitive walk through.

Empirical research methods in HCI: Introduction (motivation, issues, research question formulation techniques), Experiment design and data analysis (with explanation of one-way ANOVA).
Task modeling and analysis: Hierarchical task analysis (HTA), Engineering task models and Concur Task Tree (CTT).
Dialog Design: Introduction to formalism in dialog design, design using FSM (finite state machines), State charts and (classical) Petri Nets in dialog design.
Cognitive architecture: Introduction to CA, CA types,relevance of CA in IS design, Model Human Processor (MHP).
Object Oriented Programming: OOP - Introduction, OOM - Object Oriented Modeling of User Interface Design.
Design Case Studies: Case Study 1, Case Study 2.

References:

1. Human Computer Interaction - Dix A., Finlay J.,Abowd G. D. and Beale R; Pearson Education.
2. Human Computer Interaction - Preece J., Rogers Y.,Sharp H.,Baniyon D., Holland S. and Carey T; Addison-Wesley.
3. Designing the User Interface - B.Shneiderman; Addison-Wesley.

CSEC702T: Imprecise Mathematics

Fuzzy logic: Review of fuzzy mathematics, Higher type fuzzy sets, applicability.
Rough mathematics: Introduction, Information system, Indiscernibility, Rough sets, Rough set theory, Set approximation, Rough membership, Attributes, Dependency of attributes, Rough equivalence, Reducts, Rough Reducts based on SVM.
Vague mathematics: Vague set theory, Vague set approximation, Vague membership, Attributes, Rough equivalence.
Hybridization: Introduction to Hybrid AI systems, Neuro-Fuzzy, Coactive Neuro Fuzzy Modeling, Fuzzy-rough set systems, Topological structures of rough sets over fuzzy lattices, Neuro-Fuzzy-GA systems and case studies of Hybrid systems.

References:

1. Rough set data analysis: A Road to Non-invasive Knowledge Discovery - Duntsch, I and Gediga, G; Methodos Publishers.
2. Fuzzy Sets and Fuzzy Logic, Theory and Applications - George J. Klir and Bo Yuan; Prentice Hall of India.

3. Fuzzy Set Theory and its Application - H. J. Zimmermann; Allied Publishers.
4. Vague sets - W.L. Gau, D.J. Buehrer; IEEE Transactions on Systems, Man and Cybernetics, 23(2)(1993), pp. 610-614.

CSEC703T: Information Theory

Information Theory: Quantification of information, concept of entropy, computation of entropy of symbol sources, joint and conditional entropies.
Lossless Source Coding: Entropy encoding - Source coding theorem, Prefix Codes, Kraft MacMillan Inequality, Shannon-Fano coding, static and dynamic Huffman coding, Vitter's algorithm; dictionary based coding - LZ77, LZ78 and LZW algorithm.
Information Channels: Joint and conditional entropy between input and output, K-L distance and mutual information, Binary symmetricity, discrete memoryless and binary erasure channel, channel capacity, Shannon's channel capacity theorem.
Channel Coding Techniques: Minimum free distance of coding algorithm, block codes - Hamming codes, convolution code and Hard decision and soft decision Viterbi decoding, Turbo code, overview of LDPC and Polar codes.

References:

1. Coding Theory: A First Course - San Ling and Chaoping; Cambridge University Press.
2. Information and Coding Theory - A Jones and J M Jones; Springer Verlog.
3. Communication Systems - Simon Haykin; John Wiley & Sons, Inc.

CSEC704T: Internet of Things

Introduction : Basics of analog electronics, types of diodes and transistors, template amplifier and switching circuits, batteries and power supply modules - BMS and buck and boost converters, voltage and current ratings.
Arduino board types and IDE - Arduino R3/R4, ESP32/EP8266 boards, Pin modes and differences, timers and interrupts; interface types - serial, I2C, SPI; types of sensors and libraries, actuators and libraries, Programming sensor-actuator systems using single microcontroller.
Wireless Sensor Networks - Communication between different sensor-actuator modules using ESPNow, NRF, WiFi, bluetooth, cellular modems, WPAN technologies.
IP Based Protocols for IoT - IPv6, 6LowPAN, AMPQ, CoAP, MQTT, Edge connectivity and protocols.
Raspberry Pi - Using Raserry PI GPIO, implementation of sensor actuator systems using Rasberry pi, introduction to ROS.

References:

1. Internet of things - A Hands-on Approach - Arshdeep Bahga, Vijay Madisetty; University Press.
2. IoT for Beginners - Vibha Soni; BPB publications.
3. Internet of Things - Shriram Vasudevan, Abhishek S Nagarajan, RMD Sundaram; Wiley Emerging Technology Series.

CSEC705T: Modelling and Simulation

Introduction : Concepts in discrete-event system simulation, event scheduling, components of a simulation model, examples.
Overview of basic concepts from probability and statistics concerning random variables, correlation, estimation, confidence intervals, hypothesis testing.
Generation of random numbers : Random number generators, congruential generators - issues and challenges, testing of random numbers, generation of random variates and random vectors, correlated random variables and stochastic processes.
Queuing systems : Characteristics, performance measures and analysis, simulation of single server and multi-server queuing systems.
Input modelling : Useful probability distributions, estimation of parameters, goodness-of-fit test and assessing sample dependence, Multivariate input models.
Output Analysis : Output analysis of a single system, statistical analysis of transient systems and systems in statistical equilibrium, comparing alternate system configurations, precision, confidence intervals, ranking and selection.
Monte Carlo Simulation : Introduction and problem solving using Monte-Carlo simulation and examples.

References:

1. Simulation Modeling and Analysis - A. M. Law and W. D. Kelton; McGraw Hill International - Industrial Engg. Series.
2. Discrete Event System Simulation - J. Banks, J. S. Carson, B. L. Nelson and D. M. Nicol; Pearson Education International Series.
3. Probability and Statistics with Reliability, Queuing and Computer Science Applications - K. S. Trivedi; Eastern Economy Edition, Prentice-Hall (India).

CSEC706T: Numerical Methods

Introduction : Floating point representation and arithmetic, significant digits, Errors - round-off error, global truncation error, convergence and terminal conditions.
Interpolation and extrapolation : Lagrange's form, Newton's form, Finite difference operator, Gregory Newton forward and backward difference, Linear, piecewise linear and piecewise polynomial interpolation, Cubic-spline interpolation.
Finding roots of equations : Bisection Methods, Secant method, regula-falsi method, Newton-Raphson method.
Numerical differentiation and Integration : Differentiation - First and second order derivatives, Richardson's extrapolation formula; Integration - Trapezoidal Rule, Simpson's 1/3rd and 3/8 rule, Newton-Cote open formulae, Romberg integration, Gaussian quadrature.
Ordinary differential equation : Euler's method, Finite difference method for linear ODE, modified Euler's method, Heun method, Mid-point method, Ralston's method, Runge-Kutta 2nd and 4th order methods.

References:

1. Applied Numerical Analysis Using MATLAB - Laurence V. Fausett; Pearson.
2. Numerical Methods for Scientific and Engineering Computation - M.K. Jain, S.R.K. Iyengar and R.K. Jain; New Age International Publisher.
3. Applied Numerical Methods with MATLAB for Engineers and Scientists - Steven C Chapra; Tata McGraw Hill.

CSEC707T: Operations Research

Introduction: Features of OR Approach. Different Models, Methods for Solving OR Models. Notion of Convex set.
Linear Programming: Structure of LP Model, LP Model Formulation, Graphical Method Extreme Point Solution Method, Simplex Method Maximization case and Minimization case, Two-Phase Method and Big-M Method.
Duality: Primal and Dual, Rules for Constructing the Dual from Primal, Interpretation of Dual Variables and Constraints, Comparisons of the Solutions, Advantages of Duality. Revised Simplex Method: Computational Procedure, Comparison with Simplex Method.
Dual-Simplex Method: Dual-Simplex Algorithm to Solve some LP Problems.
Integer Linear Programming: Types of ILP, Cutting Plane Method, Branch and Bound Method, Zero-One ILP.
Transportation Problem: Balanced and Unbalanced TP, General Mathematical Model of TP, The Transportation Algorithm, North-West Corner Method, Least Cost Method, Vogel's Approximation Method, Optimality Testing, Dual of Transportation Model, MODI Method, Close-Loop.

Assignment Problem: Balanced and Unbalanced AP, Mathematical Model of an AP, Hungarian Method, Variations of the AP.

References:

1. Operations Research Theory and Applications - J.K. Sharma; Trinity Press.
2. Operations Research Principles and Applications - G.Srinivasan; PHI Learning Private Limited.
3. Operations Research An Introduction - Hamdy A. Taha; Pearson.
4. Operations Research Principles and Practice Ravindran, Phillips and Solberg; Wiley India.
5. Operations Research Concepts and Cases - Hillier and Liberman; McGraw Hill.

CSEC708T: Quantum Computation and Quantum Information

Introduction: History of quantum computation and quantum information, Very brief Introduction in Quantum Mechanics, Quantum Mechanics versus Classical Mechanics, latest status of quantum computer and quantum information.
What is Quantum bits(qubits)? Representation of qubit, qubits versus classical bits, multiple qubits.
Introduction to Quantum Mechanics: Bases and linear independence, Linear operator and Matrices, Pauli matrices, Inner product, Eigenvectors and Eigen values, Hermitian operators, Properties of Hermitian operators.
The postulates of Quantum Mechanics, State Space, Superposition of states, Schrodinger CAT; Evolution, Quantum Measurement; EPR Bell sates, Bell inequality.
Entangle sates Quantum entanglement, No cloning theorem, Quantum Teleportation and super dense coding.
Quantum Gates: Classical Gates and Quantum Gates, Quantum Gates, Matrix representation of Quantum Gates, Unitarity, Universal Quantum Gates, Quantum Circuits and its matrix representation, Quantum Registers, entangle registers.
Fourier Transform, Quantum Fourier Transform and circuit representation of quantum Fourier Transformation order finding, factoring.
Quantum Algorithm: Single qubit operation, controlled operations, Grover's Search algorithms, discussion on complexity, Shor algorithm, discussion on complexity.
Quantum Cryptography: Quantum Cryptography and Classical cryptography, Quantum Key distributions, comparison with RSA cryptosystems, quantum communication , BB84 Protocol, E91 protocol, Post quantum cryptography.

References:

1. Quantum Computation and Quantum Information - M.A Nielsen & I L Chuang; Cambridge University Press.
2. Quantum Computing - J Gruska; McGraw Hill.

CSEC709T: VLSI Design

Introduction to VLSI Design : Basic terminology and complexity issues, NP-hard problems and basic algorithms.
Data Structures for VLSI: Basic data structures (linked list, bin-based method), Specialized data structures (corner stitching, multi-layer operations), Atomic operations for layout editors, Limitations of existing data structures, Layout specification languages.
Graph Algorithms in Physical Design: Classes of graphs in VLSI, Graph algorithms for physical design, Interval, permutation, and circle graphs.
Partitioning Algorithms: Problem formulation, Partitioning algorithms- Kernighan-Lin algorithm, group migration, simulated annealing, Performance-driven partitioning.
Floorplanning and Pin Assignment: Problem formulation, Floorplanning algorithms: constraint based, integer programming, hierarchical methods, Pin assignment strategies: general and channel pin assignment, Timing-driven and performance-driven techniques.
Placement Algorithms : Problem formulation and design-specific placement, Simulation-based algorithms: simulated annealing, force-directed placement, Partitioning-based placement: Breuer's algorithm, terminal propagation, Performance-driven placement.
Global and Detailed Routing: Global routing: classification of algorithms, maze and Steiner tree-based approaches, Detailed routing: channel and switchbox routing algorithms, Single-layer and multi-layer routing algorithms.

References:

1. Algorithms for VLSI Physical Design Automation - Naveed Sherwani; Springer-Verlag NY.
2. VLSI Physical Design Automation: Theory and Practice - Sadiq M. Sait, Habib Youssef; IEEE press.
3. Graph Theory and Algorithms for VLSI Design - C. L. Liu and James B. Muirhead; John Wiley & Sons.

II. List for Elective II (Choose any one)

CSEC810T: Algorithmic Graph Theory

Introduction to Graph Theory: Basic Definitions and Notations, Intersection Graphs, Circular-arc Graphs, Interval Graphs, Line graphs of bipartite graphs, Chromatic number and Chromatic polynomial.
Perfect and Triangulated Graphs: Definition of perfect graph, Perfect Graph Theorem, p-Critical Graphs, A Polyhedral Characterization of Perfect Graphs, The Strong Perfect Graph Conjecture, Characterizing Triangulated Graphs, Recognizing Triangulated Graphs, Time Complexity, Generating a PEO, Testing an Elimination Scheme, Triangulated Graphs Are Perfect, Some Optimization Algorithms on Triangulated Graphs.
Comparability Graphs: Implication Classes, The Triangle Lemma, Decomposition of Graphs, Uniquely Partially Orderable Graphs, Comparability Graph Recognition.
Some interesting graph families characterized by intersection: Introduction, Permutation graphs, F- Graphs, Tolerance graphs, Bounded and unbounded tolerance graphs.
Flow Network: Basic properties of a flow, Max Flow - Min Cut theorem, Ford Fulkerson's algorithm, and Edmond Karp's algorithm.

References:

1. Algorithmic Graph Theory and Perfect Graphs - Martin Charles Golumbic; Academic Press.
2. Advanced Topics in Graph Algorithms - Ron Shamir; http://www.cs.tau.ac.il/∼rshamir/atga/atga.html.
3. Network Flows: Theory, Algorithms, and Applications - Ravindra K. Ahuja, Thomas L. Magnanti, James B. Orlin; Pearson.

CSEC811T: Android Programming

Introduction to android : Development environment setup, Dalvik virtual machine and .apk file extension, fundamentals - basic building blocks, UI components and components for communication; Emulator - Launching, editing settings, shortcuts.
UI design - Form widgets, text fields, layouts, preferences, menu, intents, styles and themes.
Content providers and views - SQLite programming, SQLite OpenHelper, SQLite Database, Cursor and related topics, Adapters and Widets, ListView, GridView, Gallery.
Threads - UI thread, Worker thread, runnable and handlers, AsyncTask.
Advanced topics - Multimedia files and stream handling, interfacing with Google Map, Sensors, WiFi, Controlling the phone, telephonic and data activity, camera and bluetooth.

References:

1. Fundamentals of Android App Development – Sujit Kumar Mishra; BPB Publications.
2. Android Application Development with Kotlin – Hardik Trivedi; BPB Publications.
3. Android Programming Concepts - Trish Cornez, Richard Cornez; Jones and Bartlett.

CSEC812T: Bioinformatics

Introduction, branches, aim, scope, research areas.
The genetic material: nucleotides, orientation, base pairing, central dogma.
Gene Structure: Promoter sequence, Genetic code, Introns and exons.
Pairwise Alignment: Gaps, dynamic programming, Needleman and Wunsch Algorithm, Smith-Waterman algorithm.
Databases in Bioinformatics: Structures - sequence and molecular file formats, conversion tools, databases, classification schema, retrieval systems.
Sequence Databases: Nucleotide sequence databases, secondary nucleotide, protein sequence databases, secondary and specialized protein sequence databases.
Data Analysis Tools: Introduction to BLAST, PSI-BLAST.
Data visualization in proteins using RasMol/Chime.

References:

1. Bioinformatics: Databases, Tools and Algorithms - O Basu and S K Thukral; Oxford University Press.
2. Bioinformatics: Principle and Applications - Z Ghosh and B Ballick; OUP India.
3. Fundamental Concepts of Bioinformatics - D E Krane and M L Raymer; Pearson Education India.
4. Bioinformatics: A Modern Approach - V R Srinivas; Prentice Hall India.

CSEC813T: Cloud Computing

Overview of Computing Paradigm - Recent trends in Computing, Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing.
Introduction to Cloud Computing - Introduction to Cloud Computing, History of Cloud Computing, Cloud service providers, Benefits and limitations of Cloud Computing. Cloud Computing Architecture - Comparison with traditional computing architecture (client/ server), Services provided at various levels, Service Models - Infrastructure as a Service(IaaS), Platform as a Service(PaaS), Software as a Service(SaaS), How Cloud Computing Works, Deployment Models- Public cloud, Private cloud, Hybrid cloud, Community cloud, Case study of NIST architecture. Case Studies – Case study of Service model using Google App Engine, Microsoft Azure, Amazon EC2 , Eucalyptus. Service Management in Cloud Computing - Service Level Agreements(SLAs), Billing & Accounting, Comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling. Cloud Security - Infrastructure Security- Network level security, Host level security, Application level security, Data security and Storage- Data privacy and security Issues, Jurisdictional issues raised by Data location, Authentication in cloud computing.

References:

1. Cloud Computing Bible - Barrie Sosinsky; Wiley-India.
2. Cloud Computing: Principles and Paradigms - Rajkumar Buyya, James Broberg, Andrzej M. Goscinski; Wiley.
3. Cloud Security: A Comprehensive Guide to Secure Cloud Computing - Ronald L. Krutz, Russell Dean Vines; Wiley-India.

CSEC814T: Cryptography and Network Security

Introduction: Basic concepts of confidentiality, integrity, authentication.
Basic Cryptography: Historical background, transposition/substitution, Caesar cipher, introduction to symmetric crypto primitives, asymmetric primitives and hash functions.
Secret key cryptography: Applications, DES, Message Digests.
Public Key cryptography: Euclidean algorithm, Euler theorem, Fermat Theoretical functions, multiplicative and additive inverse; RSA algorithm, Elliptic Curve, Knap-Sack algorithms.
Authentication: Certification authorities and key distribution centre, digital signatures.
Hash functions: MD4, MD5 message digest algorithm, Secure Hash algorithm, HMAC digital signatures.
Network and E-mail Security: Kerberose -X.509 authentication service, PGP, S/MIME, IPSec.
System level security: Intrusion detection, password management, Viruses and other Malware

threats, Firewalls, Trusted Systems, Access Control.

References:

1. Network Security and Cryptography – William Stallings; Pearson Education.
2. Computer Security: Art and Science – Matt Bishop; Pearson Education.
3. Cryptography And Network Security - Atul Kahate; McGraw Hill.

CSEC815T: Introduction to Data Sciences

Concept of data, information. Representation of data – histogram (absolute and relative), concept of probability, data distribution, Random Variables – continuous and discrete, probability mass function, probability density function, probability distribution function.
Concept of correlation (Pearson, rank), concept of co-linearity, regression (linear and nonlinear).
Prediction and forecasting.
Classification of data, Learning – supervised, unsupervised, self supervised, reinforced, Various training-learning models, Classifiability – linear, non-linear.
Various frequently used classifiers – Perceptron, Radial Basis Function, Support Vector Machines.
Clustering of data, Various frequently used clustering techniques, K-means, Fuzzy c-means.
Challenges and tackling outliers.

References:

1. Foundations of Data Science - Avrim Blum, John Hopcroft, Ravindran Kannan; Cambridge University Press.
2. Big Data: A Revolution That Will Transform How We Live, Work, and Think Paperback – Viktor Mayer-Schönberger, Kenneth Cukier; Harper Business.

CSEC816T: Parallel Algorithms

Basic Concepts: Basic definitions and terminologies.
Models of parallel computation: DAG, PRAM, interconnection networks etc.
Performance of parallel algorithms, Basic algorithm design techniques.
Sorting: Odd Even Merge Sort, Bitonic Sort, Parallel merge sort, Sorting in meshes, Hypercubes, Butterfly networks, CCC networks, Cole's Merge Sort.
Searching Algorithms.

Optimal List ranking and applications, Tree Traversal and related parallel algorithms, Euler tool technique.
Graph Algorithms: Connected Components, Minimum Spanning Tree, Shortest paths.
Limits to parallelizability. Lower bounds.
Limits to parallelizability. NC-Reductions, P-Completeness.

References:

1. Parallel Algorithms - Joseph Jaja; OUP India.
2. Design and Analysis of Parallel Algorithms - S G Akl; PHI.
3. Parallel Computing: Theory and Practice - M J Quinn; Tata-McGraw Hill.

CSEC817T: Pattern Recognition

Basic Mathematical and Statistical concepts: Metric, Positive definite matrix, mean, median, mode, variance, co-variance, correlation, dispersion matrix, binomial distribution, normal distribution, multi-variate normal distribution, basic concepts in probability theory such as Bayes theorem, Chebyshev's inequality, Laws of large numbers, Central limit theorem, Unbiased estimate, consistent estimate, maximum likelihood estimate.
Classification: Bayes decision rule, normal distribution cases, training and test sets, prob. of misclassification, estimation of parameters for normal distribution, minimum distance classifier, standardization, normalization, Mahalanobis distance, Naive-Bayes rule, K-NN decision rule, its properties, Density estimation, Perceptron (linear separable case), MLP, Assessment of classifiers.
Clustering: Similarity measures, minimum within cluster distance criterion, K-means algorithm, Hierarchical clustering, Density based clustering, FCM, cluster validation.
Dimensionality reduction: Feature selection: Different criterion functions, Algorithms, BBA. Feature extraction: PCA, LDA Decision trees, Random forests.

References:

1. Pattern Classification - Richard O. Duda, Peter E. Hart and David G. Stork; Wiley.
2. The Elements of Statistical Learning (2nd Edition) - Trevor Hastie Robert Tibshirani and Jerome Friedman; Springer.
3. Pattern recognition and Machine Learning - Christopher M. Bishop; Springer.

CSEC818T: Randomized Algorithms

Introduction: Las Vegas Algorithm, Monte Carlo Algorithm, Computation Models and Probabilistic Classes.
Game-Theoretic Techniques: Game Tree Evaluation, Minmax Principle.
Moments and Deviations: Markov and Chebyshov's Inequality, Randomized Selection,Two-point Sampling, Stable Marriage Problem.
Probabilistic Methods: Max Satisfiability, Lovasz Local Lemma, Method of Conditional Probability.
Markov Chains: Markov Chains and Random Walk on Graphs.
Algebraic Techniques: Finger Printing and Freivalds' Technique, Polynomial Identity Verification,Perfect Matching, Verifying Equality of Strings, Comparison of Fingerprinting Techniques, Pattern Matching, PCP and Efficient Proof Verification.
Graph Algorithms: All pair Shortest Paths, Min-cut, Minimum Spanning Trees.

References:

1. Randomized Algorithms - Rajeev Motwani and Prabhakar Raghavan; Cambridge University Press.
2. Design and Analysis of Randomized Algorithms - J. Hromkovic; Springer.

CSEC819T: Wireless Networks

Evolution of Cellular Networks: Standards, Generic architecture of PLMN, CN and RAN split within PLMN; overview of network selection, registration, IP packet flow, idle mode mobility, connected mode mobility and handovers.
GPRS CN and RAN Architecture over GSM and UMTS: PMM, SM and RABM procedures in CN; QoS classes; RAN architecture and functions, channel structures.
Evolution of LTE systems from UMTS: Optimizations in CN and RAN architectures, CN procedures for mobility and bearer service managements, inter-system handovers.
Basics of LTE RAN: OFDMA in downlink and SC-FDMA in uplink, physical resource blocks; physical channels, transport channels and RB mapping; MAC and RLC functions; RRC procedures - cell acquisition, RRC connection management, bearer services, handovers.
LTE-Advanced features and evolution towards 5G systems.
Wi-Fi systems - 802.11x standards, air interface, authentication.

References:

1. WCDMA for UMTS: HSPA Evolution and LTE - Harri Holma, Antti Toskala; John-Wiley.

2. LTE – The UMTS Long Term Evolution - From Theory to Practice - Stefania Sesia, Issam Toufik, Matthew Baker; John-Wiley.

III. List for Elective III (Choose any one)

CSEC820T: Advanced Graphics

3D Object Representation: Polyhedra and Different Quadratic Surfaces, Splines.
3D Viewing: 3D Transformation, Projections: Perspective and Parallel.
Surface Determination: Visible Surface and Hidden Surface Determination Algorithms, Wire-frame Algorithm, Polygon Cully.
Light and Shading: Light Source, Basic Elimination Models, Ambient Light, Ray Tracing Methods, Different Shading Models.
Texture Mapping: Linear, Surface and Volume Texture Graphics.

References:

1. Computer Graphics with OpenGL - Donald Hearn and M. Pauline Baker; Pearson.
2. Procedural Elements for Computer Graphics - David F. Rogers; Tata McGraw Hill.
3. Mathematical Elements for Computer Graphics - David F. Rogers, J. Alan Adams; Tata Mc-Graw Hill Education.

CSEC821T: Approximation Algorithms

Introduction: Review of theory of NP-Completeness. Need for approximation algorithms, Basic definitions and terminologies.
Design Techniques: Concept of lower bound and its use, Minimum spanning tree based approximation algorithms, Greedy method, Recursive greedy method, LP based design techniques, Randomized approximation algorithm design techniques.
Approximation Schemes: Design techniques of PTAS, FPTAS.
Hardness of Approximation: Approximation preserving reductions, Classes APX, APX-Hard, APX-Complete, Gap introducing reductions, Gap preserving reductions, PCP theorem.

References:

1. The Design of Approximation Algorithms - Williamson and Shmoys; Cambridge University Press.
2. Approximation Algorithms - V V Vazirani; Springer Verlag.

CSEC822T: Big Data Analytics

Data Definitions and Analysis Techniques: Elements, Variables, and Data categorization, Measurement, Data management and indexing, Introduction to statistical learning and R-Programming.
Descriptive Statistics: Measures of central tendency, Measures of location of dispersions, Practice and analysis with R.
Basic analysis techniques: Statistical hypothesis generation and testing, Chi-Square test, t-Test, Analysis of variance, Correlation analysis, Maximum likelihood test, Inferential Statistics through hypothesis tests, Permutation and Randomization Test, Regression, ANOVA (Analysis of Variance), Practice and analysis with R.
Data analysis techniques: Regression analysis, Classification techniques, Clustering, Association rules analysis, Practice and analysis with R. Prescriptive analytics: Creating data for analytics through designed experiments, Active learning, and Reinforcement learning.
Big Data: Distributed file system, Big Data and its importance, Big data applications, Algorithms using MapReduce, Matrix-Vector, Multiplication by MapReduce, Apache Hadoop and Hadoop Eco System, Moving Data in and out of Hadoop, Understanding inputs and outputs of Map Reduce Data Serialization.
Case studies: Understanding business scenarios, Feature engineering and visualization, Scalable and parallel computing with Hadoop and MapReduce, Sensitivity Analysis.

References:

1. The Elements of Statistical Learning - Trevor Hastie, Robert Tibshirani, Jerome Friedman; Vol. No. 1. New York: Springer.
2. Applied Statistics and Probability for Engineers - Montgomery, Douglas C., and George C. Runger; John Wiley and Sons.

CSEC823T: Complex Networks

Introduction: Review of topics in probability and linear algebra, examples of real-world networks and their properties.
Fundamentals of network theory: Adjacency matrix, weighted graphs, directed graphs, hypergraphs, bipartite graphs, trees, planar graphs, degree, paths, diameter, components, independent paths, connectivity, cut sets, maximum flows and cut sets on weighted graphs, graph Laplacian, random walks, Centrality measures - degree centrality, closeness centrality, betweenness centrality, eigenvector centrality, Katz centrality, hub and authority centrality, Cliques, plexes, cores, clustering coefficient, Similarity measures - cosine similarity, Pearson coefficient, Katz similarity, Assortative and disassor tative mixing, Shortest paths and the

small-world effect, degree distributions, scale-free and power laws.
Fundamental network algorithms: degree distributions, clustering coefficients, shortest paths and breadth-first search, betweenness centrality, maximum flows and minimum cuts, spanning trees, independent paths, minimum cut sets.
Matrix algorithms and graph partitioning: eigenvectors and eigenvector centrality, graph partitioning and community detection/clustering, spectral partitioning.
Random graphs and network formation: Erdos-Renyi random graphs, tree structure, giant component, fixed degree distributions, configuration model, small-world (Watts-Strogatz) model, exponential random graphs, Markov graphs, network growth, preferential attachment, Barabasi-Albert model, power-law networks.
Dynamics on networks: Random walks, diffusion, Epidemics/contagion, mean-field models.

References:

1. Networks: An Introduction - M. E. J. Newman; Oxford University Press.
2. Complex Graphs and Networks - F. Chung and L. Lu; CBMS Regional Conference Series in Mathematics.
3. Scale-Free Networks - Guido Caldarelli; Oxford University Press.
4. Random Graph Dynamics – R. Durrett; Oxford University Press.

CSEC824T: Computer Vision

Machine vision systems, introduction to low, mid and high level vision, low and mid level image processing, edge detection, image segmentation, image and texture features Camera geometry, object to image geometric transformations, orthographic and perspective view transformations, camera calibration Binocular vision system, epipolar geometry, 3D scene reconstruction, recovering shape from stereo Human vision structure, neurovisual model, scale space representation Motion estimation and tracking, active contours, recovering shape from motion, video processing Reflectance map and photometric stereo, surface reflectance model, recovering albedo and surface orientation, recovering shape from shading Machine learning for computer vision, Classification models for vision, deep learning architectures for vision, Model based recognition system.
Object recognition, recognition of arbitrary curved object sensed either by stereo or by range sensor, Recognition under occlusion, Aspect graph of an arbitrary 3D object viewed from different directions, Recognition of 3D objects based on 2D projections Projects and Assignments: At least two assignments and one class project, assignments should include implementation of computer vision algorithm using a programming language.

References:

1. Computer Vision, A Modern Approach - David Forsyth, et al.; Prentice Hall Pearson.
2. Computer Vision: Algorithms and Applications - Richard Szeliski; Springer.
3. Multiple View Geometry in Computer Vision - Richard Hartley, et al., Andrew Zisserman; Cambridge University Press.
4. Robot Vision - B.K.P. Horn; MIT Press, Cambridge.

CSEC825T: Data Mining

Basic concepts: Knowledge discovery in database, data mining, distinction, basic approaches – association, classification, clustering, networking, soft computing approaches, data cleaning, privacy preservation.
The Architecture of BI and DW: BI and DW architectures and its types - Relation between BI and DW, OLAP (Online analytical processing) definitions, Difference between OLAP and OLTP, Dimensional analysis, What are cubes? Drill-down and roll-up, slice and dice or rotation, OLAP models, ROLAP versus MOLAP, defining schemas: Stars, snowflakes and fact constellations.
Preprocessing of data: Reduction of dimensionality, data coding, discovering structural relationships in data: Rules and trees, Descriptive analytics: Density estimation, anomaly detection, Predictive analytics: Regression.
Association rule discovery: Apriori and FP-growth algorithm, FP-tree algorithm, GA based rule mining, applications in gene expression analysis, genome-wide association studies.
Classification and Clustering: CART, ensemble classifier, Random-Forest, GA-based clustering, Clus- tering High-Dimensional Data, Constraint Based Cluster Analysis – Outlier Analysis, bi- and tri- clustering, GA-based biclustering.
Case Studies: Mining Time-Series and Sequence Data, Mining Text Databases, Mining the World Wide Web, Recommendation System, Data Mining Application.

References:

1. Data Mining: Concepts and Techniques - Jiawei Han, Jian Pei and Micheline Kamber; Elsevier.
2. Introduction to Data Mining - Pang-Ning Tan, Vipin Kumar and Michael Steinbach; Pearson.
3. Data Mining: Introductory and Advanced Topics - Margaret H Dunhum; Pearson India.

CSEC826T: Deep Learning

Introduction - Feedforward Neural networks, Gradient descent and the backpropagation algorithm, Unit saturation, aka the vanishing gradient problem, and ways to mitigate it, RelU Heuristics for avoiding bad local minima, Heuristics for faster training, Nestors accelerated gradient descent, Regularization, Dropout.
Convolutional Neural Networks - Architectures, convolution / pooling layers.
Recurrent Neural Networks - LSTM, GRU, Encoder Decoder architectures.
Deep Unsupervised Learning - Autoencoders (standard, sparse, denoising, contractive, etc.), Variational Autoencoders, Adversarial Generative Networks, Autoencoder and DBM.
Attention and memory models, Dynamic memory networks.
Applications of Deep Learning to Computer Vision - Image segmentation, object detection, automatic image captioning, Image generation with Generative adversarial networks, video to text with LSTM models, Attention models for computer vision tasks.
Applications of Deep Learning to NLP - Introduction to NLP and Vector Space Model of Semantics, Word Vector Representations: Continuous Skip-Gram Model, Continuous Bag-of-Words model (CBOW), Glove, Evaluations and Applications in word similarity, analogy reasoning, Named Entity Recognition, Opinion Mining using Recurrent Neural Networks, Parsing and Sentiment Analysis using Recursive Neural Networks, Sentence Classification using Convolutional Neural Networks, Dialogue Generation with LSTMs, Applications of Dynamic Memory Networks in NLP, Recent Research in NLP using Deep Learning: Factoid Question Answering, Similar question detection, Dialogue topic tracking, Neural Summarization, Smart Reply.

References:

1. Deep learning - Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Cambridge: MIT press.
2. Neural networks and deep learning - Michael A Nielsen, USA: Determination press.
3. Deep Learning: A Practitioner's Approach - Adam Gibson and Josh Patterson, O'Reilly Media, Inc.
4. Deep Learning: Foundations and Concepts - Christopher M. Bishop, Hugh Bishop, Springer-Nature New York Inc.

CSEC827T: Ethics of AI

Introduction - Definition of morality and ethics in AI, Impact on society, Impact on human psychology, Impact on the legal system, Impact on the environment and the planet, Impact on trust. *Exploration of the concept of "dharma" (righteousness) in Indian philosophy in the*

context of AI ethics.
Ethical Initiatives in AI - International ethical initiatives, Ethical harms and concerns, Case study: healthcare robots, Autonomous Vehicles, Warfare and weaponization.
AI Standards and Regulation - Model Process for Addressing Ethical Concerns During System Design, Transparency of Autonomous Systems, Data Privacy Process, Algorithmic Bias Considerations, Ontological Standard for Ethically Driven Robotics and Automation Systems.
Roboethics: Social and Ethical Implication of Robotics - Robot, Roboethics, Ethics and Morality, Moral Theories, Ethics in Science and technology, Ethical Issues in an ICT Society, Harmonization of Principles, Ethics and professional Responsibility, Roboethics Taxonomy.
AI and Ethics, Challenges and Opportunities - Challenges, Opportunities, ethical issues in artificial intelligence, Societal Issues concerning the Application of Artificial Intelligence in Medicine, decision-making role in industries, and National and International Strategies on AI.

References:

1. The ethics of artificial intelligence: Issues and initiatives - Y. Eleanor Bird, Jasmin Fox-Skelly, Nicola Jenner, Ruth Larbey, Emma Weitkamp and Alan Winfield; EPRS — European.
2. Robot Ethics: The Ethical and Social Implications of Robotics - Patrick Lin, Keith Abney, George A Bekey; The MIT Press.
3. Towards a Code of Ethics for Artificial Intelligence (Artificial Intelligence: Foundations, Theory, and Algorithms) - Paula Boddington; Springer.

CSEC828T: Natural Language Processing

Regular Expressions and Automata: Introduction to NLP, Regular Expression, Finite State Automata.
Tokenization: Word Tokenization, Normalization, Sentence Segmentation, Named Entity Recognition, Multi Word Extraction, Spell Checking – Bayesian Approach, Minimum Edit Distance. *Concept of word segmentation and sandhi-vicheda in Sanskrit texts; early use of compound word decomposition.*
Morphology: Morphology – Inflectional and Derivational Morphology, Finite State Morphological Parsing, The Lexicon and Morpho-tactics, Morphological Parsing with Finite State Transducers, Orthographic Rules and Finite State Transducers, Porter Stemmer. *Paninian morphological analysis using rule-based derivations (prakriyā), kridanta formations, and verb root transformations (dhatu-pratyaya framework).*
Language Modeling: Introduction to N-grams, Chain Rule, Add-One Smoothing, Witten-Bell Discounting; Backoff, Deleted Interpolation, N-grams for Spelling and Word Prediction, Evaluation of language models.

Hidden Markov Models and POS Tagging: Markov Chain, Hidden Markov Models, Forward Algorithm, Viterbi Algorithm, Part of Speech Tagging – Rule based and Machine Learning based approaches, Evaluation. *Rule-based grammatical tagging derived from Paninian system of karaka relations and function-word identification*
Information Retrieval: Boolean Retrieval, Term-document incidence, The Inverted Index, Query Optimization, Phrase Queries, Ranked Retrieval – Term Frequency – Inverse Document Frequency based ranking, Zone Indexing, Query term proximity, Cosine ranking, Combining different features for ranking, Search Engine Evaluation, Relevance Feedback. *Traditional indexing in ancient Indian texts like Nighantu and Koshas (Amarakosha), thematic and semantic organization in classical Indian lexicons*
Text Classification: Text Classification, Naïve Bayes' Text Classification, Evaluation, Sentiment Analysis – Opinion Mining and Emotion Analysis, Resources and Techniques.

References:

1. Speech and Language Processing - Jurafsky and Martin; Pearson Education.
2. Foundation of Statistical Natural Language Processing - Manning and Schutze; MIT Press.
3. Natural Language Processing with Python - Steven Bird, Ewan Klein and Edward Loper; O'Reilly Media.

CSEC829T: Quantum Algorithms

Introduction to Quantum Computing: Basics of Quantum Mechanics (Superposition, Entanglement), Quantum Bits (Qubits) and Quantum States, Quantum Gates and Circuits, Quantum Measurement and No-Cloning Theorem, Quantum Parallelism.
Classical Complexity and Quantum Computing: Overview of Classical Computational Complexity, Comparison between Classical and Quantum Models of Computation, Quantum Circuit Complexity, Introduction to BQP (Bounded Quantum Polynomial Time).
Quantum Gates and Circuits: Single-Qubit Gates (X, Y, Z, H, S, T), Multi-Qubit Gates (CNOT, Toffoli, Controlled-U), Universal Quantum Gates, Quantum Circuit Design, Quantum Fourier Transform (QFT).
Quantum Search Algorithms: Grover's Algorithm for Unstructured Search, Quantum Oracle Concept, Amplitude Amplification, Applications of Grover's Algorithm.
Quantum Algorithms for Number Theory: Shor's Algorithm for Factoring and Discrete Logarithms, Quantum Phase Estimation, Modular Exponentiation in Quantum Circuits, Quantum Algorithms for Order Finding, Implications for Cryptography (RSA, ECC).
Quantum Walk Algorithms: Introduction to Quantum Walks (Discrete and Continuous), Quantum Walks vs. Classical Random Walks, Algorithms Based on Quantum Walks (Ele-

ment Distinctness, Search on Graphs).
Quantum Simulation Algorithms: Simulating Quantum Systems on Quantum Computers, Hamiltonian Simulation and Trotterization, Quantum Chemistry Algorithms, Applications in Physics and Material Science.
Quantum Machine Learning Algorithms (Optional): Introduction to Quantum Machine Learning, Quantum-enhanced Supervised and Unsupervised Learning, Quantum Support Vector Machines (QSVM), Quantum Principal Component Analysis (QPCA).
Error Correction and Fault-Tolerant Quantum Computing: Sources of Errors in Quantum Systems, Introduction to Quantum Error-Correcting Codes (QECC), Shor Code, Steane Code, Fault-Tolerant Quantum Gates and Circuits Threshold Theorem.
Advanced Topics : Quantum Annealing and Adiabatic Quantum Computing, Topological Quantum Computation, Hybrid Quantum-Classical Algorithms (VQE, QAOA).
Quantum Algorithmic Complexity: Query Complexity in Quantum Algorithms, Lower Bounds for Quantum Algorithms, Speedups over Classical Algorithms.
Implementations and Simulations: Quantum Programming with Qiskit, Cirq, or other Quantum SDKs, Simulating Quantum Algorithms, Running Quantum Algorithms on Real Quantum Devices (IBM Q, Rigetti, etc.)

References:

1. Quantum Computing - J Gruska; McGraw Hill.
2. Quantum Computation and Quantum Information - Michael Nielsen and Isaac Chuang; Cambridge University Press.
3. Quantum Computing for Computer Scientists - Noson Yanofsky and Mirco Mannucci; Cambridge University Press.
4. An Introduction to Quantum Computing Paperback – Michele Mosca, Phillip Kaye, Raymond Laflamme; Oxford University Press.

Table 7: Descriptors for qualifications at levels 6.0 on the NHEQF

Element of the descriptor	NHEQF level descriptors
Knowledge and understanding	KU1: Advanced knowledge about a specialized field of enquiry, with depth in one or more fields of learning within a broad multidisciplinary/ interdisciplinary context. KU2: A coherent understanding of the established methods and techniques of research and enquiry applicable to the chosen fields of learning. KU3: An awareness and knowledge of the emerging developments and issues in the chosen fields of learning. KU4: Procedural knowledge required for performing and accomplishing professional tasks associated with the chosen fields of learning.
General, technical and professional skills required to perform and accomplish tasks	GT1: A range of cognitive and technical skills required for performing and accomplishing complex tasks relating to the chosen fields of learning. GT2: Cognitive and technical skills relating to the established research methods and techniques. GT3: Cognitive and technical skills required to evaluate complex ideas and undertake research and investigations to generate solutions to real-life problems. GT4: Generate solutions to complex problems independently, requiring the exercise of full personal judgement, responsibility, and accountability for the output of the initiatives taken as a practitioner.
Application of knowledge and skills	AK1: Apply the acquired advanced technical and/or theoretical knowledge and a range of cognitive and practical skills to analyze the quantitative and qualitative data gathered drawing on a wide range of sources for identifying problems and issues relating to the chosen fields of learning. AK2: Apply advanced knowledge relating to research methods to carry out research and investigations to formulate evidence-based solutions to complex and unpredictable problems.
Generic learning outcomes	GL1: Listen carefully, read texts and research papers analytically, and present complex information in a clear and concise manner to different groups/ audiences, GL2: Communicate technical information and explanations, and the findings/ results of the research studies relating to specialized fields of learning, GL3: Present in a concise manner one's views on the relevance and applications of the findings of research and evaluation studies in the context of emerging developments and issues. GL4: Meet own learning needs relating to the chosen fields of learning, GL5: Pursue self-paced and self-directed learning to upgrade knowledge and skills that will help accomplish complex tasks and pursue a higher level of education and research. GL6: A keen sense of observation, enquiry, and capability for asking relevant/ appropriate questions. GL7: The ability to problematize, synthesize and articulate issues and design research proposals. GL8: The ability to define problems, formulate appropriate and relevant research questions, formulate hypotheses, test hypotheses using quantitative and qualitative data, establish hypotheses, make inferences based on the analysis and interpretation of data, and predict cause-and-effect relationships. GL9: The capacity to develop appropriate tools for data collection. GL10: The ability to plan, execute and report the results of an experiment or investigation. GL11: The ability to acquire the understanding of basic research ethics and skills in practicing/doing ethics in the field/ in own research work, regardless of the funding authority or field of study. GL12: Examine and assess the implications and consequences of emerging developments and issues relating to the chosen fields of study based on empirical evidence. GL13: Make judgement in a range of situations by critically reviewing and consolidating evidence. GL14: Exercise judgement based on evaluation of evidence from a range of sources to generate solutions to complex problems, including real-life problems, associated with the chosen field(s) of learning requiring the exercise of full personal responsibility and accountability for the initiatives undertaken and the outputs/outcomes of own work as well as of the group as a team member.
Constitutional, humanistic, ethical, and moral values	CH1: Embrace and practice constitutional, humanistic, ethical, and moral values in life. CH2: Adopt objective, unbiased, and truthful actions in all aspects of work related to the chosen field(s) of learning and professional practice CH3: Present coherent arguments in support of relevant ethical and moral issues. CH4: Participate in actions to address environmental and sustainable development issues. CH5: Follow ethical practices in all aspects of research and development, including avoiding unethical practices such as fabrication, falsification, or misrepresentation of data or committing plagiarism.
Employability and job-ready skills, and entrepreneurship skills and capabilities or qualities and mindset	EJ1: Adapting to the future of work and to the demands of the fast pace of technological developments and innovations that drive a shift in employers' demands for skills, particularly with respect to the transition towards more technology-assisted work involving the creation of new forms of work and rapidly changing work and production processes. EJ2: Managing complex technical or professional activities or projects, requiring the exercise of full personal responsibility for the output of own work as well as for the outputs of the group as a member of the group/team. EJ3: Exercising supervision in the context of work having unpredictable changes.

Table 8: Level 6 (Year 4)

		SEMESTER-VII							SEMESTER-VIII					
Elements of the Descriptor		MJCS71	MJCS72	MJCS73	MNCS04	MJCS61	CSR71	CSEC7X	MJCS81	MJCS82	MNCS04	CSR81	CSEC8AY	CSEC8BZ
Knowledge and understanding	KU1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	KU2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	KU3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	KU4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
General, technical and professional skills required to perform and accomplish tasks	GT1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GT2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GT3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GT4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Application of knowledge and skills	AK1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	AK2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Generic learning outcomes	GL1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL2						✓					✓		
	GL3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL6	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL7	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL8	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL9	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL12	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL13	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	GL14	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Constitutional, humanistic, ethical, and moral values	CH1													
	CH2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	CH3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	CH4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	CH5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Employability and job-ready skills, and entrepreneurship skills and capabilities/qualities and mindset	EJ1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	EJ2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	EJ3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓