M.D. UNIVERSITY, ROHTAK

SCHEME OF STUDIES AND EXAMINATION
(Under NEP-2020)

Bachelor of Technology 2nd year
Common for
B. Tech. (Computer Science and Engineering)
B. Tech. (Computer Science and Technology)
B.Tech. (Computer Science & Information Technology)
B. Tech. (Computer Science and Design)
B.Tech. (Computer Engineering)

'H' Scheme (Under NEP-2020)

Effective from:

Academic Session 2026-27 (for 2nd Year)

1. Definition of Credit:

1 Hr. Lecture (L) per week	1 credit
1 Hr. Tutorial (T) per week	1 credit
1 Hr. Practical (P) per week	0.5 credits
2 Hours Practical (Lab) per week	1 credit

2. Structure of the Undergraduate Engineering program:

Sr. No.	Category
1	Humanities and Social Sciences including Management courses
2	Basic Science Courses
3	Engineering Science courses
4	Program Core Courses (Branch specific)
5	Professional Elective courses relevant to chosen specialization/branch
6	Open Elective Courses (from Humanities, Technical Emerging or other Subjects)
7	Project work, Seminar and Internship in Industry or elsewhere
8	Mandatory Courses [Environmental Sciences, Induction Program, Indian Constitution, Essence of Indian Knowledge Tradition] etc. (non-credit)

3. Course code and definition:

Course Code	**Definitions**
L	Lecture
T	Tutorial
P	Practical
BSC	Basic Science Courses
ESC	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management courses
PCC	Professional Core Courses
LC	Laboratory Courses
MC/AU	Mandatory Courses/ Audit Course
PT	Practical Training
PEC	Professional Elective courses
OEC	Open Elective courses
PRROJ/SEM	Seminar/ Employment Enhancement Courses (Project/Summer Internship/ Seminar)

PROGRAM OUTCOMES (PO)

Engineering Graduates will be able to:

PO1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

Semester-3

Sr. No.	Course Code	Course Title	Hours per week		Total Contact Hrs. per week	Credit	Examination Scheme (Marks)				Duration of Exam (Hours)
			L	P			Internal awards	Theory	Practical	Total	
1	26PCC-CSE-201H	Data Structures and Algorithms	3	0	3	**3**	25	50		75	3
2	26PCC-CSE-203H	Python Programming	3	0	3	**3**	25	50		75	3
3	26PCC-CSE-205H	Operating Systems	3	0	3	**3**	25	50		75	3
4	26PCC-CSE-207H	Software Engineering	3	0	3	**3**	25	50		75	3
5	26PCC-ECE-206H	Digital Electronics	3	0	3	**3**	25	50		75	3
6	26BSC-MATH-201 H	Mathematics-III(Multivariable Calculus and Differential Equations)	3	0	3	**3**	25	50		75	3
7	26LC-CSE-211H	Data Structures and Algorithms Using C Lab	0	2	2	**1**	5		20	25	3
8	26LC-CSE-213H	Python Programming Lab	0	2	2	**1**	5		20	25	3
9	26LC-CSE-215H	Operating Systems Lab	0	2	2	**1**	5		20	25	3
10	26LC-ECE-214H	Digital Electronics Lab	0	2	2	**1**	5		20	25	3
		Total			26	22				550	
For Honors course											
1	26PCC-CSE-209H	Introduction to MATLAB	3	0	3	**3**	25	50		75	3
2	26LC-CSE-217H	MATLAB	0	2	2	**1**	5		20	25	3
		Total			**5**	4				100	

Semester-4

Sr. No.	Course Code	Course Title	Hours per week			Total Contact Hrs. per week	Credit	Examination Scheme (Marks)				Duration of Exam (Hours)
			L	T	P			Internal awards	Theory	Practical	Total	
1	26PCC-CSE-202H	Discrete Structures	3	1	0	4	**4**	30	70		100	3
2	26PCC-AIML-221	Artificial Intelligence	3	0	0	3	**3**	25	50		75	3
3	26PCC-CSE-206H	Database Management Systems	3	0	0	3	**3**	25	50		75	3
4	26PCC-CSE-208H	Object-Oriented Programming	3	0	0	3	**3**	25	50		75	3
5	26PCC-CSE-210H	Computer Networks	3	0	0	3	**3**	25	50		75	3
6	Professional Elective courses-I	Refer to Annexure -I	3	0	0	3	**3**	25	50		75	3
7	26LC-CSE-212H	Database Management Systems Lab	0	0	2	2	**1**	5		20	25	3
8	26LC-CSE-214H	Object-Oriented Programming Lab	0	0	2	2	**1**	5		20	25	3
9	26LC-CSE-216H	Computer Networks Lab	0	0	2	2	**1**	5		20	25	3
10	26LC-AIML-215H	Artificial Intelligence Lab	0	0	2	2	**1**	5		20	25	3
11	26-PROJ-CSE-218H	Professional Competency Lab	0	0	2	2	1	5		20	25	3
12	26-AU-ENV-202-H	Environmental Science	2	0	1	3	**0**	Refer Note 1				
		Total				32	24				600	
	For Honors course											
1	26PCC-CSE-212H	Web Technologies	3		0	3	**3**	25	50		75	3
2	26LC-CSE-218H	Web Technologies Lab	0		2	2	**1**	5		20	25	3
		Total				**5**	4				100	

Note-1: AU**/**MC is a mandatory non–credit course in which students will be required to obtain passing marks in theory. The marks in this qualifying paper will not be included in determining the percentage of marks obtained for the award of degree. However, these marks will be shown in detailed marks certificate of student.

Marks Distribution:

- Internal Assessment: 30 marks
- Theory Examination: 70 marks

The institute will conduct the theory examination at its own level.

Note-2: At the end of the 4th semester, each student has to undergo Practical Training of 4/6 weeks in an Industry/ Institute/ Professional organization/ Research Laboratory/ training centre, etc., and submit a typed report along with a certificate from the organization & its evaluation shall be carried out in the 5th Semester.

Annexure -I

Professional Elective courses-I

1. 26PEC-CSE-202H : Computer Graphics
2. 26PEC-CSE-204H : Cloud Computing
3. 26PEC-CSE-206H: Optimization Techniques

Data Structures & Algorithms

Course code	26PCC-CSE-201H				
Category	Program Core Course				
Course title	Data Structures & Algorithms				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester 3
	3	**0**	**0**	**3**	
Class work	25 Marks				
Exam	50 Marks				
Total	75Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To impart the basic concepts of data structures and algorithms.
2. To understand concepts about searching and sorting techniques
3. To understand basic concepts about stacks, queues, lists, trees and graphs.
4. To enable them to write algorithms for solving problems with the help of fundamental data structures.

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Define various types of data structures and the operations that can be implemented on these data structures. Understand the importance of data structures and algorithms in the context of writing efficient programs. (Level 1: Remember and Understand)
CO2: Choose the appropriate data Structure and algorithm for the specific problems. (Apply)
CO3: Compare and evaluate the performance of various data structures and algorithms. (Level 4: Analyze and Evaluate).
CO4: Develop skills to design data structures in problem-solving.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-		-	-	-	-	-	-
CO2	3	2		-	2		-	-	-	-	-
CO3	3	3					-	-	-		
CO4	3				2	-					

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

Unit I

Introduction: Basic Terminologies: Concept of Data Structures, Choice of right Data Structure. Algorithms, how to design and develop algorithm, Complexity of algorithm. Operations: insertion,

deletion, traversal etc.; Analysis of an Algorithm, Searching: Linear Search and Binary Search and their complexity analysis.

Unit II

Stacks and Queues: Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and Evaluation -corresponding algorithms and complexity analysis. Queue and its operations: Algorithms and their complexity analysis, Types of Queues: Simple Queues, Circular Queues, and Priority Queues.

Unit III

Linked Lists: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion, Deletion; Linked representation of Stack and Queue; Header linked list, Doubly linked list, Circular Linked Lists: operations on it and algorithmic analysis and the complexity analysis.

Trees: Basic Tree Terminologies, Different types of Trees: Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. M-way Trees, B Trees, B+ Trees: definitions, algorithms and analysis.

Unit IV

Sorting and Hashing: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Selection Sort Quick Sort, Merge Sort, Heap Sort; Analysis and Comparison of different sorting algorithms.

Graph: Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis. Minimum Spanning Tree: Kruskal's Algorithm, Prim's Algorithm.

Alternative NPTEL/SWAYAM Course:

SNo	SWAYAM course name	Instructor	Host Institute
1.	Data structures and Algorithms Design	Prof. Nitin Saxena	IIT Kanpur

Suggested books

1. "Data Structures with C", by Lipschutz Seymour, McGraw Hill Education India.
2. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.

Suggested reference books

1. "Data Structures with C", by A.K Sharma, Pearson.

Python Programming

Course code	26PCC-CSE-203H				
Category	Program Core Course				
Course title	Python Programming				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3
	3	**0**	**0**	**3**	
Classwork	25Marks				
Examination	50Marks				
Total	75Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To impart the basic concepts of Python programming.
2. To understand syntax of Python language
3. To create dynamic applications in Python language.
4. To implement objectoriented concepts using Python language

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Definebasic fundamental concepts of python programming**,** explain complex data structures like lists and dictionaries, concepts of turtle graphics, image transformations, Graphical User Interface, and Object-oriented programming fundamental concepts

CO2: Understand the concepts of data types, control statements, string operations, file handling, data structures, and functions

CO3: Applythe knowledge & ideas of basic syntax of python to build complex programs in python that involve making drawings using turtle graphics, doing image transformations, and making simple graphical user interface programs

CO4: Analyze theprogramming problem and make programs using various modules such as turtle graphics, image and GUI and design dynamic applications using OOPS concepts

CO - PO mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**
CO1	3	-	-	-	3	-	-	-	3	2	2
CO2	3	3	3	1	2	-	-	-	-	1	2
CO3	2	2	3	3	-	-	-	-	-	-	2
CO4	3	2	3	2	1	-	3	-	-	2	2

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

UNIT-I

Introduction: Installing Python; basic syntax, interactive shell, editing, saving, and running a script; data types; variables, assignments; numerical types; arithmetic operators and expressions; Loops and selection statements, Control statements String manipulations: subscript operator, indexing, slicing a string; text files: reading/writing text and numbers from/to a file; creating and reading a formatted file

UNIT-II

Lists, dictionary and Design with functions: Basic list operators, replacing, inserting, removing an element; searching and sorting lists; dictionary literals, adding, and removing keys, accessing and replacing values; traversing dictionaries. Hiding redundancy, complexity; arguments and return values; Program structure and design. Recursive functions.

UNIT-III

Simple graphics and image processing: Simple graphics, Turtle operations, Manipulating turtle screen, Drawingtwo dimensional shapes, examining an object attributes, Taking a random walk, Color and RGB scheme, Image processing: Image manipulation operations, properties of images, image module, copying, blurring and reducing image. Graphical User Interfaces: Terminal based and GUI based programs, Simple GUI-Based Programs, Windows and Window Components, Input and Output with Entry Fields, Defining and Using Instance Variables

UNIT-IV

Object Oriented concepts: Classes and OOP: classes, objects, attributes and methods; defining classes; design with classes, data modelling; Inheritance, polymorphism, operator overloading; abstract classes; exception handling, try block. Multithreading, Networks, And Client/Serverprogramming

Alternative NPTEL/SWAYAM Course:

SNo	NEPTEL course name	Instructor	Host Institute
1.	**Programming, Data Structures And Algorithms Using Python**	Prof. Madhavan Mukund	**Chennai Mathematical Institute**

Suggested Text Books:

1. “Fundamentals of Python: First Programs” Kenneth Lambert, Course Technology, Cengage Learning, 2012
2. "Programming with Python," T.R. Padmanabhan, 1st Ed., 2016.

Suggested Reference Books:

1.Python for Everyone: Learn and Polish Your Coding skills in Python by Saurabh Chandrakar (BPB Publications).

Operating Systems

Course code	26PCC-CSE-205H				
Category	Program Core Course				
Course title	Operating Systems				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3/4
	3	**0**	**0**	**3**	
Branches (B. Tech.)	Computer Science and Engineering				
Classwork	25 Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To understand the role and functions of an operating system and its impact on the overall performance of a computer system.
2. To understand the concepts and techniques involved in process management, such as process creation, scheduling, inter-process communication, and synchronization.
3. To understand the concepts and techniques involved in memory management, such as virtual memory, swapping, paging, and segmentation.
4. To gain hands-on experience with the design and implementation of operating systems through programming projects and case studies.

COURSE OUTCOMES (CO):

After successful completion of the course, the students will be able to:

CO1: Define various fundamental concepts like processes, threads, functions, and design characteristics of various types of operating systems.
CO2: Compare and contrast various process synchronization and Critical Section problems and discuss the solutions to these problems using Semaphores
CO3: Apply operating system design concepts for solving problems regarding scheduling, memory management, disk scheduling, deadlocks, etc.
CO4: Analyze and evaluate the performance of various operating systems Algorithms related to CPU Scheduling, Page Replacement Algorithms, File Allocation, etc.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO 11**
CO1	3	-	-	-	-	-	-	-	-	-	-
CO2	2	2		1	-	2	1	-	-	-	-
CO3	3	2			2	-	-	3	2		2
CO4	2	1	2	2		-				2	

UNIT I

Introduction: Concept of Operating Systems, Generations of Operating Systems, Types of Operating Systems, OS Functions and Objectives.

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Light Weight and Heavy Weight Processes.

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers- Long Term and Short Term Schedulers, Dispatcher, Context Switching time, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Average Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS (First Come First Serve), SJF (Shortest Job First) , SRTF (Shortest Remaining Time First), RR (Round Robin)/Time Sharing Scheduling Algorithms.

UNIT II

Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Semaphores-Binary and Counting Semaphores, Busy-waiting (Spinning) and Wait and Signal operations for Synchronization, Classical IPC Problems: Reader's & Writer Problem, Producer-Consumer Problem, Dinning Philosopher Problem etc.

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, and Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

UNIT III

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory Management, Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Optimal Page Replacement and Least Recently used (LRU).

UNIT IV

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Case study on UNIX and WINDOWS Operating Systems.

Alternative NPTEL/SWAYAM Course:

	Course Name	Instructor	Host Institute
1	Introduction to Operating Systems	Prof. CHESTER REBEIRO	IIT, Madras

Suggested books

1. Operating System Concepts Essentials, 9th Edition by Avi Silberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.

2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice-Hall of India.

Suggested reference books

1. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
2. Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley
3. Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India
4. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

Software Engineering

Course code	26PCC-CSE-207H				
Category	Program Core Course				
Course title	Operating Systems				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3/4/5
	3	**0**	**0**	**3**	
Branches (B. Tech.)	Computer Science and Engineering				
Classwork	25 Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. Ability to translate end-user requirements into system and software requirements, using e.g. UML, and structure the user requirements in a Software Requirements Specification Document (SRS).
2. Identify and apply appropriate software architectures and patterns to carry out high-level design of a system and be able to make critical comparisons of various choices.
3. To gain exposure/or awareness of testing problems and to be able to develop a simple testing report.
4. Application of quality management techniques to small size software projects.

COURSE OUTCOMES:

After successful completion of the course, the students will be able to:

CO1: Understand and explain fundamental concepts, principles, and processes of Software Engineering and Software Development Life Cycle (SDLC).
CO2: Apply appropriate software process models (Agile, Waterfall, Spiral, etc.) for developing quality software systems.
CO3: Analyze user requirements and prepare Software Requirement Specification (SRS) documents using standard practices.
CO4: Design software systems using structured and object-oriented design approaches, including UML diagrams. Implement testing strategies and quality assurance techniques to ensure reliable and maintainable software.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO 11**
CO1	2	-	-	2	-	2	-	2	-	2	-
CO2		2	1	-	-	-	-	-	-	-	-
CO3	1			2	2	-	2	-	2		2
CO4	3	2	3			1		1		1	

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

UNIT I

Introduction to Software Engineering: Basic Concepts & characteristics of Software, Types of software, Phases of Software development life cycle - feasibility study, planning, requirement analysis, designing, coding, testing & debugging, implementation & maintenance.

SDLC models: Waterfall, Prototype, Incremental Process, Spiral, Agile Model. Basic concepts of Software Engineering, David Hooker's Seven Principles Model of Software Engineering, 4 P's of Software Project Management, introduction to Software metrics, and risk management.

UNIT II

Requirement Engineering: Types of Software Requirements – functional and non-functional, Requirement analysis, Requirement elicitation techniques, documenting User Requirements- SRS, Static verification of user requirements, Requirement Management & Modelling.

Software Designing: Different levels of software design: Architectural Design, High level design (HLD), Detailed Design/ Low level Design (LLD), Coupling and Cohesion concepts. Software design approaches-Top down, Bottom up approach, Software designing tools.

UNIT III

Coding & Testing: Programming languages and coding styles, Software Testing Life Cycle-Test Case Planning, Test Case Designing, Test Environment Setup-Stubs and Drivers, Test Case Execution, Test Closure and test adequacy criteria.

Testing Strategies: Static and Dynamic Testing,Levels of Software Testing: Unit, Integration and System Testing. Types of Software Testing: Functional and Non-functional Testing. V Model for Software testing, alpha and beta testing; System testing: Recovery testing, security testing, stress testing, Regression Testing and its Types, Introduction to Software Bugs, Debugging approaches.

UNIT IV

Software Maintenance & Quality Assurance:Corrective, adaptive and Perfective maintenance. Software quality attributes and models- Mc Call's Model, Boehm's Model. CMM-Capability maturity model and Six Sigma Model for quality management, ISO 9001, Reverse Engineering & Reengineering, Reusability of software components.

Alternative NPTEL/SWAYAM Course:

	Course Name	**Instructor**	**Host Institute**
1	Introduction to Software Engg.	Prof. N.L SARDA, Prof. R. K. Joshi	IIT, Mumbai

Suggested books

1. Software Engineering – A Practitioner's Approach, Roger S. Pressman, 1996, MGH.

2. Software Engineering- Sommerville, 7th edition, Pearson Education.

Suggested reference books

1. Fundamentals of software Engineering, Rajib Mall, PHI Software Engineering by Nasib Singh Gill, Khanna Book Publishing Co (p) Ltd
2. Software Engineering by Ian Somerville, Pearson Edu, 5 edition, 1999, AW,
3. Software Engineering – David Gustafson, 2002, T.M.H
4. Software Engineering Fundamentals Oxford University, Ali Behforooz and Frederick J. Hudson 1995 JW&S,
5. An Integrated Approach to software engineering by Pankaj jalote , 1991 Narosa,

Digital Electronics

Course code	26PCC-ECE-206H				
Category	Program Core Course				
Course title	**Digital Electronics**				
Scheme and Credits	**L**	**T**	**P**	**Credits**	**Semester-III/IV**
	3	0	0	3	
Classwork	25 Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

Course Objectives:

CO1: To introduce fundamental concepts of digital systems, number systems, codes, and logic representations.

CO2: To develop skills for designing and analyzing combinational logic circuits, including simplification and synthesis techniques.

CO3: To understand various logic families, their characteristics, and practical implications in digital design.

CO4: To impart knowledge of sequential circuits, state machine design, FPGAs and ASICs.

COURSE OUTCOMES (CO):

After completing this course, the student will be able to:

CO1	Understand number systems, number representation, binary codes and error detection methods.
CO2	Analyze and simplify Boolean expressions using K-Map, Quine–McCluskey and design various combinational logic circuits.
CO3	Explain properties of different logic families such as TTL, CMOS, ECL and their performance metrics.
CO4	Design, analyze and implement sequential circuits using latches, flip-flops ,state machines, Memories ,FPGAs and ASICs

CO - PO mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11
CO1	3	2	–	–	–	–	–	–	–	–	2
CO2	3	3	2	–	–	–	–	–	–	–	2
CO3	2	3	3	–	2	–	–	–	–	–	2
CO4	2	2	–	–	3	–	–	–	–	–	2

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

UNIT I

Introduction to Digital Techniques: Digital Systems; Logic circuits, Analysis, design and implementation of digital systems, Number Systems and Codes-; Binary, octal and hexadecimal number systems; Methods of base conversions; Binary, octal and hexadecimal arithmetic; Representation of signed numbers; Fixed and floating point numbers; Binary codes: BCD codes, Excess-3, Gray codes; Error detection and correction codes -parity check codes and Hamming code, Canonical and standard forms, POS and SOP simplifications, K-Map method, Q-M method, don't care conditions

UNIT II

Logic families: Introduction to different logic families; such as RTL, TTL, ECL, and CMOS with their characteristics including propagation delay, power dissipation, fan-in, fan-out, and noise margin.
Combinational design using MST devices: Encoders, Decoders, multiplexers, Demultiplexer and their use as logic elements; Parity circuits and comparators; Arithmetic modules- adders, subtractors , BCD arithmetic circuits

UNIT III

Sequential circuits: Definition of state machines, state machine as a sequential controller; Basic sequential circuits- latches and flip-flops: SR-latch, D-latch, D flip-flop, JK flip-flop, T flip-flop; Timing hazards and races ;Analysis of state machines using D flip-flops and JK flip-flops; Design of state machines - state table, state assignment, transition/excitation table,

Unit IV

State machine design: Designing state machine using ASM charts, Designing state machine using state diagram, Design of registers, counters-asynchronous and synchronous, up/down counter, Ring and Johnson counters.

Memory – Organization, Functional Diagram, Memory operations, Classification of semiconductor memories, Read and Write Memories, ROM ,Programmable Logic Devices-PLAs, PALs and their applications, Generic Array logic devices, Sequential PLDs and their applications; Introduction to field programmable gate arrays (FPGAs) and ASICS

Alternative NPTEL/SWAYAM Course:

S. No.	NPTEL Course Name	Instructor	Host Institute
1	Digital Circuits	**Prof.Santanu Chattopadhyay**	IIT Kharagpur **NPTEL**
2	Digital Electronics	**Dr. P. Sivasankar**	**NITTTR Chennai**, SWAYAM

TextBooks:

1. G.K .Kharate: Digital Electronics,1st edition,Oxforduniversity press,2010

ReferenceBooks:

1. M .M. ManoandM.D.Ciletti:Digital design 4th edition, PrinteceHall.2006
2. R.P .Jain:Modern Digital Electronics, 3rdedition,TMH.2003
3..A.A.Kumar:Fundamentals of digital circuits,2ndedition,Printece Hall of India
4. A.P.MalvinoandD.P.Leach:Digital principle and applications, 6th edition,TMH,2008
5. Z.Kohavi ,Switching and FiniteAutomataTheory,Mc GrawHill,1970.

MATHEMATICS-III (Multivariable Calculus and Differential Equation)

Course code	26BSC-MATH-201 H				
Category	Basic Science Course				
Course title	**Mathematics-III (Multivariable Calculus and Differential Equation)**				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3
	3	**0**	**0**	**3**	
Classwork	25 Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To develop a clear understanding of nature of infinite series, functions of several variables, including limits, continuity, partial derivatives, and multiple integrals.
2. To equip students with analytical techniques to determine local maxima and minima of functions of two variables using standard methods.
3. To enable students to apply double and triple integrals for computing areas and volumes relevant to physical and engineering problems.
4. To build proficiency in solving ordinary differential equations of first, second, and higher order arising in mathematical modeling and engineering applications.

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Define the concepts and terminology of multivariable calculus and differential equations including convergence of series, partial derivatives, multiple integrals, exact differential equations, linear and Bernoulli's equations, linear differential equations of second and higher order, gradient, divergence and curl.

CO2: Understand the significance and contribution of various concepts/ theorems/methods such as tests for series convergence, Euler's Theorem, Lagrange's method of undetermined multipliers, multiple integrals, change of order, change of variable, exactness of differential equations, method of variation of parameters, Green, Stokes and Gauss theoremsetc.

CO3:Apply the notion of multivariable calculus in finding maxima and minima of functions.

CO4:Solve ordinary differential equations of first, second and higher order

CO - PO mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**
CO1	3	3	1	-	2	-	-	-	-	-	-
CO2	3	3	2	2	2	1	-	-	2	1	-
CO3	3	3	2	1	2	1	-	-	2	1	-
CO4	3	3	1	3	2	1	-	-	2	1	-

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

Unit-I

Sequences and Series: Convergence of sequence and series, Tests for convergence: Comparison test, Cauchy's Root test, D' Alembert's Ratio test, Raabe's test, Logarithmic test, Gauss test, Cauchy's Integral test, Alternating series, Absolute and conditionally convergent series, Fourier series, Half range sine and cosine series

Unit-II

Multivariable Differential Calculus: Functions of several variables: Limit and Continuity, Partial derivatives, Homogeneous functions, Euler's Theorem, Total derivative, Local maxima and minima for function of two variables, Lagrange's method of undetermined multipliers, Gradient of a scalar field and Directional derivative, Divergence and Curl of a vector field

Unit-III

Multivariable Integral Calculus: Double integral, Change of order of integration, Triple integral, Change of variable (Cartesian to polar coordinate) for double and (Cartesian to Spherical and Cylindrical polar coordinates) for triple integrals, Line integral, Surface integral, Volume integral, Green, Stokes and Gauss theorems (without proof) and their applications.

Unit-IV

Differential Equations: First order linear equations, Bernoulli's equations, Exact differential equations, Equations reducible to exact differential equations, Linear differential equations of second and higher order, Complete solution, Complementary function and Particular integral, Method of variation of parameters to find particular integral, Cauchy's and Legendre's linear equations

Suggested Reference/Text Books

1. B. S. Grewal, Higher Engineering Mathematics, Khanna Publishers.
2. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, Pearson Education.
3. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons.
4. B. V. Ramana, Higher Engineering Mathematics, Tata McGraw-Hill Publishing Company Limited.
5. N. P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications.
6. W. E. Boyce and R. C. Di Prima, Elementary Differential Equations and Boundary Value Problems, Wiley India.
7. S. L. Ross, Differential Equations, Wiley India.
8. E. A. Coddington, An Introduction to Ordinary Differential Equations, Prentice Hall India.

Data Structures and Algorithms Using C Lab

Course code	26LC-CSE-211H				
Category	Laboratory Course				
Course title	Data Structures & Algorithms Using C Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester 3
	0	**0**	**2**	**1**	
Class work	05 Marks				
Exam	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

Course Objectives:

1. To develop practical skills in implementing fundamental data structures such as arrays, linked lists, stacks, queues, trees, and graphs using the C programming language.
2. To enable students to analyze and apply appropriate data structures for solving real-world computational problems efficiently.
3. To provide hands-on experience in designing and testing algorithms for operations like searching, sorting, insertion, deletion, and traversal.
4. 4.To enhance problem-solving and programming proficiency by integrating theoretical concepts of data structures with practical C-based implementations.

Course Outcomes:

By the end of the Course, Students will be able to:

CO1: Implement various Data Structures and related Operations (Level3: Apply)

CO2: Compare solution on the basis of appropriateness of data structures. (Level 5: Evaluate)

CO3: Create lab record for assignments that includes problem definitions, design of solutions. (Level 6: Create)

CO4: Demonstrate use of ethical practices, self-learning, and team spirit. (Level 3: Apply)

CO-PO Mapping:

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2			2						
CO2	3	3									
CO3	3	2			2						
CO4								3	2		

List of Experiments:

1. Write a menu driven program that implements following operations (using separate functions) on a linear array:
    1. Insert a new element at end as well as at a given position
    2. Delete an element from a given whose value is given or whose position is given
    3. To find the location of a given element
    4. To display the elements of the linear array

2. Write a menu driven program that maintains a linear linked list whose elements are stored in on ascending order and implements the following operations (using separate functions):

- Insert a new element
- Delete an existing element
- Search an element
- Display all the elements

3. Write a program to demonstrate the use of stack (implemented using linear array) in converting arithmetic expression from infix notation to postfix notation.

4. Program to demonstrate the use of stack (implemented using linear linked lists) in evaluating arithmetic expression in postfix notation.

5. Program to demonstration the implementation of various operations on a linear queue represented using a linear array.

6. Program to demonstration the implementation of various operations on a circular queue represented using a linear array.

7. Program to demonstration the implementation of various operations on a queue represented using a linear linked list (linked queue).

8. Program to illustrate the implementation of different operations on a binary search tree.

9. Program to illustrate the traversal of graph using breadth-first search

10. Program to illustrate the traversal of graph using depth-first search.

11. Program to sort an array of integers in ascending order using bubble sort.

12. Program to sort an array of integers in ascending order using selection sort.

13. Program to sort an array of integers in ascending order using insertion sort.

14. Program to sort an array of integers in ascending order using merge sort.

15. Program to sort an array of integers in ascending order using quick sort.

16. Program to sort an array of integers in ascending order using heap sort.

17. Program to sort an array of integers in ascending order using shell sort.

18. Program to demonstrate the use of linear search to search a given element in an array.

19. Program to demonstrate the use of binary search to search a given element in a sorted array in ascending order.

Suggested books

1. "Data Structures with C", by Lipschutz Seymour, McGraw Hill Education India.

Virtual Lab Resources

Sr. No.	Name of the Virtual Lab	Organizing Institute	Link
1.	Data Structures -I	IIIT Hyderabad	https://ds1-iiith.vlabs.ac.in/
2.	Data Structures -II	IIIT Hyderabad	https://ds2-iiith.vlabs.ac.in/
3.	Data Structures	IIIT Hyderabad	https://cse01-iiith.vlabs.ac.in/

Python Programming Lab

Course code	26LC-CSE-213H				
Category	Laboratory Course (LC)				
Course title	Python Programming Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3
	0	**0**	**2**	**1**	
Internal Lab	05 Marks				
External lab	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

Course Objectives:

1. **To develop foundational programming skills using Python**, including variables, data types, control structures, and functions.
2. **To provide hands-on experience with core Python data structures** such as lists, tuples, sets, and dictionaries for efficient data handling.
3. **To enable students to design, implement, and debug Python programs** for solving computational and real-world problems.
4. **To introduce practical applications of Python**, including file handling, basic object-oriented programming, and use of standard libraries.

Course Outcomes:

The students will be able to:

CO1: Write, test, and debug simple Python programs involving loops, files, strings and texts.. (Level 1: Understand)

CO2: Apply basic python programming to handle Python lists, tuples, dictionaries for representing compound data (Level 3: Apply)

CO3: Analyze the turtle module and GUI modules (Level 4: Analyze)

CO4: Create basic object oriented program andcollaborate on group projectsusing ethical practices, self-learning, and team spirit. (Level 1: Understand, Level 6: Create)

CO-PO Mapping:

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2							2	2
CO2										2	2
CO3	3							2	2	2	2
CO4	3							2	2	2	3

LIST OF EXPERIMENTS:-

1. Write a program that calculates income tax.
2. Write a program that computes an investment report.

3. Write a program that computes square roots.
4. Write a program that computes the Flesch Index and grade level for text stored in a text file.
5. Write a program that generates sentences.
6. Write a program that emulates a nondirective psychotherapist.
7. Write a program that allows the user to obtain information about the file system.
8. Write a program that allows the user to draw a particular c-curve in varying degrees.
9. Write a program that allows the user to play and study the game of craps.
10. Write a program that simulates a simple ATM.
11. Replace the terminal-based interface of the ATM program with a GUI.
12. Write a program that supports an online chat room.

Suggested Books:

1. "Fundamentals of Python: First Programs" Kenneth Lambert, Course Technology, Cengage Learning, 2012

Note: Students must perform at least 10 experiments.

1). Each laboratory class/section shall not be more than about 20 students.

2). To allow fair opportunity of practical hands on experience to each student, each experiment may either done by each student individually or in group of not more than 3-4 students. Larger groups are strictly discouraged/disallowed.

3). Pre-experimental & post-experimental quiz /questions may be offered for each lab experiment to reinforce &aid comprehension of the experiment.

S. N	Experiment Name	Experiment Link(s)
1.	Arithmetic Operations Used in Python	https://python-iitk.vlabs.ac.in/exp/arithmetic-operations/
2.	Basics of Functions Used in Python	https://python-iitk.vlabs.ac.in/exp/built-in-functions/
3.	Understand the flow of controls and types of loops used in Python	https://python-iitk.vlabs.ac.in/exp/loops/
4.	Learn about the various array types in Python	https://python-iitk.vlabs.ac.in/exp/data-types/
5.	Learn the concepts of string in Python	https://python-iitk.vlabs.ac.in/exp/strings/
6.	Learn the concept of class and object	https://python-iitk.vlabs.ac.in/exp/classes-and-objects/
7.	In this experiment you will perform specific functions	**https://python-iitk.vlabs.ac.in/exp/built-in-modules/**
8.	Learn the concepts of Constructor and Inheritance in Python	**https://python-iitk.vlabs.ac.in/exp/constructors-and-inheritance/**

9.	In this experiment user will understand file operations	**https://python-iitk.vlabs.ac.in/exp/file-operators/**

EXPERIMENTS THAT MAY BE PERFORMED THROUGH VIRTUAL LABS:

Operating Systems Lab

Course code	26LC-CSE-215H				
Category	Laboratory Course (LC)				
Course title	Operating Systems Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3
	0	**0**	**2**	**1**	
Branches (B. Tech.)	Computer Science and Engineering				
Classwork	5 Marks				
Examination	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

Course Objectives:

1. Develop a solid understanding of the UNIX environment, including file systems, processes, and basic commands.
2. Gain proficiency in shell scripting to automate tasks and manage system operations efficiently.
3. Learn system-level programming using shell scripts based on looping, file handling and inter-process communication.
4. Build problem-solving skills by creating and debugging UNIX-based applications and utilities

COURSE OUTCOMES (CO):

Course Outcomes:

By the end of the Course, Students will be able to:

CO1: Apply commands related to vi editor, general utilities, and file systems.

CO2: write basic shell scripts and understand the structure and architectural components of the UNIX Operating System

CO3: Analyze the various different probable solutions/ logics for a given shell script and find the optimized solution

CO4: create lab record for assignments that includes problem definitions, design of solutions and conclusions and Demonstrate use of ethical practices, self-learning, and team spirit.

List of Experiments:

1 Introduction to Types of Shells and UNIX File System.

2. File and Directory Related Commands in UNIX.

3. Essential UNIX Commands for working in a UNIX environment.

4. I/O Redirection and Piping and Introduction to VI editor and Modes-Command Mode, Insert Mode etc.

5. Introduction of Processes and Communication in UNIX

6. Introduction of the concept of Shell Scripting.

7. Decision and Iterative Statements in Shell Scripting.

8. Writing the Shall Scripts for unknown problems. e.g. Fibonacci Series, Even/Odd number, Factorial of a number, Counting up to any number N, Arithmetic Operations

Suggested Books:

1. UNIX Shell Programming by YashavantKanetkar.
2. UNIX Concepts and Applications by Sumitabha Das
3. Introduction To Unix And Shell Programming by M.G. VenkateshMurthy, Pearson Publication.

Digital Electronics Lab

Course code	26LC-ECE-214H				
Category	Program Core Course				
Course title	**Digital Electronics Lab**				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-III/IV
	0	**0**	**2**	1	
Internal Lab	05 Marks				
External lab	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

Course Objective

This Laboratory will help the students to understand and design various logic gates, FF and Counters.

Course Outcomes

CO1	Demonstrate the working of basic TTL gates (AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR) and construct truth tables.	
CO2	Simplify Boolean expressions using Karnaugh Maps (K-Maps) and implement the corresponding logic circuits.	
CO3	Verify the functionality of combinational devices like Multiplexers, Demultiplexers, and 2-bit Comparators.	
CO4	Verify the operation and truth tables of flip-flops: S-R, J-K, T, D.	

CO - PO mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11
CO1	3	3	2	3	2	-	-	-	-	-	2
CO2	3	3	2	3	2	1	-	-	-	-	2
CO3	3	3	3	3	2	1	-	-	-	-	2
CO4	3	3	3	3	2	1	-	-	-	-	2

List of experiments:

Perform any 10 Experiments:-

1. Study of TTL gates AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR.
2. Design and realize given function using K-Maps and verify its performance.
3. To verify the operation of Multiplexer and Demultiplexer.
4. To verify the operation of 2bit Comparator using gates.
5. To verify the truth table of S-R,J-K,T, DFlip-flops.
6. To verify the operation of Bi-directional shift register.

7. To design and verify the operation of 3-bit asynchronous counter.
8. To design and verify the operation of 3-bit synchronous counter.
9. To design and verify the operation of asynchronous Up/down counter using J-KFFs.
10. To design and verify the operation of synchronous Up/down counter using J-KFFs.
11. To design and verify the operation of skipping state synchronous counter using JK&T FF
12. To design and verify the operation of 3-bit Ring counter .
13. To design and verify the operation of 3-bit Johnson's counter.
14. Design a state machine of 4 states.
15. To design a sequence detector.

EXPERIMENTS THAT MAY BE PERFORMED THROUGH VIRTUAL LABS:

S. N	Experiment Name	Experiment Link(s)
1.	On line verify the AND, OR, NOT, NAND, NOR, XOR, MUX, demux, comparators, flip flops, etc., and simulate behavior.	https://academo.org/demos/logic-gate-simulator/?utm_source=chatgpt.com
2.	**verify truth tables of RS, JK, T, and D flip-flops** using logic gate**)**	https://de-iitr.vlabs.ac.in/exp/truth-tables-flip-flops/simulation.html?utm_source=chatgpt.com
3.	Synchronous and asynchronous counter, Ring counter	https://deldsim.com/?utm_source=chatgpt.com
4.	State machine and sequence Detector	https://www.elab.iitb.ac.in

Introduction to MATLAB

Course code	26PCC-CSE-209H				
Category	Program Core Courses				
Course title	Introduction to MATLAB				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3
	3	**0**	**0**	**3**	
Class work	25 Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To introduce the MATLAB environment, basic syntax, and array/matrix operations for fundamental computational problem-solving.
2. To develop proficiency in writing MATLAB scripts, utilizing control flow statements, and creating user-defined functions.
3. To equip students with data visualization skills using advanced 2D and 3D plotting techniques to represent engineering and scientific data.
4. To apply MATLAB tools for mathematical modeling, statistical analysis, curve fitting

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Define the features of the MATLAB environment, basic syntax, matrix operations, and built-in functions.

CO2: Understand the concepts of script creation, control structures, iterative loops, and Simulink block diagrams.

CO3: Apply user-defined functions and plotting tools to visualize mathematical functions and represent real-world datasets.

CO4: Analyze engineering problems using advanced mathematical operations like polynomials, curve fitting, and numerical simulations.

CO - PO mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	-	3	-	-	-	-	-	-
CO2	3	3	2	1	3	-	-	-	-	1	-
CO3	3	3	3	2	3	-	-	-	1	1	-
CO4	3	3	3	3	3	-	-	-	1	2	-

UNIT-I

Introduction to MATLAB & Basic Mathematics: MATLAB Environment (Command Window, Workspace, Editor), Features and Advantages, Variables and Data Types, Basic Mathematical Operations, Built-in Math Functions.

UNIT-II

Vectors and Matrices: Row and Column Vectors, Matrix Creation, Matrix Operations (Addition, Multiplication, Transpose, Determinant, Inverse), Array Addressing and Indexing, Element-by-Element Operations.

Programming in MATLAB: M-Files (Scripts and Functions), Input/Output Commands (disp, fprintf, input).

UNIT-III

Control Flow and Decision Making: Relational and Logical Operators, if, if-else, if-elseif-else statements, switch-case. Iterative Loops: for loops, while loops, break and continue statements. User-Defined Functions, Anonymous Functions, Local and Global Variables.

UNIT-IV

Data Visualization and Plotting: 2D Plotting: The plot command, Formatting Plots (Title, Labels, Legends, Grid, Colors, Line Styles), Multiple Plots in One Figure (hold on, subplot), Scatter Plots, Bar Graphs, Histograms, Pie Charts. 3D Plotting: Line plots in 3D (plot3), Mesh and Surface Plots (mesh, surf), Contour Plots. Representing Polynomials, Roots of Polynomials, Polynomial Addition/Multiplication, Curve Fitting tools (polyfit, polyval).

Alternative NPTEL/SWAYAM Course:

SNo	NEPTEL course name	Instructor	Host Institute
1.	MATLAB Programming for Numerical Computation	Prof. NiketKaisare	IIT Madras
2	Applied Linear Algebra for Signal Processing, Data Analytics and Machine Learning (Uses MATLAB)	Prof. Aditya K. Jagannatham	IIT Kanpur

Suggested References Books:

1. MATLAB: A Practical Introduction to Programming and Problem Solving, 6th Edition by Stormy Attaway, Butterworth-Heinemann.
2. Getting Started with MATLAB: A Quick Introduction for Scientists and Engineers by Rudra Pratap, Oxford University Press.
3. MATLAB Programming for Engineers, 6th Edition by Stephen J. Chapman, Cengage Learning.

MATLAB

Course code	26LC-CSE-217-H				
Category	Laboratory Courses				
Course title	MATLAB				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-3
	0	**0**	**2**	**1**	
Internal Lab	05 Marks				
External lab	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

COURSE OUTCOMES:

The students will be able:

CO1: To gain hands-on familiarity with the MATLAB IDE, command window, and workspace for performing foundational matrix and array computations.

CO2:To develop practical programming skills by writing, debugging, and executing M-file scripts and user-defined functions using logical control structures and loops.

CO3: To programmatically visualize engineering data and mathematical models by creating, formatting, and manipulating complex 2D and 3D plots.

CO4: To apply advanced MATLAB toolboxes to perform polynomial curve fitting, solve numerical problems, and simulate basic dynamic systems.

CO - PO mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**
CO1	3	2	1	-	3	-	-	-	-	-	-
CO2	3	3	2	1	3	-	-	-	-	-	-
CO3	3	2	2	2	3	-	-	-	-	2	-
CO4	3	3	3	3	3	-	-	-	-	1	-

LIST OF EXPERIMENTS

1. Install/Access MATLAB, explore the Command Window, Workspace, and perform basic arithmetic operations using scalar variables.
2. Write a script to create row vectors, column vectors, and 2D matrices, and extract specific elements using indexing.
3. Perform standard matrix operations (Addition, Multiplication, Transpose, Inverse) and element-wise (dot) operations.
4. Solve a system of linear equations using matrix division (the \ operator).
5. Use built-in mathematical functions (trigonometric, exponential, logarithmic) to compute values for given arrays.
6. Write a script that takes user input (input), calculates the area and perimeter of various geometric shapes, and displays the result (fprintf).

7. Write a program using if-elseif-else to evaluate student grades based on inputted marks.
8. Create a switch-case script to act as a simple calculator (Add, Subtract, Multiply, Divide).
9. Use a for loop to generate the Fibonacci sequence and compute the factorial of a given number.
10. Create a User-Defined Function (.m file) that accepts parameters, performs a calculation, and returns multiple outputs.
11. Plot sine and cosine waves on the same graph using hold on, and add a title, labels, and legends.
12. Use the subplot command to display four different mathematical functions in a 2x2 grid in a single figure window.
13. Generate random data and visualize it using a Scatter Plot and a Histogram to analyze distribution.
14. Create a Pie Chart and a Bar Graph to represent categorical data (e.g., sales figures over a year).
15. Plot a 3D line graph using the plot3 function.
16. Generate a 3D Surface Plot (surf) and a Mesh Plot (mesh) for a mathematical function of two variables.
17. Write a script to define a polynomial, find its roots using the roots function, and evaluate it at specific points using polyval.

Suggested Reference Books:

1. MATLAB: A Practical Introduction to Programming and Problem Solving, 6th Edition by Stormy Attaway, Butterworth-Heinemann.
2. Getting Started with MATLAB: A Quick Introduction for Scientists and Engineers by Rudra Pratap, Oxford University Press.
3. MATLAB Programming for Engineers, 6th Edition by Stephen J. Chapman, Cengage Learning.

Semester-4

Discrete Structures

Course code	26PCC-CSE-202H				
Category	Program Core Course				
Course title	Discrete Structures				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4
	3	**1**	**0**	**4**	
Classwork	30 Marks				
Examination	70 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To develop a strong foundation in set theory, relations, functions, and propositional logic, enabling students to model and analyze discrete structures logically and mathematically.
2. To introduce fundamental counting principles and number-theoretic concepts**,** including recurrence relations, to enhance problem-solving skills relevant to computer science and engineering applications.
3. To familiarize students with algebraic structures such as groups, rings, fields, and Boolean algebra, and enable them to understand their properties and applications in abstract and applied contexts.
4. To enable students to analyze and apply graph-theoretic concepts**,** including trees, paths, circuits, and spanning trees, for solving real-world problems related to networks, optimization, and algorithms.
5. To build analytical and logical reasoning abilities necessary for advanced studies in mathematics, computer science, data structures, algorithms, and theoretical computing.

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1:Define the concepts of set theory, relations, functions, lattices, propositions, quantifiers, recurrence relations, algebraic structures, Boolean algebra, and graph theory with the support of examples.

CO2:Understand the significance and contribution of various concepts/ theorems/ algorithms such asCantor's diagonal argument, the Power Set theorem, Pigeon hole principle, Division algorithm, Euclidean algorithm, Boolean identities, shortest path algorithms, minimal spanning tree algorithms, etc.

CO3:Solverecurrence relations, permutation and combination problems.

CO4:Classify and **justify** the type of any relation, function, lattice, proposition, algebraic structure, and graph with the help of proof**.**

CO5:Develop the given problem as graph networks and solve with techniques of graph theory.

CO - PO mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1		3	1	-	2	-	-	-	-	-	-
CO2	3	3	2	2	2	-	-	-	2	-	-
CO3	3	3	2	1	2	-	-	-	2	-	-

CO4	3	3	1	3	2	-	-	-	2	-	-
CO5	3	3	2	2	3	-	-	-	2	-	-

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

Unit-I

Sets, Relation, Function and Propositional Logic: Operations on Sets, Laws of Sets, Cartesian Products, Representation of relations, Binary Relation, Equivalence Relation, Partial Ordering Relation, POSET, Hasse Diagram, Lattices and its types, Function, Bijective functions, Inverse and Composite Function, Countable and Uncountable Sets, Cantor's diagonal argument and The Power Set theorem, **S**chroeder-Bernstein theorem, Propositions, Logical operations, Conditional Statements, Tautologies, Contradictions, Logical Equivalence, The use of Quantifiers

Unit II

Basic Counting Techniques and Recurrence Relation: Pigeon-hole principle, Permutation and Combination**,** the Division algorithm: Prime Numbers, The GCD: Euclidean Algorithm, The Fundamental Theorem of Arithmetic., Linear recurrence relation with constant coefficients, Homogenous Solutions, Particular Solutions, Total Solutions, Solving recurrence relation using generating functions.

Unit III

Algebraic Structures: Definitions and examples of Algebraic Structures with one Binary Operation: Semi Groups, Monoids, Groups; Congruence Relation and Quotient Structures, Permutation Groups, Cyclic groups, Normal Subgroups, Definitions and examples of Algebraic Structures with two Binary Operation: Rings, Integral Domain, Fields; Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form

Unit IV

Graphs and Trees: Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Multigraph and Weighted graph, Shortest path in Weighted graphs, Eulerian paths and circuits, Hamiltonian path and circuits, Planar Graphs, Euler's formulae, Graph Colouring, Trees, Binary trees and its traversals, Trees Sorting, Spanning tree, Minimal Spanning tree

Text/Reference Books:

- Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw – Hill
- Satinder Bal Gupta: A Text Book of Discrete Mathematics and Structures, University Science Press, Delhi.
- C. L. Liu and D. P. Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, Tata McGraw – Hill.
- J.P. Tremblay and R. Manohar, Discrete mathematical structures with applications to computer science, TMG Edition, TataMcgraw-Hill
- Discrete Mathematics, Babu Ram, Pearson Publication
- Discrete Mathematics, SemyourLipschutz and Marc Lipson, Schaum's outline

Artificial Intelligence

Course code	26-PCC-AIML-221H				
Category	Program Core Courses				
Course title	Artificial Intelligence				
Scheme and Credits	L	T	P	Credits	Semester III
	3	0	0	3	
Classwork	25				
Examination	50				
Total	75				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. Understand the foundational concepts of AI, its history, nature, and real-world problem domains.
2. Apply and analyze uninformed, informed, local, population-based, and adversarial search techniques for problem solving.Represent knowledge using Propositional Logic, Predicate Logic, Semantic Nets, Frames, and Rule-Based Systems.
3. Handle uncertainty using Bayesian and Dempster-Shafer reasoning, and understand AI planning techniques.
4. Explore various learning paradigms, Neural Networks, Expert Systems, and current trends in AI.

COURSE OUTCOMES (CO):

CO1: Define the fundamental concepts of Artificial Intelligence. **Discuss** the nature and examples of AI Problems (Level 1, 2: Remember and Understand)**. Identify** the type of search strategy (blind/heuristic/adversarial) that is more appropriate to address a particular problem and implement the selected strategy. (Level 1,2: Remember and Understand)
CO2:Understand and **Apply**the concept of Knowledge Representation, Reasoning, planning and learning to solve AI Problems. (Level 2, 3: Understand and Apply)
CO3:Comparethe strength and weaknesses of various AI search techniques, Knowledge representation techniques, Reasoning techniques etc. (Level 4: Analyze)
CO4:Formulizationof AIproblems using the strategies introduced during the course. (Level 6: Create)

CO - PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-	3	-	-	-	-	-	-
CO2	3	2		-	3	-	-	-	-	-	-
CO3	3	3			3	-	-	-	-	1	
CO4	3	3			3	-				2	

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

UNIT 1

Introduction: Definition of AI, History of AI, nature of AI problems, examples of AI problems.

Problem solving by search: Uninformed Search: Depth First Search (DFS), Breadth First Search (BFS). Informed Search: Best First Search, A*. Local Search: Hill Climbing. Problem Reduction Search: AO*. Population Based Search: Ant Colony Optimization, Genetic Algorithm. Game Playing: Min Max Algorithm, Alpha-Beta Pruning.

UNIT 2

Knowledge Representation: Types of Knowledge, Knowledge Representation Techniques/schemes: Propositional Logic, Predicate Logic, Semantic nets, Frames. Knowledge representation issues. Rule based systems.

UNIT 3

Reasoning under Uncertainty: Basics of Probability Theory, Probabilistic Reasoning, Bayesian Reasoning, Dempster-Shafer Theory.

Planning: Introduction to Planning, Representation of Planning, Partial-order Planning.

UNIT 4

Learning: Introduction to Learning, Types of Learning: Learning by Induction, Rote Learning, Symbol Based Learning, Identification Trees, Explanation Based Learning, Transformational Analogy, Introduction to Neural Networks, Expert Systems, Current trends in Artificial Intelligence.

Alternative NPTEL/SWAYAM Course:

SNo	Swayam course name	Instructor	Host Institute
1.	**An Introduction to Artificial Intelligence**	Prof. Mausam	**IIT Delhi**

Suggested books:

1. Artificial Intelligence: A Modern Approach Third Edition Stuart Russell and Peter Norvig, 2010, Pearson Education.

Suggested reference books:

1. Elaine Rich, Kevin Knight, & Shivashankar B Nair, Artificial Intelligence, McGraw Hill, 3rd ed.,2009.
2. Introduction to Artificial Intelligence & Expert Systems, Dan W Patterson, PHI.,2010.
3. Artificial intelligence, Patrick Henry
4. Winston, 1992, Addition Wesley 3 Ed

Database Management System

Course code	26PCC-CSE-206H				
Category	Program Core Course				
Course title	Database Management System				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester 4
	3	**0**	**0**	**3**	
Class work	25 Marks				
Exam	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

Objectives of the course

1. To understand the different issues involved in the design and implementation of a database system, study the physical and logical database designs, database modeling, relational, hierarchical, and network models
2. To understand and use data manipulation language to query, update, and manage a database
3. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
4. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

Course Outcomes (CO)

CO1: For a given query write relational algebra expressions for that query and optimize the developed expressions, for a given specification of the requirement, design the databases using E R method and normalization.

CO2: For a given specification, construct the SQL queries for Open source and Commercial DBMS - MYSQL, ORACLE, and DB2, For a given query optimize its execution using Query optimization algorithms

CO3: For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.

CO4: Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

CO-PO Mapping:

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	1	2	2	2	2	1					
CO2	2	2	3	3	3	1			2	2	1
CO3	3	3	3	3	3	1	1	1	2	2	2
CO4	3	3	3	3	3	1	1	1	2	2	2

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

Unit: I

Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Unit: II

Relational query languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server.

Relational database design: Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design.

Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.

Unit: III

Storage strategies: Indices, B-trees, hashing, Transaction processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

Unit: IV

Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

Advanced topics: Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

Alternative NPTEL/SWAYAM Course

S.No.	NEPTEL course name	Instructor	Host Institute
1	Data Base Management System Prof. Samiran Chattopadhyay	Prof. Partha Pratim Das, Prof. Samiran Chattopadhyay	IIT Kharagpur

Suggested books:

1. "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

Suggested reference books

1. **"**Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.
2. "Fundamentals of Database Systems", 5th Edition by R. Elmasri and S.Navathe, Pearson Education
3. "Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley

Object Oriented Programming

Course code	26PCC-CSE-208H				
Category	Program Core Course				
Course title	Object Oriented Programming				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4
	3	**0**	**0**	**3**	
Class work	25 Marks				
Exam	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

Course Objectives:

1.To understand the basic concepts of Object Oriented Programming such as Data Abstraction andEncapsulation, classes & objects, Abstract Classes etc.

2. Identify different methods of Inheritance and Memory management using Pointers

3. To understand the concepts such as constructor, destructor, Operator Overloading, Virtual function.

4. Some advance concepts such as Exception Handling and Templates.

COURSE OUTCOMES (COs)

After successful completion of this course, students will be able to:

CO1: Explain the basic concepts of Object-Oriented Programming such as classes & objects, abstraction, encapsulation, inheritance etc.

CO2: Apply inheritance, pointers, and dynamic memory management to design programs based on object-oriented concepts.

CO3: Implement constructors, destructors, operator overloading, type conversion, and virtual functions in C++ programs.

CO4: Develop reliable and reusable programs using exception handling and templates.

CO-PO Mapping

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-	-	-	-	-	-	1	-
CO2	2	3	2	-	2	-	1	-	-	-	-
CO3	2	2	3	1	2	-	1	-	-	-	-
CO4	2	2	3	2	2	-	-	-	-	1	-

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

Unit - I

Object-Oriented Programming Concepts: Introduction, comparison between procedural programming paradigm and object-oriented programming paradigm, basic concepts of object-oriented programming — concepts of an object and a class, implementation of a class, operations on objects, relationship among objects, abstraction, encapsulation, data hiding, inheritance, overloading, polymorphism, messaging.

Classes and Objects: Specifying a class, creating class objects, accessing class members, access specifiers, static members, use of const keyword, friends of a class, empty classes, nested classes, local classes, abstract classes, container classes, bit fields and classes.

Unit - II

Inheritance: Introduction, defining derived classes, forms of inheritance, ambiguity in multiple and multipath inheritance, virtual base class, object slicing, overriding member functions, object composition and delegation, order of execution of constructors and destructors.

Pointers and Dynamic Memory Management: Declaring and initializing pointers, accessing data through pointers, pointer arithmetic, memory allocation (static and dynamic), dynamic memory management using new and delete operators, pointer to an object, this pointer, pointer related problems - dangling/wild pointers, null pointer assignment, memory leak and allocation failures.

Unit - III

Constructors and Destructors: Need for constructors and destructors, copy constructor, dynamic constructors, explicit constructors, destructors, constructors and destructors with static members, initializer lists.

Operator Overloading and Type Conversion: Overloading operators, rules for overloading operators, overloading of various operators, type conversion - basic type to class type, class type to basic type, class type to another class type.

Virtual functions & Polymorphism: Concept of binding - early binding and late binding, virtual functions, pure virtual functions, abstract classes, virtual destructors.

Unit - IV

Exception Handling: Review of traditional error handling, basics of exception handling, exception handling mechanism, throwing mechanism, catching mechanism, rethrowing an exception, specifying exceptions.

Templates and Generic Programming: Template concepts, Function Templates, Class Templates, illustrative examples.

Alternative NPTEL/SWAYAM Course (if any):

S.No.	NEPTEL course name	Instructor	Host Institute	Link
1	Programming in C++	Prof. Partha Pratim Das	IIT Kharagpur	https://nptel.ac.in/courses/106105151

TEXT BOOKS, AND/OR REFERENCE MATERIAL:

1. BjraneStroustrup, “C++ Programming language”,3rd edition, Pearson education Asia(1997)
2. LaforeR.”Object oriented Programming in C++”,4th Ed. Techmedia,New Delhi(2002).

3. E Balagurusamy,” Object oriented Programming with C++”, 6e, McGraw Hill Education (India), New Delhi.
4.Yashwant Kenetkar,”Let us C++”,1stEd.,Oxford University Press(2006)
5.B.A. Forouzan and R.F. Gilberg,CompilerScience,”A structured approach using C++” Cengage Learning, New Delhi.

Computer Networks

Course code	26PCC-CSE-210H				
Category	Program Core Course				
Course title	Computer Networks				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester 4/5
	3	**0**	**0**	**3**	
Class work	25 Marks				
Exam	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

Course Objectives:

1. To develop an understanding of modern network architectures from a design and Performance perspective
2. To make students able to analyse simple protocols.
3. To introduce the student to the major concepts involved in networks like addressing, routing, etc.
4. To introduce the network security aspects and basic primitives to build security.

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Students should be able to understand the basics of data communication, networking, the internet and their importance.

CO2: Understand and appreciate the computer communication protocols

CO3: Analyze the services and features of various protocol layers in data networks.

CO4: Evaluate the design of a computer communication networks and understand the concepts, and implement/ maintain a typical computer network and routing.

CO-PO MAPPING

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**
CO1	3	3	3	3	2	1	3	1	3	1	2
CO2	2	1	2	1	1	1	1	1	1	1	1
CO3	3	1	2	1	2	3	2	1	3	2	1
CO4	3	2	2	1	1	3	1	1	3	3	3

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

Unit-I

Introduction: Data communication,Components, Computer networks and their historical development,

Network Models: OSI model and TCP/IP Model

Physical Layer – Physical layer functions, Data Representation, Simplex, Half Duplex and Full Duplex Transmission, Packet and circuit switching, Topologies, connectionless and connection-oriented services.

Data Link Layer: Data link layer functions and services, MAC Addressing, Framing, Error control: - Parity bit, Hamming codes, CRC.

Medium Access Control: MAC layer functions, Random access-ALOHA, CSMA, CMA/CD, Controlled Access- Reservation, Polling, Token Passing, Channelization protocols-FDMA, TDMA, CDMA

Unit-II

Network Layer: Network layer functions and services, Logical addressing. IPv4 addressing: classful and classless addressing, subnetting, and special addresses. NAT, DHCP

Network Devices: Repeater, hub, bridge, switch, router.

Delivery and forwarding of IP Packets: Delivery-Direct and Indirect, Forwarding-Based on Label, Based on Destination Address, Structure of a router.

IPv4-datagrams, header format, fragmentation, ARP: Packet Format, its working, proxy ARP ICMPv4:- Message Format, Error Reporting Messages, Query Messages

Unit-III

Routing Algorithms: Introduction to routing, Shortest Path Algorithm, Flooding, Hierarchical Routing, Link State and Distance Vector Routing

Transport Layer: Transport layer functions and services, Process-to-Process Communication, User Datagram Protocol (UDP):- User datagram format, UDP services, applications. Transmission Control Protocol (TCP):- TCP services, features, TCP segment format, TCP connection management, retransmission, windows in TCP, Flow control -Silly Window Syndrome, Congestion control, TCP timers.

Unit-IV

Application Layer: Application layer functions and services, DNS, EMAIL, SMTP, File Transfer Protocol (FTP),

LAN and WAN: Ethernet:- Frame Format, Addressing, Standard, Fast, Gigabit and Ten-Gigabit ethernet. ATM :- Cell, ATM architecture, Layers

Congestion Control:- Open and Closed loop policies, **Quality of Service:** Flow Control to Improve QOS, QoS Improving techniques

Next Gen Protocols: IPv6 addressing:- Notations, Address Space, Address Types, Address space allocation, Global Unicast addresses, Autoconfiguration, IPv6:- Packet Format, IPv4vs IPv6. ICMPv6:- Introduction, different types of messages.

Alternative NPTEL/SWAYAM Course (if any):

S. No	Course Name	Instructor	Host Institute
1.	Computer Networks-Video course	Prof. Sujoy Ghosh	IIT Kharagpur

2.	Computer Networks and Internet Protocol	Prof. Soumya Kanti Ghosh, Prof. Sandip Chakraborty	IIT Kharagpur

Suggested books:

1. TCP/IP Protocol Suite, 4th edition, Behrouz A. Forouzan, McGraw-Hill.
2. Data Communications and Networking with TCPIP Protocol Suite, 6th Edition, Behrouz A. Forouzan, McGraw-Hill.
3. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.
4. Computer Networking: A Top-Down Approach by James F Kurose and Ross, 9th edition. Pearson India.

Suggested reference books:

1. Computer Networks, Andrew S. Tanenbaum, Pearson New International Edition.
2. Internetworking with TCP/IP, Volume 1, 6th Edition Douglas Comer, Prentice Hall of India.
3. TCP/IP Illustrated, Volume 1, W. Richard Stevens, Addison-Wesley, United States of America.

Database Management System Lab

Course code	26LC-CSE-212H				
Category	Program Core Course				
Course title	Database Management System Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4
	0	**0**	**2**	**1**	
Internal Lab	05 Marks				
External Lab	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

Course Objectives:

1. Keep abreast of current developments to continue their own professional development
2. To engage themselves in lifelong learning of Database management systems theories and technologies this enables them to purse higher studies.
3. To interact professionally with colleagues or clients located abroad and the ability to overcome challenges that arises from geographic distance, cultural differences, and multiple languages in the context of computing.
4. Develop team spirit, effective work habits, and professional attitude in written and oral forms, towards the development of database applications.

CO-PO Mapping

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	1	1	1	2	1			1		1
CO2	3	3	2	3	3	2			2	2	2
CO3	3	3	3	3	3	3	2		2	3	3
CO4	3	3	3	3	3	3	2		3	3	3

List of Practicals

1. Creation of a database and writing SQL queries to retrieve information from the database.
2. Performing Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions.
3. Creation of Views, Synonyms, Sequence, Indexes, Save point.
4. Creating an Employee database to set various constraints.
5. Creating relationship between the databases.
6. Study of PL/SQL block.
7. Write a PL/SQL block to satisfy some conditions by accepting input from the user.
8. Write a PL/SQL block that handles all types of exceptions.
9. Creation of Procedures.
10. Creation of database triggers and functions

11. Mini project (Application Development using Oracle/ MySQL)
    i. Inventory Control System
    ii. Material Requirement Processing.
    iii. Hospital Management System.
    iv. Railway Reservation System.
    v. Personal Information System.
    vi. Web Based User Identification System.
    vii. Time Table Management System.
    viii. Hotel Management

Object Oriented Programming Lab

Course code	26LC-CSE-214H				
Category	Program Core Course				
Course title	Object Oriented Programming Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester 4
	0	**0**	**2**	**1**	
Branches (B. Tech.)	Computer Science and Engineering				
Class work	5 Marks				
Exam	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. Understand basic concepts of **object-oriented programming** using classes and objects and Learn the use of **constructors, destructors, and data members**.
2. Apply **operator overloading and inheritance** in program design.
3. Understand **polymorphism and exception handling** techniques.
4. Develop skills in **generic programming using templates**.

COURSE OUTCOMES:

CO 1: Apply and use concepts of **classes and objects, constructors, destructors, and data members** effectively in C++ programs.

CO 2: Implement **operator overloading and inheritance**.

CO 3: Demonstrate **polymorphism and exception handling**.

CO 4: Develop **generic programs using templates**.

CO-PO Mapping:

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	1	2	-	-	-	1	1	-
CO2	3	2	2	1	2	-	-	-	1	1	-
CO3	3	2	1	1	2	-	-	-	1	1	-
CO4	3	2	2	1	2	-	-	-	1	1	-

List of Practicals(Using C++)

1. [Classes and Objects] Write a program that uses a class where the member functions are defined inside a class.
2. [Classes and Objects] Write a program that uses a class where the member functions are defined outside a class.
3. [Classes and Objects] Write a program to demonstrate the use of static data members.
4. [Classes and Objects] Write a program to demonstrate the use of const member functions.
5. [Constructors and Destructors] Write a program to demonstrate the use of zero argument, parameterized and copy constructors.

6. [Constructors and Destructors] Write a program to demonstrate the use of dynamic constructor.
7. [Constructors and Destructors] Write a program to demonstrate the use of explicit constructor.
8. [Initializer Lists] Write a program to demonstrate the use of initializer list.
9. [Operator Overloading] Write a program to demonstrate the overloading of increment and decrement operators.
10. [Operator Overloading] Write a program to demonstrate the overloading of binary arithmetic operators.
11. [Operator Overloading] Write a program to demonstrate the overloading of memory management operators.
12. [Inheritance] Write a program to demonstrate the multilevel inheritance.
13. [Inheritance] Write a program to demonstrate the multiple inheritance.
14. [Inheritance] Write a program to demonstrate the virtual derivation of a class.
15.[Inheritance] Write a program to demonstrate the order of execution of constructors and destructors in inheritance.
16 [Pointers] Write a program to demonstrate the use of this pointer.
17. [Polymorphism] Write a program to demonstrate the runtime polymorphism.
18. [Exception Handling] Write a program to demonstrate the exception handling.
19. [Templates and Generic Programming] Write a program to demonstrate the use of function template.
20. [Templates and Generic Programming] Write a program to demonstrate the use of class template.

TEXT BOOKS/ REFERENCE MATERIAL

1. BjraneStroustrup, “C++ Programming language”,3rd edition, Pearson education Asia(1997)
2. LaforeR.”Object oriented Programming in C++”,4th Ed. Techmedia,New Delhi(2002).
3. E Balagurusamy,” Object oriented Programming with C++”, 6e, McGraw Hill Education (India), New Delhi.
4.Yashwant Kenetkar,”Let us C++”,1stEd.,Oxford University Press(2006)
5.B.A. Forouzan and R.F. Gilberg,CompilerScience,”A structured approach using C++” Cengage Learning, New Delhi.

Computer Networks Lab

Course code	26LC-CSE-216H				
Category	Laboratory Course				
Course title	Computer Networks Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester 4/5
	0	**0**	**2**	**1**	
Class work	05 Marks				
Exam	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To design different network topologies and evaluate their performance under various conditions.
2. To understand practical aspects of networking concepts such as protocols, topologies, and layered architectures through hands-on experiments.
3. To gain experience in network configuration and management, including IP addressing, subnetting, and routing techniques.
4. To implement and analyze common networking protocols such as TCP, UDP, HTTP, FTP, and DNS using simulation tools and real-time applications.

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Understand fundamental underlying principles of computer networking

CO2: Understand details and functionality of layered network architecture and apply mathematical foundations to solve computational problems in computer networking.

CO3: Analyse performance of various communication protocols and compare routing algorithms.

CO4: Practice packet/file transmission between nodes.

Co-Po Mapping:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**
CO1	3	3	3	2	2	1	3	1	1	1	1
CO2	2	1	2	1	1	1	1	1	1	1	1
CO3	2	1	3	1	2	3	3	1	3	2	1
CO4	2	2	2	1	1	3	1	1	3	2	3

Note: Simulation tools like Wireshark /Cisco Packet Tracer / NS3 may be used.

List of Practicals

1. Study of networking devices, cables, and tools.
2. Design of network topologies (Bus, Star, Ring, Mesh)
3. IPv4 addressing and subnetting exercises

4. Network configuration using commands (ping, ipconfig, traceroute etc.)
5. Configuration of static and dynamic routing
6. Study and configuration of DNS
7. Packet capturing and analysis using Simulator
8. TCP client-server communication
9. UDP client-server communication
10. Simulation of routing algorithms (Distance Vector, Link State)

The experiments can also be performed using the Computer Networks Lab available at Virtual Lab (www.vlab.co.in).

The list of available practical/experiments is as follows:

1. **Carrier Sense Multiple Access with Collision Detection (CSMA/CD)**
2. **Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)**
3. **Subnetting**
4. **Distance Vector Routing Algorithm**
5. **Link State Routing Algorithm**
6. **Chat Application using UDP**
7. **File Transfer using TCP**
8. **Domain Name System (DNS)**

Artificial Intelligence Lab

Course code	26LC-AIML-215H				
Category	Program Core Course				
Course title	Artificial Intelligence Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4/5
	0	**0**	**2**	**1**	
Class work	05 Marks				
Exam	20 Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

Course Outcomes:

By the end of the Course, Students will be able to:

CO1: Formulate and Solve Search Problems. Apply fundamental AI search strategies (e.g., Breadth-First Search, Depth-First Search, \(A\star\) algorithm) by writing Python scripts to solve classic state-space and constraint satisfaction problems.

CO2: Develop Game Playing and Adversarial Agents. Design and implement competitive game-playing agents and adversarial search algorithms (such as Minimax and Alpha-Beta Pruning) using Python to solve puzzle and strategy scenarios.

CO3: Process Text and Build NLP Pipelines. Construct Natural Language Processing (NLP) solutions in Python using tools like the Natural Language Toolkit (NLTK) to perform text classification, tokenisation, stemming, and part-of-speech (POS) tagging.

CO4: Implement Supervised and Unsupervised Machine Learning. Utilise standard Python libraries (e.g., NumPy, Pandas, and Scikit-learn) to train, evaluate, and deploy foundational machine learning models for real-world data analysis and classification.

List of Experiments(Using Python):

1. Write a Program to Implement Breadth First Search Using Python.
2. Write a Program to Implement Depth First Search Using Python.
3. Write a Program to Implement Tic- Tac-Toe game using Python.
4. Write a Program to Implement 8-Puzzle Problem Using Python.
5. Write a Program to Implement Water-Jug Problem Using Python.
6. Write a Program to Implement Travelling Salesman Problem Using Python.
7. Write a Program to Implement Tower of Hanoi Using Python.
8. Write a Program to Implement Monkey Banana Problem Using Python.
9. Write a Program to Implement Missionaries-Cannibals Problem Using Python.
10. Write a Program to Implement 8-Queens Problem Using Python.

Note: At least 5 to 10 more exercises to be given by the teacher concerned.

The experiments can also be performed using the Artificial Intelligence using Python Lab available at Virtual Lab (www.vlab.co.in).

Suggested Books:

1. Artificial Intelligence: A Modern Approach Third Edition Stuart Russell and Peter Norvig, 2010, Pearson Education.

Suggested Reference book:

1. Elaine Rich, Kevin Knight, & Shivashankar B Nair, Artificial Intelligence, McGraw Hill, 3rd ed.,2009.
2. Introduction to Artificial Intelligence & Expert System, Dan W Patterson, PHI., 2010.
3. Artificial intelligence, Patrick Henny Winston, 1992, Addition Wesley 3 Ed.

CO - PO mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-	2	-	-	-	-	-	-
CO2	3	2		-	2	-	-	-	-	--	-
CO3		-	-	-	-	-	-	3	2		-
CO4		-	-	-	-	-	-	3	2		-

Professional Competency Lab

Course code	**26-PROJ-CSE-218H**				
Category	**Program Core Course**				
Course title	**Professional Competency Lab**				
Scheme and Credits	**L**	**T**	**P**	**Credits**	**Semester 4**
	0	**0**	**2**	**1**	
Class work	**05 Marks**				
Exam	**20 Marks**				
Total	**25 Marks**				
Duration of Exam	**03 Hours**				

Course Outcomes (COs)

CO1: Demonstrate effective written and oral communication skills in professional settings.

CO2: Apply quantitative aptitude, logical reasoning, and verbal ability skills to solve placement-oriented problems.

CO3: Develop professional documents, digital profiles, and interview competencies for career advancement.

CO4: Utilize innovative AI-enabled digital tools to address real-world engineering challenges.

Suggested Assignments

1. Diagnostic Placement Readiness Assessment.
2. Quantitative Aptitude Practice Tests.
3. Logical Reasoning Practice Tests.
4. Verbal Ability and Reading Comprehension Exercise.
5. Professional Email Writing Activity.
6. Resume Development Workshop.
7. LinkedIn and GitHub Profile Creation.
8. Programming Challenges.
9. Coding Contest using Online Assessment Platform.
10. Technical Presentation on Emerging Technologies.
11. Group Discussion Practice Session.
12. AI-Based Productivity Tool Exercises
13. Cybersecurity and Digital Awareness Activities
14. HR and Technical Interview Simulation.
15. Comprehensive Mock Placement Drive.

Recommended Reference Books

1. Quantitative Aptitude for Competitive Examinations – R.S. Aggarwal.
2. A Modern Approach to Verbal & Non-Verbal Reasoning – R.S. Aggarwal.

3. Cracking the Coding Interview – Gayle Laakmann McDowell.
4. The Google Resume.
5. Soft Skills.
6. Business Communication Today.

Course code: 26AU-ENV-202-H
Course Title: Environmental Science
Semester: III/IV
Branch: Compulsory for all branches of Engineering

Environmental Science							
Lecture	Tutorial	Practical/Field visit	Total contact hours per week	Credit	Theory	Practical/ Field visit	Total
02	0	01	03	0	70	30	100

Course Objectives: People working in industries or elsewhere essentially require the knowledge of environmental science so as to enable them to work and produce most efficient, economical and eco-friendly finished product. It will guide the students living in a historic transitional period of burgeoning awareness of the conflict between human activities and environmental constraints to help and save the fragile and endangered planet with the natural resources already overexploited.

- Solve various engineering problems applying environmentally sustainable approach to produce eco-friendly products.
- Use relevant pollution control methods to solve domestic and industrial problems.
- To recognize eco-friendly and sustainable energy resources for combating global problems like climate change, acid rain and resource depletion etc.
- To solve local solid hazardous and e-waste problems.

26AU-ENV-202-H: (ENVIRONMENTAL SCIENCE)

Theory 70 Marks Field Work 30 Marks (Practical/Field visit)

Unit-1 The Multidisciplinary nature of environmental studies. Definition, scope and importance. Structure and composition of atmosphere, hydrosphere, lithosphere and biosphere. Interaction between earth, man, and environment. (2 lectures)

Unit-2 Natural Resources:

Renewable and non-renewable resources: Natural resources and associated problems.

a) Forest resources: Use and over-exploitation; deforestation, case studies. Timber extraction, mining dams and their effects on forests and tribal people.

b) Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams- benefits and problems.

c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies.

d) Food resources: World food problems, changes, caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, Water logging, salinity, case studies.

e) Energy resources: Growing energy needs; renewable and non- renewable energy source, use of alternate sources, case studies.

f) Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification.

* Role of an individual in conservation of natural resources.

* Equitable use of resources for sustainable lifestyles.

(5 lectures)

Unit-3 Ecosystem and biodiversity conservation

* Structure of ecosystem, Biotic and Abiotic components
* Energy flow in the ecosystem.
* Ecological succession.
* Food chains, food webs and ecological pyramids.
* Aquatic and terrestrial ecosystems
* Different level of biodiversity: Genetic, Species and ecosystem diversity.
* Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values.
* Biodiversity at global, National and local levels.
* India as a mega-diversity nation.
* Hot-spots of biodiversity.
* Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts.
* Endangered and endemic species of India.
* Conservation of biodiversity: In-situ and ex-situ conservation of biodiversity.

(3 lectures)

Unit-4 Environmental Assessment and Management:

Aim**s** and objectives of environmental impact assessment (EIA), Environmental Impact Statement (EIS) and environment management plan (EMP). Structure and role of central and state pollution control board, ISO14000: implementation in industries benefit.

* Environment Protection Act.
* Air (Prevention and Control of pollution) Act.
* Water (Prevention and Control of pollution) Act.
* Wildlife Protection Act.
* Forest Conservation Act.
* Issues involved in enforcement of environmental legislation.
* Public awareness.

(5 lectures)

Unit-5 Environmental pollution:

Definition, causes, effects and control measures of :

a) Air pollution.
b) Water pollution
c) Soil pollution
d) Marine pollution
e) Noise pollution
f) Thermal pollution
g) Nuclear hazards

* Bag filters, cyclone separator, electrostatic precipitator, catalytic converter.
* Role of an individual in prevention of pollution.
* Pollution case studies.
* Disaster management: floods, earthquakes, cyclones, and landslides.

(5 lectures)

Unit-6 Contemporary environmental issues:

* From unsustainable to sustainable development.
* Urban problems related to energy, National solar mission.

* Water conservation, rain water harvesting, watershed management, Namami Gange and Yamuna action plan.
* Resettlement and rehabilitation of people: its problems and concerns case studies.
* Environmental ethics: Issues and possible solutions.
* Climate change, global warming, acid rain, ozone layer depletion, sea level rise, nuclear accidents and holocaust. Case studies.
* Wasteland reclamation.
* Carbon sequestration and carbon credits, carbon foot print

* Waste management- Swachha Bharat Abhiyan
* Consumerism and waste products.

(5 lectures)

Unit-7 Human population and the Environment.

Population growth, variation among nations. Population explosion- Family Welfare Programme. Environment and human health. Epidemiological issues: Fluorosis, Arsenocosis, Goitre, Dengue. Human Rights, Value Education, HIV/AIDS. Woman and Child Welfare. Role of Information Technology in Environment and human health. Case Studies. (6 lectures)

Unit-8 Field Work:

* Visit to a local area to document environmental assets - river/forest/grassland/hill/mountain.
* Visit to a local polluted site-urban/Rural/ Industrial/ Agricultural.
* Study of common plants, insects, birds.
* Study of simple ecosystems- pond, river, hill slopes, etc. (Field work equal to 15 lecture hours).

References

1. Agarwal, K.C. 2001 Environmental Biology, Nidi Pub. Ltd. Bikaner.
2. Bharucha, Frach, The Biodiversity of India, MApin Publishing Pvt. Ltd. Ahmedabad-380013, India, E-mail :mapin@icenet.net (R).
3. Brunner R.C. 1989, Hazardous Waste Incineration, Mc. Graw Hill Inc. 480p.
4. Clark R.S., Marine pollution, Slanderson Press Oxford (TB).
5. Cunningham, W.P. Cooper, T.H. Gorhani, E & Hepworth, M.T. 2001, Environmental Encyclopedia, Jaico Pub. House, Mumbai 1196 p.
6. De A.K., Environmental Chemistry, Wiley Eastern Ltd.
7. Down to Earth, Centre for Science and Environment (R).
8. Gleick, H.P., 1993. Water in crisis, Pacific Institute for Studies in Dev. Environment & Security Stockholm Env. Institute, Oxford Univ. Press, 473p.
9. Hawkins R.E. Encyclopedia of Indian Natural History, Bombay Natural History Society, Bombay (R).
10. Heywood, V.H. & Watson, R.T. 1995. Global Biodiversity Assessment, Cambridge Uni. Press 1140p.
11. Jadhav, H & Bhosale, V.M. 1995. Environmental Protection and Laws. Himalaya Pub. House, Delhi 284p.
12. Mackinney, M.L. & Schoch, RM 1996, Environmental Science systems & solutions, Web enhanced edition. 639p.
13. Mhaskar A.K., Mayyer Hazardous, Tekchno-S cience Publications (TB).
14. Miller T.G. Jr. Environmental Science, Wadsworth Publishing Co. (TB).
15. Odum, E.P. 1971, Fundamentals of Ecology. W.B. Saunders Co. USA, 574p.
16. Rao M.N. & Datta, A.K. 1987 Waste Water Treatment. Oxford & TBH Publ. Co. Pvt. Ltd. 345p.
17. Sharma, B.K. 2001, Environmental Chemistry, Goal Publ. House, Meerut.
18. Survey of the Environment, The Hindu (M).
19. Townsend C., Harper J. and Michael Begon. Essentials of Ecology, Blackwell Science (TB).

20. Trivedi R.K., Handbook of Environmental Laws, Rules, Guidelines, Comliances and Standards, Vol. I and II Enviro Media (R).
21. Tridevi R.K. and P.K. Goal, Introduction to air pollution, Techno Science Publications (TR).
22. Wagner K.D., 1998, Environmental Management, W.B. Saunders co. Philadelphia, USA 499p.
23. A text book environmental education G.V.S. Publishers by Dr. J.P. Yadav. (M) Magazine (R) Reference (TB) Textbook

Web Technologies

Course code	26PCC-CSE-212H				
Category	Professional Core Course				
Course title	Web Technologies				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4
	3	**0**	**0**	**3**	
Classwork	25Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To provide basic understanding of basic concepts of Web Technologies
2. To understand various client-side technologies.
3. To build interactive logic using JavaScript and DOM manipulation.
4. To teach Database connectivity and web services.

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Define basic concepts, lifecycle and applications of web technologies.
CO2: Understand style-heavy pages with CSS& JavaScript.
CO3: Know how to build login forms and contact pages that actually send and save information.
CO4: Students will establish themselves as effective professionals by solving real problems.

CO - PO mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	2	1	1	2					1	2
CO2	3	2	2	2	3				1	1	3
CO3	2	3	3	3	2				1	1	2
CO4	2	2	3	3	2				2	1	2

Note: Examiner will set 9 questions in all with two questions from each unit and one question covering all sections which will be Q.1. This Q1 is compulsory and of short answer type. Each question carries equal marks . Examinee has to attempt 5 questions in total selecting at least one from each unit.

Unit-I

Introduction to Web & Internet

Concepts of WWW, Internet, and HTTP/HTTPS Protocols, Web Browser architecture and Web Server functionality, Web 2.0 features, design principles, user-centric design, and sitemaps.

Web Design: Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache,Display resolution, Look and Feel of the Website, Page Layout and linking, User-centric design, Sitemap,Planning and publishing website, Designing effective navigation.

Unit-II

Client-Side Technologies (Front-End)

HTML/HTML5: Basics, formatting, lists, tables, images, forms, frames, and meta tags.

CSS/CSS3: Syntax, colors, borders, boxes, margins, padding, positioning, and responsive design.

Unit-III

Server-Side Programming (Back-End)

PHP: Basics, server-side scripting, and database interaction.

JavaScript: Client-side scripting, variables, functions, loops, DOM manipulation, event handling, and form validation, CGI, Active Server Pages (ASP).

Unit-IV

Database& Web Services

XML: Structure, parsing, and document type definitions.

AJAX: Asynchronous JavaScript and XML for dynamic page updates, JDBC/ODBC for connecting web apps to databases.

Alternative NPTEL/SWAYAM Course:

S.No.	NEPTEL course name	Instructor	Host Institute
1.	SWAYAM – Web-designing and multimedia Technologies.	Dr. B. Yogameena	National Institute of Technical Teachers' Training and Research, Chennai

Suggested Text Books:

1. Internet and World Wide Web How to program, P.J. Deitel& H.M. Deitel Pearson.
2. Steven Holzner,"HTML Black Book", Dremtech press.
3. Web Technologies, Black Book, Dreamtech Press
4. Web Applications : Concepts and Real World Design, Knuckles, Wiley-India

Reference Books:

1. Paul Deitel , Harvey Deitel, Abbey Deitel ,"Internet and world wide web – How to Program",
2. Web Technologies by Uttam K. Roy Oxford University Press.
3. Jennifer Robbins - Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics, O'Reilly Media, 2018
4. Jason Beaird and Alex Walker - The Principles of Beautiful Web Design, SitePoint, 2020 (4th Edition).

Web Technologies Lab

Course code	26LC-CSE-218H				
Category	Laboratory Course				
Course title	Web Technologies Lab				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4
	0	**0**	**2**	1	
Classwork	05 Marks				
Examination	20Marks				
Total	25 Marks				
Duration of Exam	03 Hours				

LIST OF EXPERIMENTS: -

1. Demonstrate basic HTML Tags, Table Tags, List Tags, Image Tags, Forms.
2. Design a Webpage for your college containing a description of courses, department, faculties, library etc. using list tags, href tags, and anchor tags.
3. Implement forms using HTML,FRAMES,CSS.
4. Write an HTML page that has one input, which can take multi-line text and a submit button. Once the user clicks the submit button, it should show the number of characters,lines and words in the text entered using an alert message. Words are separated with white space, and lines are separated with new line characters.
5. Use all the CSS (inline, internal and external) to format the college web page that you have created.
6. To write a program for implementing Library information using XML .
7. Create XML file to store Student Information like Register Number, Name, Mobile Number,DOB, and Email-Id.
8. Create XSL file to convert XML file to XHTML file.
9. Write an HTML/JavaScript page to create a login page with validations.
10. Develop a Simple calculator for addition, subtraction, multiplication and division operations using JavaScript.
11. Write a PHP Script to find out the Sum of the Individual Digits.
12. Write a PHP Script to check whether the given number is a palindrome or not.

Course Outcomes:

The computer lab course will consist of experiments illustrating the principles of web design relevant to the study of computer science and engineering.

The students will be able to:

CO1: Gain knowledge on designing static and dynamic web pages.

CO2: Able to validate web pages at client-side.

CO3: Design and validate XML documents.

CO4: Gain knowledge to develop a web site application.

Note: At least 10 experiments are to be performed by the students.

1). Each laboratory class/section shall not be more than about 30 students.

2). To allow fair opportunity of practical hands-on experience to each student, each experiment may either be done by each student individually or in group of not more than 3-4 students. Larger groups are strictly discouraged/disallowed.

3). Pre-experimental &post experimental quiz /questions may be offered for each lab experiment to reinforce & aid comprehension of the experiment.

CO - PO mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	3	3	2			1	2	2	2
CO2	3	2	2	2	3			1	2	2	3
CO3	3	3	3	2	2			1	2	2	2
CO4	3	2	3	2	2			1	2	2	2

Suggested Books:

1. Laura Lemay, Mastering HTML, CSS & Java Script Web Publishing, BPB Publications, 2016
2. Thomas A. Powell, The Complete Reference HTML & CSS, Fifth Edition, 2017
3. Tanweer Alam, Web Technologies, Khanna Book Publishing, 2011.

Web Resources:

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

Annexure-I

Computer Graphics

Course code	26PEC-CSE-202H				
Category	Professional Elective Course - I				
Course title	Computer Graphics				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester 4
	3	**0**	**0**	**3**	
Classwork	25 Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

Course Objective:

The objectives of this course are:

1. To introduce the **fundamental concepts of computer graphics**, graphics hardware, and display systems.
2. To study **scan conversion algorithms** for drawing basic graphical primitives such as lines, circles, and polygons.
3. To understand **2D and 3D geometric transformations** and their applications in graphical object manipulation.
4. To learn **viewing and clipping techniques** used for rendering graphical scenes.
5. To explore **illumination models, shading techniques, curves, surfaces, and animation** used in modern graphics applications.

Course Outcomes (CO)

After successful completion of this course, students will be able to:

CO1: Understand basic concepts, graphics systems, and display devices.

CO2: Apply line, circle, and polygon drawing algorithms.

CO3: Implement 2D and 3D geometric transformations and clipping techniques.

CO4: Analyze visible surface detection, illumination, and shading models.

CO-PO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**
CO1	3	2	1								
CO2	3	3	2	1	1						
CO3	2	3	3	2	2						
CO4	2	3	2	3	2						

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

UNIT-I

Introduction and Scan Conversion Algorithms

Introduction to Computer Graphics, Applications of Computer Graphics, Graphics Hardware and Software, Raster Scan and Random Scan Displays, Input Devices and Display Devices, Scan Conversion Algorithms: Line Drawing Algorithms: DDA Algorithm, Bresenham Line Algorithm; Circle Drawing Algorithms: Midpoint Circle Algorithm; Polygon Filling Algorithms: Boundary Fill Algorithm, Flood Fill Algorithm.

UNIT-II

2D Transformations and Viewing

Basic Geometric Transformations: Translation, Rotation, Scaling, Reflection, Shearing; Homogeneous Coordinates; Composite Transformations; Windowing and Clipping: Cohen-Sutherland Line Clipping; Liang-Barsky Algorithm, Polygon Clipping.

UNIT-III

3D Transformations and Object Representation: 3D Graphics Concepts; 3D Transformations: Translation, Rotation, Scaling; Parallel and Perspective Projections; Visible Surface Detection: Back Face Detection, Z-Buffer Algorithm, Painter's Algorithm; Curves and Surfaces: Bezier Curves, B-Spline Curves, Parametric Curves

UNIT-IV

Illumination, Shading and Applications

Illumination Models: Ambient Reflection, Diffuse Reflection, Specular Reflection; Shading Techniques: Flat Shading, Gouraud Shading, Phong Shading; Ray Tracing; Animation Techniques

Text Books

1. Foley, Van Dam, Feiner & Hughes – **Computer Graphics: Principles and Practice**.
2. Donald Hearn & Pauline Baker – **Computer Graphics (C Version)**.
3. Edward Angel – **Interactive Computer Graphics using OpenGL**.

Reference Books

1. Steven Harrington – **Computer Graphics: A Programming Approach**
2. Hill Jr. – **Computer Graphics using OpenGL**
3. Rogers – **Procedural Elements of Computer Graphics**

NPTEL Video Lectures

1. **Computer Graphics – IIT Delhi**
   https://nptel.ac.in/courses/106102063
2. **Computer Graphics – IIT Madras**
   https://nptel.ac.in/courses/106106090

3. **Introduction to Computer Graphics – IIT Bombay**
   https://nptel.ac.in/courses/106101001

Cloud Computing

Course code	26PEC-CSE-204H				
Category	Professional Elective Courses-I				
Course title	Cloud Computing (Theory)				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4
	3	**0**	**0**	**3**	
Classwork	25Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To introduce cloud computing concepts, AWS global infrastructure, core services, pricing models, and billing mechanisms.
2. To develop understanding of cloud security principles, identity and access management (IAM), networking concepts, and secure data handling.
3. To enable students to explore and utilize AWS compute services (VMs, containers, serverless) along with storage and database solutions, including their selection and management.
4. To familiarize students with cloud architecture concepts such as high availability, scalability, monitoring, and cost optimization using AWS tools.

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1: Describe cloud computing concepts, service models, and cloud services offered by providers such as AWS.

CO2: Explain cloud service implementation models, along with security and basic networking concepts in cloud environments.

CO3: Analyze and utilize AWS compute and storage services, including EC2, storage volumes, and their deployment and management.

CO4: Evaluate AWS features such as Auto Scaling and Elastic Load Balancing for achieving scalability, high availability, and performance optimization.

CO - PO mapping

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	-	2	-	-	-	-	2	-
CO2	3	3	1	1	2	2	1	1	-	2	-
CO3	3	3	2	2	3	-	-	-	1	2	1
CO4	3	3	3	2	3	1	1	-	1	2	2

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students

have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

Unit I

Cloud Concepts Overview: Introduction to cloud computing, Introduction to AWS, Moving to the AWS Cloud, Cloud Economics and Billing, Fundamentals of pricing, Total cost of ownership, AWS Organizations, AWS billing and cost management.

AWS Global Infrastructure Overview: AWS global infrastructure, AWS services and service categories, AWS Management Console, AWS shared responsibility model, AWS IAM, Securing a new AWS account.

Unit II

Cloud Security and Networking: Securing accounts, securing data, working to ensure compliance, Networking basics, Amazon VPC, VPC networking, Amazon VPC Console, VPC security, Route 53, CloudFront.

Compute: Compute services overview, Amazon EC2, Amazon EC2 Versus Managed Services, Amazon EC2 cost optimization, Container services, Introduction to AWS Lambda, Introduction to AWS Elastic Beanstalk.

Unit III

Storage and Databases Amazon Elastic Block Store Console, AWS S3, AWS EFS, AWS S3 Glacier, Storage Technology Selection, Amazon RDS, Build a Database Server, Amazon DynamoDB, Amazon Redshift, Amazon Aurora.

Unit IV

Cloud Architecture :AWS Well-Architected Framework design principles, Operational excellence, Security, Reliability, Performance efficiency, Cost optimization, Reliability & high availability, AWS Trusted Advisor Elastic Load Balancing, Amazon CloudWatch, Amazon EC2 auto scaling.

Alternative NPTEL/SWAYAM Course:

S.No.	NEPTEL course name	Instructor	Host Institute
1.	Cloud Computing	Prof. Soumya Kanti Ghosh	Indian Institute of Technology Kharagpur

Suggested Text Books:

1. Ben Piper, David Clinton – *AWS Certified Solutions Architect Study Guide*
2. Anthony T. Velte, Toby J. Velte – *Cloud Computing: A Practical Approach*
3. Andreas Wittig, Michael Wittig – *Amazon Web Services in Action*
4. Michael J. Kavis – *Architecting the Cloud: Design Decisions for Cloud Computing Service Models*

Reference Books:

1. Welcome to AWS Documentation, Amazon.
2. Cloud Computing, A Practical Approach by Toby Velte, Anthony Velte, Robert C. Elsenpeter, MC Graw Hill

Software:

AWS Academy Platform

Optimization Techniques

Course code	26PEC-CSE-206H				
Category	Professional Elective Courses-I				
Course title	Optimization Techniques (Theory)				
Scheme and Credits	**L**	**T**	**P**	**Credits**	Semester-4
	3	**0**	**0**	**3**	
Classwork	25 Marks				
Examination	50 Marks				
Total	75 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVES:

1. To study terminologies related to Natural language processing & its models.
2. Formulate real-world problems as optimization models
3. Apply classical and modern optimization algorithms
4. Analyze optimality conditions mathematically
5. Use optimization in AI, ML, networks, and operations research

COURSE OUTCOMES (CO):

By the end of the Course, Students will be able to:

CO1:Define fundamental concepts of Natural Language Processing including linguistic levels, text preprocessing techniques, and feature extraction methods.
CO2:Understand real-world engineering problems as linear programming models and solve them using graphical and simplex methods.
CO3:Apply classical machine learning algorithms and deep learning models for solving Natural Language Processing tasks.
O4:Analyze optimization problems by identifying feasible regions, optimal solutions, and different classes of optimization problems.

CO - PO mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**
CO1	3	2	-	-	1	-	-	-	-	1	-
CO2	3	3	2	2	3	-	-	-	1	1	-
CO3	3	3	2	2	2	-	-	-	-	1	1
CO4	3	3	2	2	2	-	-	-	-	1	-

Note: The examiner will set a total of 9 questions. Question one will carry questions from all units, and the remaining eight questions will be set by taking two questions from each unit. The students have to attempt five questions in total, with the first being compulsory and selecting one from each unit. All questions will carry equal marks.

UNIT-I

Introduction to optimization, engineering applications of optimization, Formulation of structural optimization problems as programming problems, Optimization techniques, Introduction to NLP, scope, challenges and its applications, Morphology, Syntax, Semantics, Pragmatics, Ambiguity in natural language, Tokenization, Stop word removal, Stemming and Lemmatization, Normalization, TF-IDF, Bag of Words (BoW), N-grams, TF-IDF.

UNIT-II

Classical ML for NLP: Naïve Bayes, Logistic Regression, Support Vector Machine (SVM), Decision Trees.

Deep learning models: Recurrent Neural Networks (RNN), LSTM & GRU, Sequence-to-Sequence models.

UNIT-III

Introduction to Linear Programming: Formulation, graphical method, simplex, Dual simplex, North west rule, least cost rule.

Optimization Techniques: Introduction to optimization, classification, convex vs non-convex problems, feasible region, optimal solution.

UNIT-IV

Genetic Algorithms (GA): Introduction to genetic algorithms, basic principles and biological inspiration, Encoding strategies such as binary encoding, permutation encoding, value encoding and tree encoding.

Genetic operators: Selection methods, crossover operators, mutation operators, fitness function and GA cycle, problem solving using GA.

Alternative NPTEL/SWAYAM Course

SNo	NEPTEL course name	Instructor	Host Institute
1.	Optimization	Prof. A Goswami	IITM
2.	Optimization Techniques	Sukanta Nayak	Youtube

Suggested Text Books:

1. Foundations of Statistical NLP – Manning & Schütze
2. Operations Research by Hamdy Taha
3. Convex Optimization by Boyd
4. Introduction to Linear Optimization by Bertsimas
5. Speech and Language Processing – Daniel Jurafsky & James Martin
6. Operations Research", H.S. Kasene& K.D. Kumar, Springer (India), Pvt. Ltd.
7. "Engineeringoptimization:Theoryandpractice",S.S.Rao,NewAgeInternational (P)Limited,3rdedition,1998.